



CYBERARK[®]

Privileged Account Security Implementation Guide

Version 9.6

Including:

Privileged Identity Management Suite

Privileged Session Management Suite

Important Notice

Conditions and Restrictions

This guide is delivered subject to the following conditions and restrictions:

- This guide contains proprietary information and ideas belonging to CyberArk Software Ltd. which are supplied solely for the purpose of assisting explicitly and properly authorized users of the CyberArk software.
- No part of its contents may be used for any other purpose, disclosed to any person or firm or reproduced by any means, electronic and mechanical, without the express prior written permission of CyberArk Software Ltd.
- The software described in this document is furnished under a license. The software may be used or copied only in accordance with the terms of that agreement.
- Information in this document, including the text and graphics which are made available for the purpose of illustration and reference only, is subject to change without notice. Corporate and individual names and data used in examples herein are fictitious unless otherwise noted.
- Third party components used in the CyberArk Vault may be subject to terms and conditions listed on www.cyberark.com/privatemark/acknowledgement.htm.

Acknowledgements

- This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (<http://www.openssl.org/>).
- This product includes cryptographic software written by Eric Young (eay@cryptsoft.com).
- This product includes software written by Tim Hudson (tjh@cryptsoft.com).
- This product includes software written by Ian F. Darwin.
- This product includes software developed by the ICU Project (<http://site.icu-project.org/>) Copyright © 1995-2009 International Business Machines Corporation and other. All rights reserved.

Copyright

© 1999-2016 CyberArk Software Ltd. All rights reserved.

CyberArk®, the CyberArk logo, and all other names and logos that appear in this guide are trademarks of CyberArk Software Ltd. and their respective owners.

Information in this document is subject to change without notice.

PASIMPG-009-6-0-1

Table of Contents

Introducing the Privileged Account Security Solution	13
The Privileged Account Security Solution.....	14
Solution Benefits.....	15
Vaulting Technology®.....	16
Security Layers.....	16
How the Vault Protects your Passwords.....	18
Privileged Account Security Solution Architecture.....	19
The Vault.....	20
The Password Vault Web Access Interface.....	20
PrivateArk Administrative Interfaces.....	21
Privileged Session Manager.....	21
Privileged Session Manager SSH Proxy.....	22
The Central Policy Manager.....	22
The On-Demand Privileges Manager.....	23
The Password Upload Utility.....	23
SDK Interfaces.....	23
Administrative APIs.....	24
Working with the Privileged Account Security Solution	25
Getting Started with the PrivateArk Administrative Client.....	26
Defining a Vault.....	26
Logging onto the Vault.....	30
Logging onto the PrivateArk Client.....	30
Logging on through a Browser Interface.....	31
Logging onto the PVWA.....	32
Creating and Managing Locations, Users, and Groups.....	39
Managing Locations.....	39
Managing Users.....	40
Network Areas.....	51
Trusted Network Areas.....	54
Managing Groups.....	56
Predefined Users and Groups.....	59
Inspecting User Activity.....	63
Creating and Managing Safes and Owners.....	64
Adding and Managing Safes.....	65
Adding and Managing Safe Members.....	68
Object Level Access Control.....	76
Advanced Safe Management.....	79
Transparent User Management.....	86
Managing Directory Maps.....	87
Modifying External User Accounts.....	92
Managing Safe Ownership for LDAP Users and Groups.....	96
Configuring LDAP Connectivity for Users' Logon.....	99
Managing Users and Groups who are Listed in Multiple Directories.....	100
The Master Policy.....	102
Working with Master Policy Rules.....	103
Exceptions.....	105

Auditing Master Policy Activity	106
Viewing the Effective Policy by Platform	106
Managing Target/Service Account Platforms	108
Modifying Target Account Platforms	109
Adding New Platforms.....	110
Deleting Platforms	111
Activating and Deactivating Platforms	112
Limit Platforms to Specific Safes.....	119
Linked Accounts	120
Creating Accounts.....	122
Adding Accounts.....	122
Managing Service Accounts.....	151
Provisioning Accounts Automatically	154
Checking for New Accounts	160
Defining Custom Account Properties	161
Moving Accounts between Safes	163
Accounts Feed	168
Supported Target Machines	169
Managing Discovery Processes.....	170
Pending Accounts.....	183
Viewing Account Dependencies.....	188
Search and Filtering Options.....	190
Onboarding Accounts.....	191
Automatic and Manual Account Management.....	197
Changing Passwords	197
Verifying Passwords.....	205
Reconciling Passwords	207
Editing Account Properties	210
Disabling Automatic Account Management	212
Account Groups	215
Linked Accounts	220
Account Access Workflows.....	222
Accessing Accounts and Service Accounts.....	222
Finding Accounts and Service Accounts	228
Managing the Accounts List.....	231
Customizing your own Views	235
Selecting Accounts	236
Managing Accounts and Service Accounts	237
Modifying Accounts	239
Viewing Passwords	240
Copying Passwords.....	241
Accessing Accounts	243
Specifying a Reason for Accessing Passwords	244
Integration with Ticketing Systems	244
Accounts Check-out and Check-in.....	246
Dual Control.....	251
Privileged Single Sign-On	267
Connecting with Secure Connect.....	295
Accessing the Connection Window (Direct Access to Target Systems).....	299
Working in Split Password Mode	300
Password Version Control.....	302

Accessing Accounts through the Mobile PWVA.....	303
Logging onto the Mobile PWVA	303
Finding Accounts	304
Viewing Passwords	305
Changing Passwords	305
Creating Requests.....	306
Confirming Requests.....	307
Accessing the PWVA Mobile Version	309
On-Demand Privileges for UNIX Environments	310
Using the On-Demand Privileges Manager	310
Playing Session Recordings	313
Monitoring Privileged Sessions.....	315
Monitoring Privileged Session Recordings	316
Monitoring Live Sessions	333
Customizing your own Views	337
Auditing Accounts.....	339
Inspecting Accounts Activity.....	339
The Privileged Account Security Solution Reports	340
Adding and Managing Files and Documents.....	366
The Central Policy Manager	369
Administrating the Central Policy Manager	370
Managing the Central Policy Manager.....	370
Configuring the Central Policy Manager	371
CPM Activity Logs	372
CPM Log Errors Email Notifications.....	376
Configuring Accounts for Automatic Management.....	377
Applying Platforms.....	377
Managing Password Change Processes.....	377
Verifying Passwords.....	379
Reconciling Passwords	379
Associating Logon Accounts	382
Setting Up Supported Platforms.....	383
Operating Systems.....	384
Databases.....	425
Remote Access	454
Security Appliances	456
Network Devices.....	459
Directories.....	463
Applications	465
Cloud Services	491
Configuring Service Accounts.....	492
Adding Service Account Platforms.....	492
Deleting Service Account Platforms.....	492
Password Vault Web Access	493
Configuring the PWVA	494
General PWVA Configurations.....	495
Displaying Sign in Information.....	500
Configuring the Accounts List	500
Accounts Feed.....	502
User Preferences.....	505
Adding Safe Members from LDAP Directories	505

Searching for Accounts and Files	507
Identifying Accounts and Files	508
Configuring Statistics.....	509
Accounts Check-out and Check-in.....	509
Dual Control.....	513
Integrating with Ticketing Systems	516
Configuring Dual Control together with Ticketing Integration	523
Configuring Transparent Connections	527
Accessing PWWA from another Web Application	562
Reports	562
Logging	568
Caching.....	568
Configuring the Account Retrieval Form	568
Configuring Split Password Mode	569
Adding Accounts.....	570
Adding Safes	570
Configuring the PWWA Interface	570
Configuring the Mobile Password Vault Web Access	572
Configuring Authentication to the Mobile PWWA.....	572
General Configurations	572
Configuring Mobile Devices	572
Displaying Application Links.....	572
Displaying Accounts Search Results.....	573
Displaying Account Details.....	573
Displaying Request Details	573
Defining Password Details	573
Configuring the PWWA Mode	574
Direct Access to PWWA Pages	575
Accessing the Account Details Page	575
Accessing Accounts Search Results	575
Accessing Request Pages	576
Accessing Privileged Session Recordings.....	576
Accessing PSM Live Sessions.....	577
Logging	578
Disaster Recovery.....	579
Privileged Session Manager	580
Administrating the Privileged Session Manager	581
Managing the Privileged Session Manager	581
PSM Activity Logs.....	581
Architecture	583
Enabling Privileged Session Management.....	584
Configuring PSM Users.....	584
Setting PSM in the Master Policy.....	584
Customizing PSM for Specific Platforms	585
Enabling Session Recordings for Specific Users and Groups	586
Configuring Secure Connect.....	587
Configuring Privileged Session Management	592
Configuring Recordings and Audits	592
Configuring the Privileged Session Management Interface	610
Configuring Live Session Monitoring	617
Configuring PSM Connections	621
Configuring the Privileged Session Manager.....	682

Accessing PSM Recording Sessions Directly	685
Disaster Recovery	686
Auditing in PSM.....	687
Privileged Session Manager SSH Proxy	688
Introduction.....	689
Monitoring Privileged Sessions	690
Architecture	691
Configuring the PSMP	692
Configuring Accounts for Privileged Single Sign-On.....	692
Configuring Authentication Methods	693
Configuring Platforms to Enable Connections through the PSMP	694
Configuring Platforms for Copying Files with PSMP	695
Configuring SSH Key Authentication to Target Systems	696
Configuring Login Sequences	696
Using Logon Accounts for SSH and Telnet Connections	698
Configuring Accounts to Provide Specific Connection Methods	703
Authenticating with your Personal Password.....	705
Specifying a Reason for Accessing Accounts	706
Configuring Recordings and Audits	707
Configuring SSH Tunneling for PSMP	711
Configuring a Subnet Mask.....	713
Configuring UNIX Domain/NIS Accounts.....	716
Configuring PSMP Syntax Delimiters	717
Configuring the Verification of a Server's Host Key.....	719
Displaying Notifications when Sessions are Recorded.....	719
Administrating the PSMP	720
Managing the PSMP Service	720
Administrating the SSH Proxy Machine.....	721
Defining Platforms for Assorted Scenarios.....	722
Integrating with AD Bridge Capabilities	724
Architecture.....	725
Configuring the PSMP to Integrate with AD Bridge Capabilities	726
Monitoring PSMP Integration with AD Bridge Capabilities	736
Auditing.....	737
Disaster Recovery	738
On-Demand Privileges Manager	739
Introduction.....	740
Overview	742
Architecture	744
Implementing the On-Demand Privileges Manager	745
Enabling the On-Demand Privileges Manager	747
Defining Privileged Commands.....	748
Defining Commands.....	749
Modifying Command Permissions	760
Defining Privileged Command Groups.....	762
Configuring the On-Demand Privileges Manager.....	764
Displaying the Commands List.....	764
Configuring the On-Demand Privileges Manager in Platforms	764
Shared and OPM-specific Configuration	770

Caching	773
Issuing Privileged Commands When There Is No Direct Connection to the Vault.....	777
Authenticating Users	778
Updating OPM Activity in the Vault	778
Integrating with UNIX Centralized User Management Products	779
Disaster Recovery	781
Accessing the Password Vault.....	782
Managing the OPM	782
Auditing.....	783
Monitoring.....	784
Configuring Auditing and Monitoring Log Files	786
Administrating the On-Demand Privileges Manager	787
Password Upload Utility	792
Implementing the Password Upload Utility	793
Creating the Vault Environment Automatically	793
Creating the Password File	796
Specifying Passwords in the Password File	797
Configuring the Password Upload Utility	799
Running the Password Upload Utility	801
Event Notification Engine	802
Enabling the ENE	803
Configuring the ENE	805
Implementing Dynamic References	805
Configuring Email Notification Templates	815
Configuring Recipients	818
Configuring Vault Notifications	831
Configuring CPM Notifications	832
Logging	834
Operating the CyberArk Vault	835
Working with the Server Interface.....	836
Starting the Server.....	837
Stopping the Server.....	838
Viewing the Server Log.....	839
Managing the Vault Workload.....	841
Controlling Vault Concurrency Level.....	841
Controlling Vault Tasks Allocation	841
Managing Long Transactions.....	843
Managing Long Transactions Manually	845
Specifying Administration Settings	846
Accessing the Vault through a Gateway Account	847
Monitoring the Vault	848
Remote Administration for the Vault/DR Vault.....	848
Configuring Remote Monitoring	855
Monitoring the CyberArk Firewall.....	861
Monitoring Backup and DR Replications	861
Integrating with SIEM Applications	862
Troubleshooting	867

Managing the CyberArk License.....	868
Monitoring the User License	868
Reporting License Usage	869
Installing a New License.....	869
Information Recovery	870
Restoring Safes or the Vault	870
Restore Utilities	874
CyberArk HA Vault Cluster	879
Adding a New Node	879
CyberArk Disaster Recovery Vault	885
Disaster Recovery User	885
Maintaining the DR Vault.....	886
Replicating Component User Passwords	886
Monitoring Vault Availability	887
Logging	887
Advanced Digital Vault Environment	888
CyberArk Vault Structure	889
Server Components.....	889
Server Files.....	890
Server Utilities	891
Server Keys	903
The CyberArk Client	905
PrivateArk Client Components	905
PrivateArk Client Files	906
PrivateArk Client Configuration	908
The PrivateArk Client in a Remote Desktop Services (Terminal Services) Environment.....	914
The PrivateArk Client Log File	915
PrivateArk Information	916
System Configuration.....	917
Configuring the System through PWWA	918
Appendices	923
Appendix A: Account Properties	924
Appendix B: Creating User Credential Files	956
CreateCredFile Utility	956
CreateAuthFile Utility.....	968
Appendix C: Configuring Debug Levels.....	971
Appendix D: Managing Platforms for Groups.....	980
Appendix E: Adding Accounts with SSH Keys using the AccountUploader Utility	984
Appendix F: Accessing Target Machines through PSMP	985

About this Book

This book describes the Enterprise Password Vault (EPV), the Privileged Session Management (PSM), and the On-Demand Privileges Manager (OPM) components of the Privileged Account Security (PAS) solution and their implementation processes. The book has been divided into several sections, as follows:

- The first part of the book introduces you to the Privileged Account Security solution and its unique concepts and features. It introduces you to the different components and guides you through Administration and Management tasks.
- The second part of the book guides you through administration and management, including maintenance procedures.
- The third part of the book guides you through system administration and management, including maintenance procedures.
- The fourth part of the book lists configuration files that are relevant to implementing the Privileged Account Security solution.
- The fifth and final part of the book includes appendices that contain additional information that is relevant to implementing and maintaining the Privileged Account Security solution.

The Application Identity Manager (AIM) component of the Privileged Account Security (PAS) solution is described in the Application Identity Management Implementation Guide, which explains how to work with the Application Password SDK, and how to implement the Application Password Provider and the Application Server Credential Provider.

Who should use this book?

This book is intended for system and Vault administrators who implement the Privileged Account Security solution, and users who access the information stored inside it.

What you should know before reading this book

To implement and use the Privileged Account Security solution, you must have a working knowledge of Windows administration.

How to use this book

The Implementation Guide is divided into four parts:

Part 1: Introducing the Privileged Account Security Solution

- Chapter 1, Introducing the Privileged Account Security solution, gives an overview of the Vault's benefits and features and introduces its various components. This chapter explains how the Vault manages automatic password management, while facilitating constant security for your enterprise passwords.

Part 2: Administration and Management

- Chapter 2, Working with the Privileged Account Security solution, describes how to set up the Privileged Account Security solution and how to carry out basic password management activities so that you will be able to begin working.
- Chapter 3, The Central Policy Manager, describes how to administrate and configure the Central Policy Manager (CPM).
- Chapter 4, Password Vault Web Access, describes how to administrate and configure the Password Vault Web Access (PVWA).
- Chapter 5, Privileged Session Management, describes how to administrate and configure the Privileged Session Management (PSM).
- Chapter 6, On-Demand Privileges Manager, describes how to administrate and configure the On-Demand Privileges Manager (OPM).
- Chapter 7, Password Upload Utility, introduces you to the Password Upload utility and guides you through setup and implementation.
- Chapter 8, Event Notification Engine, describes how to administrate and configure the Event Notification Engine (ENE).

Part 3: System Administration and Management

- Chapter 9, Operating the CyberArk Vault, describes how to administrate the CyberArk Vault server.
- Chapter 10, Advanced Digital Vault Environment, describes the Vault environment and how to configure it.

Part 4: System Configuration

- Chapter 11, System Configuration, describes the configurations and parameter files used to configure the Privileged Account Security solution.

Part 5: Appendices

- The Appendices contain additional information that you will use during implementation in order to configure the Privileged Account Security solution.

CyberArk Components

The following table displays the CyberArk components that are released with this version of Privileged Account Security solution, and their abbreviations:

Component	Abbreviation
Privileged Identity Management	PIM
Enterprise Password Vault	EPV
Central Policy Manager	CPM
Password Vault Web Access	PVWA
Privileged Session Manager	PSM
On-Demand Privileges Manager	OPM
Application Identity Management	AIM
Note: This component is covered in the Application Identity Management Implementation Guide.	
Disaster Recovery Vault	DR Vault
Event Notification Engine	ENE

Related Documents

- Privileged Account Security Installation Guide describes how to prepare the environment and install CyberArk's Privileged Account Security solution.
- Application Identity Management Implementation Guide describes how to work with the Application Password SDK and implement the Application Password Provider and the Application Server Credential Provider.
- CLI Reference Guide lists the CyberArk Vault Command Line Interface scripts and describes how to work with them.
- ActiveX API Help lists the CyberArk Vault ActiveX APIs and describes how to work with them.
- .Net API Help lists the CyberArk Vault .Net APIs and describes how to work with them.

Introducing the Privileged Account Security Solution

The CyberArk's Privileged Account Security (PAS) solution, a full life-cycle solution for managing the most privileged accounts and SSH Keys in the enterprise, enables organizations to secure, provision, manage, control and monitor all activities associated with all types of Privileged Identities such as administrator on a Windows server, Root on a UNIX server, Cisco Enable on a Cisco device, as well as embedded passwords found in applications and scripts.

This chapter introduces you to the Privileged Account Security solution, as well as CyberArk's Vaulting Technology®, a sophisticated state-of-the-art infrastructure for secure password storage and management.

- The Privileged Account Security Solution
- Vaulting Technology®
- Privileged Account Security Solution Architecture

The Privileged Account Security Solution

CyberArk's Privileged Account Security solution, a full life-cycle solution for managing the most privileged accounts in the enterprise, enables organizations to secure, provision, manage, control and monitor all activities associated with all types of Privileged Identities such as administrator on a Windows server, Root on a UNIX server, Cisco Enable on a Cisco device, as well as embedded passwords found in applications and scripts.

Privileged passwords and SSH Keys, as well as the audit information associated with using them, must be protected according to the highest security standards. The CyberArk Privileged Account Security solution utilizes the Patented Digital Vault™, certified as highly secure by independent security evaluators (such as ICSA Labs). CyberArk's Digital Vault is the heart of the Privileged Account Security solution and was designed to meet the highest security requirements for the "keys to the kingdom". The Digital Vault provides numerous underlying security capabilities for authentication, encryption, tamper-proof audit and data protection.

The CyberArk Privileged Account Security solution includes the following products:

- **Enterprise Password Vault**® – CyberArk's award winning Enterprise Password Vault (EPV) enables organizations to secure, manage, automatically change and log all activities associated with all types of Privileged Passwords and SSH Keys.
- **Application Identity Manager**™ – CyberArk's market leading Application Identity Manager (AIM) provides the only Application Identity Management solution to fully address the challenges of hard-coded App2App credentials and encryption keys. The solution eliminates the need to store App2App credentials in applications, scripts or configuration files, and allows these highly-sensitive passwords to be centrally stored, audited and managed within CyberArk's patented Digital Vault.
- **Privileged Session Manager**® – The Privileged Session Manager (PSM) enables organizations to control and monitor privileged accesses to sensitive systems and devices. PSM provides privileged session recording with DVR-like playback and text recording, as well as secure remote access to sensitive systems using privileged single sign-on, and without divulging the used credentials to the end users.
- **Privileged Session Manager PSM SSH Proxy** - The PSM SSH Proxy (PSMP) preserves the benefits of PSM, such as isolation, control and monitoring, whilst enabling users to connect transparently to target UNIX systems from their own workstation without interrupting their native workflow. The PSMP records all activities that occur during privileged sessions in a compact format in the Vault server, where they can be accessed by authorized auditors. The PSMP also provides privileged Single Sign-On capabilities and allows users to connect to target devices without being exposed to the privileged connection password.
- **On-Demand Privileges Manager** – The On-Demand Privileges Manager (OPM) provides a comprehensive solution that empowers IT and enables complete visibility and control of super users and privileged accounts across the enterprise. Using the OPM, the complete Privileged Account Security solution solution enables centralized management and auditing from a unified product to all aspects of privileged account management.

- **SSH Key Manager** – The SSH Key Manager addresses the challenges that arise during authentication to target machines with SSH Keys, and helps organizations meet audit requirements by simplifying and automating SSH Keys management. SSH Keys are stored and protected in the Vault under strict policy and access control, similar to that of passwords, and you can determine how users access and use them, by defining access workflows. The SSH Key Manager can periodically rotate the SSH Keys that are stored in the Vault, and make sure the private key protected in the Vault is always synchronized with the public keys spread over target systems. For more information about the provisioning and managing SSH Keys in the Privileged Account Security solution, refer to the SSH Key Manager Implementation Guide.

Solution Benefits

With CyberArk's Privileged Account Security solution, enterprises can easily:

- **Set the main policy rules that define how you manage accounts in your organization using the Master Policy.** The Master Policy offers a centralized overview of the security and compliance policy of privileged accounts and SSH Keys in your organization while allowing you to configure compliance driven rules that you define as the baseline for your enterprise.
- **Manage and Protect all Privileged Accounts and SSH Keys.** Utilize a secure Digital Vault in order to store, protect, manage and control access to Privileged Accounts and SSH Keys at a centralized point using a robust policy management engine. CyberArk's patented Vaulting Technology® software utilizes a fully integrated model of critical security layers, interwoven to meet the highest security needs.
- **Control Access to Privileged Accounts.** The Privileged Account Security solution offers a simple access control interface that easily pinpoints who is entitled to use privileged accounts and SSH Keys and initiate a privileged session, when and why.
- **Initiate and Monitor Privileged Sessions.** As a central control point, the Privileged Account Security solution also provides privileged single sign-on for initiating privileged sessions, as well as recording any activities that occurred during these sessions. PIM utilizes the Digital Vault as a tamper-proof secure storage for these session recordings.
- **Manage application and service credentials.** PIM provides sophisticated and transparent solutions for securing and managing critical applications as well as Application Server accounts, and eliminating the use of hard-coded and embedded passwords, making them invisible to developers and support staff.
- **Comply with audit and regulatory requirements.** The Privileged Account Security solution provides an easy way to create audit reports required by Sarbanes-Oxley, PCI and more. It allows enterprises to enforce corporate security policies to ensure compliance with regulatory needs and security best practices related to access and usage of privileged accounts and SSH Keys for both human and application (unattended) access.
- **Streamline management of Privileged Accounts.** The Privileged Account Security solution eliminates manual administration and overhead by providing instant and automatic changing of passwords for thousands of network devices and applications, including scripts and parameter files. Its high level of automation ensures highly reliable and uninterrupted service with minimal administrator overhead and increased productivity.

- **Seamlessly integrate with enterprise systems.** With an industry leading performance, scalability and robustness, PIM can protect and manage up to hundreds of thousands of passwords and SSH Keys across a highly heterogeneous IT environment, with complex and distributed network architectures. PIM can leverage existing enterprise infrastructure and integrate with corporate core systems
- **Easily set up and deploy.** PIM ensures quick deployment and implementation proven in over 400 enterprise customers, providing immediate ROI by improving IT productivity.

Vaulting Technology®

The Privileged Account Security solution architecture is based on CyberArk's Vaulting Technology® software. CyberArk discovered that by splitting the server interfaces from the storage engine, it can remove many of today's technology barriers associated with network security. The Vaulting Technology® software creates a Single Data Access Channel, which significantly improves security and makes it possible to build 10 layers of security in a unified solution.

Security Layers

The Privileged Account Security solution ensures the security of your organization's sensitive data using multiple security concepts, some of which are detailed briefly below:

Firewall & Code-Data Isolation

The Vault must run on a dedicated server, eliminating security holes in third party products. This is enforced by the CyberArk firewall, which doesn't let any communication into the server or out of it other than its own authenticated protocol – the Vault protocol. No other component is able to communicate with the outside world, except for the Storage Engine. The fact that the Vault's code is the only code that runs on the dedicated server assures a sterile environment and total control over the server by the security system.

VPN

The VPN encrypts every transmission (i.e. transactions and data) over the network. About 95% of the encryption processes occur on the client side, thus offloading the Vault and allowing higher throughput.

Authentication

Every access to the Vault must be authenticated. The Privileged Account Security solution uses a strong two-way authentication protocol. Authentication is based on passwords, PKI digital certificates, RSA SecurID tokens, RADIUS protocol, USB tokens, or Windows authentication. Taking the latter approach requires no additional authentication to be made by the end-user. The Privileged Account Security solution also supports third-party authentication that can be integrated into the organization's existing authentication server.

Access Control

The Privileged Account Security solution provides a built-in access control mechanism. Users are totally unaware of passwords or information that is not intended for their use. Users can be permitted to read, write, delete, or administer data according to the access control rules.

Password and File Encryption

Every password and file stored on the Vault is encrypted, using an encryption infrastructure that is totally hidden from the end user. This means that neither users nor administrators need to concern themselves with any key management issues.

The Vault assigns a unique symmetric encryption key to every version of every password or file stored in it. These encryption keys are securely delivered only to authenticated users that have appropriate access control rights. This enables the administrator to grant and deny access to passwords and files without the need to re-encrypt them. Users are never exposed to extraneous encryption keys and cannot decrypt passwords or files once their permissions are removed.

This unique key management also provides the means for the client-side VPN and the encrypted backups.

Visual Security

The Vault's Visual Security is the first and only technology that lets Users see activities carried out in their Safes by other Users. Real-time monitoring of who is logged on to the Safe and the information they have retrieved enables Users to track passwords and files in the Vault. Other Visual Security features inform Users whenever activity occurs in the Vault, and mark passwords and files so that those that have been accessed by other Users are noticeable immediately.

Manual Security

Manual security enables Users to define access to Safes that contain highly sensitive information so that users require manual confirmation from one or more Safe Supervisors before they can access privileged accounts. In this way, authorized users can permit or deny a request for access to a Safe or accounts by other users, and retain complete control over their information. Authorized users can confirm requests from mobile devices regardless of their physical location, enabling continuous workflows and preventing loss of productivity.

Geographical Security

The Vault uses Geographical security to restrict Users' logon areas. That means that Users can be permitted to log on only from certain areas of the network, or from a specific terminal.

Ready-to-Use Security

The Vault is a plug-and-play, ready-to-use product that implements its security mechanisms immediately after installation. It works with any network, and an unlimited number of Users.

How the Vault Protects your Passwords

Passwords that are stored in the Safe are protected in a variety of ways:

- **Password** - The Vault cannot be entered without a password and/or key.
- **Timing restrictions** - You can limit the times during which the Vaults/Safes can be opened (e.g., 8 a.m. to 5 p.m.).
- **Protected network area** - You can determine the locations on the network from which your Vault is accessed. This process is called defining a Private Network Area. For instance, an employee at an international company can set a Private Network Area so that their User account is only available from the Boston branch where he resides.
- **Access control** - You can define the level of access to a Safe for other Users. For instance, you can authorize Users to work with files but not to delete them.
- **Auditing** - Each time files are accessed for any purpose, the activity is written in the Vault activity log. This enables you to track all file activity and benefit from detailed auditing facilities.
- **Version control** - The CyberArk Vault tracks versions of the passwords and files it stores. Every time a password or file is updated, a new version is created. This means that if the most recent version is corrupted, previous versions are still available. In addition, you can undelete passwords and files that have been previously deleted.
- **Dual control** - Users may need to receive permission from other Users in order to open a Safe. For example, before another User can access a Safe they may need to request your permission and request confirmation.
- **Activity Logs** - The CyberArk Vault keeps records of all activities that take place inside it. An alert appears each time there is illegal activity in the Vault. For instance, an alert is issued when an attempt is made to logon to the Vault without the correct password.

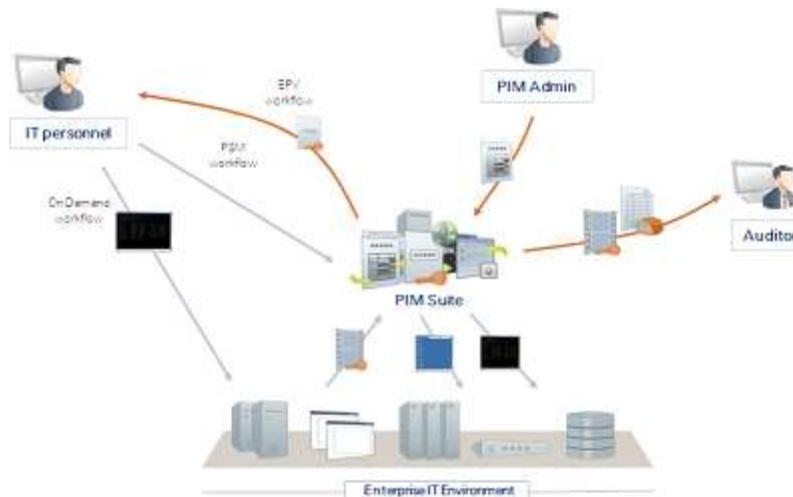
Privileged Account Security Solution Architecture

The Privileged Account Security solution provides a 'Safe Haven' within your enterprise where all your administrative passwords can be securely archived, transferred and shared by authorized users, such as IT staff, on-call administrators, and local administrators in remote locations.

The multiple security layers (including Firewall, VPN, Authentication, Access control, Encryption, and more) that are at the heart of the Privileged Account Security solution offer you the most secure solution available for storing and sharing passwords in an enterprise environment.

The Privileged Account Security solution is a plug-and-play solution which requires minimum effort to set up, and which can be fully operational in a very short period of time. It can be accessed and managed through a Windows Client, a Web interface, or a variety of APIs.

The following diagram shows the different components of the Privileged Account Security solution and how they interact.



Privileged Account Security Solution Architecture

The Privileged Account Security solution architecture consists of two major elements. One is the Storage Engine (also referred to as "the server" or simply "the Vault"), which holds the data and is responsible for securing the data at rest and ensuring authenticated and controlled access.

The second element is the interface (Windows interfaces, Web interfaces, and SDKs) that communicates with the Storage Engine on one hand and provides access to users and applications on the other. The Storage Engine and the interface communicate using CyberArk's secure protocol – the Vault protocol.

The Vault

The Vault is the most secure place in the network where sensitive data can be stored. The Vault is designed to be installed on a dedicated computer, for complete data isolation. It is packed with state-of-the-art security technology, and is already configured and ready-to-use upon installation. This means that the security system does not require any security expertise or complicated configuration to operate at peak capacity.

Constant access to your passwords is extremely important. However, when a Server fails to process requests, access to your passwords is prevented. The Vault can be installed as a high-availability cluster of servers which provide constant access to the passwords in the Vault. In this implementation, there is always at least one Server connected to the cluster that is on standby for when any other Server in the cluster fails to process requests.

The Vault is a full LDAP (Lightweight Directory Access Protocol) client, and can communicate transparently with LDAP-compliant directory servers to obtain User identification and security information. This enables automatic provisioning and creation of unique and individual users based upon the external group membership and attributes.

The Privileged Account Security Disaster Recovery Site ensures that your Vault is replicated to a Disaster Recovery Vault regularly, and can take over immediately when the Production Vault stops processes requests suddenly.

The Vault is installed with an interface that enables the Administrator to start and stop the Vault, and to monitor its operation.

The Password Vault Web Access Interface

The Password Vault Web Access (PWVA) is a fully featured web interface that provides a single console for requesting, accessing and managing privileged passwords throughout the enterprise by both end users and administrators with almost no training.

Automatically produced lists of frequently used passwords and recently used passwords for each user facilitate speedy access and usage. In addition, the Mobile PWVA enables users to access privileged accounts from mobile devices, enabling seamless connectivity and optimum workflows.

The PWVA's simple, intuitive wizard enables users to define new privileged passwords, while a powerful search mechanism enables you to find privileged passwords and sensitive files with minimum effort.

CyberArk's PWVA dashboard enables you to see an overview of activity in your Privileged Account Security solution, as well as statistics about all the activities that have taken place. The dashboard shows you a graphic representation of the passwords that have been managed, and links to specific information about users and passwords that require special attention.

PrivateArk Administrative Interfaces

PrivateArk Client

The PrivateArk Client is a regular Windows application that is used as the administrative client for the Privileged Account Security solution. It can be installed on any number of remote computers, and can access the Vault by any combination of LAN, WAN or the Internet.

In order to access the Vault, the Vault's Administrator User must define the User in the Vault. A Vault Network Area Administrator must then define the IP address or IP mask of the computer where the PrivateArk Client is installed in the Vault's Network Area.

In addition, the User must be authenticated by the Vault before being allowed access. The Privileged Account Security solution ensures a highly secured system of User authentication using a customizable combination of passwords, physical keys, and certificates.

After authentication, a User can work with the PrivateArk Client to set up a Vault hierarchy and create Safes and Users. Safe properties determine how each Safe will be accessed, and specific User properties determine the passwords that each User can access and the level of control that they have over these passwords. Users are also able to monitor and track their password activities, including who has accessed their information, when and from where.

Each command, request, file transfer and User configuration is encrypted before being transmitted between the Vault and the PrivateArk Client to ensure maximum protection for data at all times.

PrivateArk Web Client

Based on ActiveX technology, the PrivateArk Web browser interface provides the same interface as the Windows native client. The Web interface simplifies installation and distribution of the client in large organizations and permits easy access to the EPV from mobile computers.

Privileged Session Manager

The Privileged Session Manager (PSM) enables organizations to secure, control and monitor privileged access to network devices. Using the Vault technology, it manages access to privileged accounts at a centralized point and facilitates a control point to initiate privileged sessions. The PSM interface pinpoints users who are entitled to use privileged accounts and initiate a privileged session, when, and for what purpose. The PSM can record all activities that occur in the privileged session in a compact format and provide detailed session audits and DVR-like playback. Recordings are stored and protected in the Vault server and are accessible to authorized auditors.

PSM can be leveraged by enterprises to provide secure remote access to their sensitive network resources by third party vendors, without disclosing sensitive passwords or keys, and while recording the entire session. All of this can be done through HTTPS protocol without the need to open the enterprise firewall to native protocols such as SSH and RDP.

The PSM separates end users from target machines, and initiates privileged sessions without divulging passwords or keys, maintaining the highest level of security that is typical to all CyberArk components.

In addition, PSM can display a broad overview of all activity performed on every privileged account, without exception. All activities are fully monitored and meet strict auditing standards.

PSM is integrated transparently and seamlessly into existing enterprise infrastructures and does not require changes in users' workflow or password or key access procedures.

Privileged Session Manager SSH Proxy

The Privileged Session Manager SSH Proxy (PSMP) enables organizations to secure, control and monitor privileged access to network devices. Using the Vault technology, it manages access to privileged accounts at a centralized point and facilitates a control point to initiate privileged sessions. The PSMP pinpoints users who are entitled to use privileged accounts and initiate a privileged session, when, and for what purpose. The PSMP can record all activities that occur in the privileged session in a compact format. Text recordings are stored and protected in the Vault server and are accessible to authorized auditors. The PSMP also provides privileged Single Sign-On capabilities and allows users to connect to target devices without being exposed to the privileged connection password or key.

The PSMP can integrate with Microsoft's Active Directory (AD) to provision users transparently on UNIX systems, streamlining user management and reducing administrative overhead. In addition to automatic user provisioning, this CyberArk solution benefits from all standard CyberArk security and management features, including access control and auditing. Users have immediate access to UNIX machines, based on their AD permissions and groups, facilitating an uninterrupted workflow and maintaining productivity.

The PSMP separates end users from target machines, and initiates privileged sessions without divulging passwords or keys, maintaining the highest level of security that is typical to all CyberArk components. In addition, the PSMP can display a broad overview of all activity performed on every privileged account, without exception. All activities are fully monitored and meet strict auditing standards.

The Central Policy Manager

The Privileged Account Security solution provides a revolutionary breakthrough in password management with the CyberArk Central Policy Manager (CPM), which automatically enforces enterprise policy. This password management component can change passwords automatically on remote machines and store the new passwords in the EPV, with no human intervention, according to the organizational policy. It also enables organizations to verify passwords on remote machines, and reconcile them when necessary.

The CPM generates new random passwords and replaces existing passwords on remote machines. The new passwords are then stored in the EPV where they benefit from all accessibility and security features of the EPV.

Due to the Privileged Account Security solution distributed architecture, additional CPMs can be installed on different networks to manage passwords that are all stored in a single Vault. The Vault also supports shared configuration files for additional CPMs in high-availability implementations, and password management per Safe in

load-balancing implementations. This flexibility enables the Privileged Account Security solution to support complex distributed environments, for example where several data centers are managed by one Vault.

The On-Demand Privileges Manager

CyberArk's On-Demand Privileges Manager (OPM) enables organizations to secure, control and monitor privileged access to UNIX commands by using the Vault technology to allow end users to perform super-user tasks with their own personal account, whilst maintaining the least-privilege concept. It provides a comprehensive solution that empowers IT and enables complete visibility and control of super users and privileged accounts across the enterprise. Using the OPM, the complete Privileged Account Security solution enables centralized management and auditing from a unified product to all aspects of privileged account management.

The Password Upload Utility

The Password Upload utility uploads multiple password objects to the Privileged Account Security solution, making the Vault implementation process quicker and more automatic. This utility works by uploading passwords and their properties by bulk into the Vault from a pre-prepared file, creating the required environment, when necessary. It is run from a command line whenever a password upload is required.

SDK Interfaces

Application Password SDK

The Application Password SDK eliminates the need to store application passwords embedded in applications, scripts or configuration files, and allows these highly-sensitive passwords to be centrally stored, logged and managed within the Privileged Account Security solution. With this unique approach, organizations are able to comply with internal and regulatory compliance requirements of periodic password replacement, and monitor privileged access across all systems, databases and applications. The Application Password SDK provides a variety of APIs, including Java, .Net, COM, CLI and C/C++.

The Application Password Provider is a 'local server' that securely caches passwords after they have been retrieved from the Vault and provides immediate access to passwords, independent of network performance.

The Application Server Credential Provider securely and automatically manages application server credentials that are stored inside data source XML files. This prevents the need to perform any code changes to applications and can perform password replacement with no need to restart the Application Server, thus eliminating downtime and allowing business continuity.

Administrative APIs

The following CyberArk API enables organizations to automate Vault operations through scripts and programs.

Command Line Interface

CyberArk Vault's Command Line Interface, (PACLI) enables users to access the Privileged Account Security solution from any location using automatic scripts, in an extremely intuitive command line environment.

Limitations:

- PACLI v8.0 does not include commands that manage Master Policy rules, Exceptions, or Platforms.
- Commands for features that were moved from Safe level to Master Policy level (dual control, reason, exclusive passwords, auditing) have not yet been modified, but they will have no effect and will not raise an error.

Working with the Privileged Account Security Solution

This chapter shows you how to set up the Privileged Account Security solution and how to carry out basic password management activities so that you will be able to begin working with it.

This chapter comprises the following sections:

- Getting Started
 - Getting Started with the PrivateArk Administrative Client
 - Logging onto the Vault
- Setting up the Password Vault:
 - Creating and Managing Locations, Users, and Groups
 - Creating and Managing Safes and Owners
 - Transparent User Management
- Managing accounts through the Password Vault Web Access
 - The Master Policy
 - Managing Target/Service Account Platforms
 - Creating Accounts
 - Automatic and Manual Account Management
 - Account Access Workflows
 - Accessing Accounts through the Mobile PVWA
 - On-Demand Privileges for UNIX Environments
 - Monitoring Privileged Sessions
 - Auditing Accounts

Getting Started with the PrivateArk Administrative Client

This section describes how to create your Vault environment with the PrivateArk Administrative Client. It includes the following sections:

- Defining a Vault
- Creating Locations in a Vault hierarchy
- Creating Users
- Creating Groups

After this, you can create Safes and allocate owners. Then you can store passwords and files in Safes where users can access them.

Defining a Vault

In order to access a Vault, you need a connection between the Vault and the PrivateArk Client on your workstation. Once the connection is active, you can create a Vault to which authorized Users will have direct access.

To Create a Vault

1. From the **File** menu, select **New**, then **Vault**; the New Vault window appears.



2. Enter the name of the Vault and the workstation's IP address.
3. In the Default User Name edit box, type the name of the User whose name will appear by default in the Logon window.

- Click **Advanced** to display the Vault Properties - Authentication dialog box.



- Set the authentication parameters required by the Vault.
Refer to the Privileged Account Security Installation Guide for a description of authentication methods.
- Click the Connection tab to display the **Connection** dialog box and set the port parameters.



- In the General section, specify the port and the length of time in seconds that must pass after a request from a User to the Vault after which a timeout message will appear if there is no response.

8. In the Proxy or Firewall server section, specify whether you will connect to the Vault through a Proxy server or Firewall, or neither.
- If you select **I am using a Proxy server**, the following message box appears.



- Click **OK**, then click **Proxy Settings** to display the Proxy Settings dialog box and specify the type of proxy connection to use.



Note for the System Administrator: If your Proxy does not allow a 'keep alive' connection, select 'Use only HTTP 1.0'. This is not recommended, as the connection to the Vault will be noticeably slow.

- If you select **I am using a Firewall**, the following message box appears.



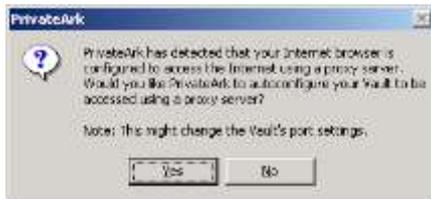
Click **OK**, then click **Firewall Settings** to display the Firewall Settings dialog box and specify the type of connection to use.



Note for the System Administrator: If your Firewall does not allow a 'keep alive' connection, select 'Use only HTTP 1.0'. This is not recommended, as the connection to the Vault will be noticeably slow.

9. If you are sharing Safes with a Gateway, in the Gateway settings section, enter the Gateway name or address, then click **OK** to set the advanced Vault properties, and return to the New Vault dialog box.

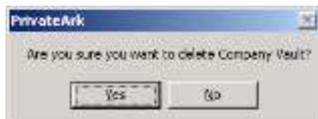
10. Click **OK** to create the new Vault; if your Internet Explorer is configured to access the Internet via a proxy server, the following window will appear.



11. Click **Yes** to enable the PrivateArk Client to autoconfigure the connection properties of the new Vault,
or,
Click **No** to create the Vault without autoconfiguring the connection properties.
The new Vault icon will appear in the PrivateArk Client working area.

To Delete a Vault

1. Select the Vault to delete.
2. From the **File** menu, select **Delete**; the following confirmation box appears.



3. Click **Yes** to delete the Vault icon, and to remove the connection between the PrivateArk Client and the Vault.

To Update a Vault's Properties

1. Select the Vault to update.
2. From the **File** menu, select **Properties**; the Vault Properties dialog box appears.
3. Update the Vault properties as necessary, then click **OK** to implement the changes.

Logging onto the Vault

In order to enter a Vault you must first logon. During the logon process, you authenticate to the Vault with a preconfigured authentication method.

Logging onto the PrivateArk Client

After installation you can logon with the default method, which is password authentication, but this can be changed. For more information about configuring authentication methods, refer to the Privileged Account Security Installation Guide.

If you log on with password authentication, the first time you logon use the logon credentials that the Vault administrator has provided for you. After you have logged onto the Vault, you can change your password to a more secure password.

To Log onto the Vault with Password Authentication

1. Double-click the Vault to enter; the logon window appears.
2. In the appropriate edit boxes, type your User name and password, then click **OK**; the Vault authenticates your User name and password and grants you access.

Note: It is recommended to change your password after logging on for the first time.

Changing a Password

Your password is created by the Vault administrator. However, you can change this password after logging on to specify a password that only you know. Although this password must be secure, make sure that you will be able to remember it for the next time you log on.

To Change your Password

1. From the **User** menu, select **Set Password**; the Set Password window appears.
2. Type in your new password, then click **OK**.

Locking your Vault

Protect your information when you take a coffee break

Other than when you retrieve files and return them, the Vault should remain locked. In particular, whenever you step away from your computer, the information in your Safe should not be left unprotected.

Each time you temporarily step away from your computer you can lock your User account. This protects your files completely, and prevents other Users accessing your account while you are away from your computer.

Note: The Vault will lock automatically after thirty minutes has elapsed without use, or after the period of time set by a Vault administrator.

To Lock a Vault

- From the **User** menu, select **Lock User Account**,
or,



Click **Lock** on the toolbar; your User account is locked and your files are protected.

To Unlock a Vault

- From the **User** menu, select **Unlock User Account**,
or,



Click the **Unlock** button on the toolbar.

- In the logon window, type your password, then click **OK**.

Logging off from the Vault

When you have finished working with files in the Vault, and you no longer need to keep your User Account open, you should log off from the Vault. This ensures that no one else accesses your Account.

When you log off from the Vault, open Safes are automatically closed and retrieved files are returned to the security of the Vault.

To Log Off from the Vault

- On the PrivateArk toolbar, click **Logoff**; all retrieved files are returned to the Safe, all open Safes are closed, and the Vault is closed.



Logging on through a Browser Interface

You can access a Vault through a browser interface using the PrivateArk Web Client.

- Using a browser, access the web site that contains a link to the Vault.
- Click on the link to the Vault to launch the Web Client; the PrivateArk Web Client components are downloaded to your computer. When the downloading process is complete, the License Agreement appears.
- Accept the terms of the License Agreement, then click Next to complete the procedure.
- If you need to restart your computer at the end of the download, a message box will appear telling you to do so.

If you do not need to restart your computer, the PrivateArk Client window will appear, displaying the Vaults that have been defined for you.
- Log on to the PrivateArk Web Client in the same way as you would log onto a local version of the PrivateArk Client.

Logging onto the PVWA

The PVWA offers several authentication options for logging on to the Vault:

- CyberArk
- Windows
- PKI
- RADIUS
- RSA SecurID
- Oracle SSO
- LDAP
- SAML

For Windows, PKI, SecurID, Oracle SSO, and LDAP, additional Vault or Radius authentication can be enforced for tighter security.

To Authenticate to the PVWA

1. In your browser, specify the following URL: `http://<host name>/passwordvault`

The PVWA displays the authentication methods you can use to log on.



or,

If the Administrator has configured a default authentication method, the relevant login page appears.

2. Select the authentication method that you will use to authenticate to the Vault; depending on the authentication method that you selected, the relevant logon page appears.

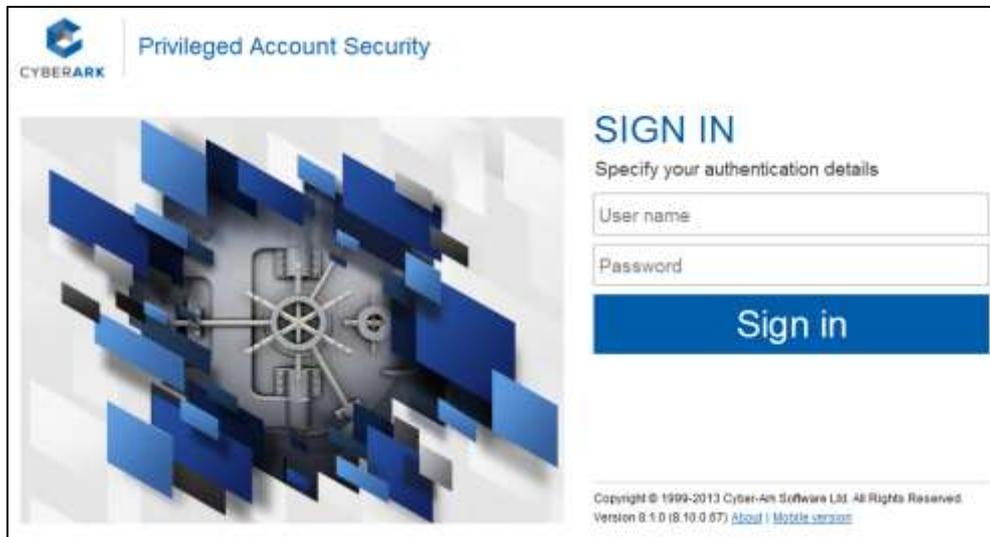
If the PVWA is configured to remember the last authentication method used from this machine, the page for that authentication method will be displayed.

Each logon procedure is described below.

CyberArk Authentication

You can log onto the Vault with a password that has already been defined for you in the Vault. After logging on the first time, it is recommended to change your password so that only you know what it is.

1. In the list of available authentication methods, click **CyberArk**; the CyberArk authentication page appears.



2. Type your CyberArk user name and password in the appropriate edit boxes, then click **Sign in**; the Vault authenticates your information and grants you access to the Vault.

To Change your CyberArk Password

Your CyberArk password is set by the Vault administrator when your user account is created. However, you can change this password after logging on to specify a password that only you know. Although this password must be secure, make sure that you will be able to remember it for the next time you log on.

1. In the toolbar, click **Customize**,
2. In the Change Password section, type in your current password.
3. Type in your new password and confirm it, then click **Save**.

LDAP Authentication

1. In the PVWA, in the list of available authentication methods, click **LDAP**; the LDAP authentication page appears.
2. Type the user's name and password as they are specified in the LDAP directory, then click **Sign in**; the Vault authenticates the user's information in the LDAP directory, then grants them access to the Vault.

To Change an Expired LDAP Password

LDAP passwords automatically expire after a predefined period of time, according to your organizational policy. You can change your expired LDAP password in the PVWA so that you can continue working seamlessly with privileged information that is stored in the Vault. This password is automatically updated in your organizational Active Directory.

Notes:

- Currently, only expired LDAP passwords stored in Active Directory can be changed in the PVWA. Expired LDAP passwords stored in other LDAP directories must be changed in the directories.
- An SSL connection to the LDAP directory is required. For more information, refer to *Configuring the Vault to Recognize LDAP Directories* in the Privileged Account Security Installation Guide.

To Change an Expired LDAP Password

1. When you try to log onto the PVWA with the expired password, a message appears informing you that your password has expired and the Change Password window appears.
2. In **Old Password**, specify your expired LDAP password.
3. In **New Password**, specify a new LDAP password.
4. In **Confirm New Password**, specify your new LDAP password again.

Your LDAP password is automatically updated and the PVWA authenticates your user.

Windows Authentication

This authentication option enables you to access a Vault without an additional logon procedure if you have already logged on to a Windows domain.

- In the list of available authentication methods, click **Windows**; the PWWA will check that you are logged on to the Windows domain and will grant you access to the Vault.

Users logging on from an Intranet zone will be logged on transparently, without requiring any additional logon information. However, users logging on from the Internet will be prompted for their Windows logon information.

Radius Authentication

You can log onto the Vault with Radius authentication, according to predefined authentication settings. After supplying your Vault username and logon information, if any more logon credentials are required, you will be prompted for them.

1. In the list of available authentication methods, click **RADIUS**.
2. Type the administrative user's Username and logon information in the appropriate edit boxes, then click **Sign in**; a secure channel is created between the client and the Vault through which this logon information is sent.
3. If the RADIUS server requires more information to authenticate the user to the Vault, a RADIUS Challenge window appears, prompting you for it.
4. Specify the additional logon details, then click **OK**; the RADIUS server authenticates you to the Vault.

RSA SecurID Authentication

1. In the list of available authentication methods, click **RSA Secure ID**; the RSA Web Agent logon page appears.
2. Specify the user's RSA username and password, then click **Sign in**; the RSA Web Agent authenticates the user and logs them onto the Vault.

PKI Authentication (User Certificate)

If your organization has a PKI (Public Key Infrastructure), you can log onto the Vault using your personal certificate.

Note: Make sure that your personal certificate is accessible. If your certificate is stored on an external hardware device, such as a Smart Card or a USB token, attach it to the computer before you try to log on.

- In the list of available authentication methods, click **pki**; depending on your browser and the security configurations, either of the following scenarios will happen:
 - The PWWA will automatically locate the user's certificate and log the user onto the Vault,
- or,
- A list of certificates will be displayed where the user can select a certificate and be logged on to the Vault

Oracle SSO Authentication

1. In the list of available authentication methods, click **Oracle SSO**; the Oracle Identity Management page appears.
2. Specify the user's Oracle SSO username and password, then click **OK**; the Oracle Identity Manager authenticates the user and logs them onto the Vault.

SAML Authentication

1. In the PVWA, in the list of available authentication methods, click **SAML**; the SAML authentication page appears.
2. Type your user's name and password as specified in the Identity Provider (IdP), then click **Sign in**; a secure channel is created between the IdP and the Vault through which this logon information is sent.
3. If the Idp is configured for multi-factor authentication, you will be required to specify additional logon details. The Idp will then pass the logon details to the PVWA in a secured channel.

Selecting a Specific Authentication Method via URL

- In the logon page, click the arrow to the left of SIGN IN; the list of available authentication methods appears. Select an authentication method.

Or,

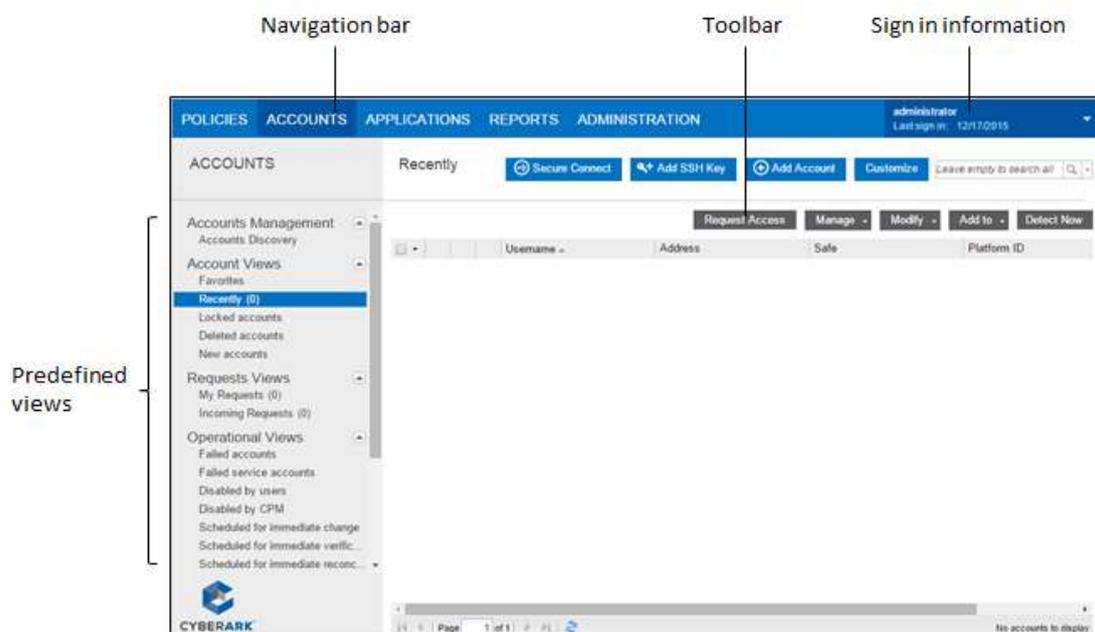
- Select an authentication method directly using the relevant URL.

Each authentication module that has been configured in the PVWA can be accessed directly through a URL. The following list shows the authentication method and the direct URL.

- Windows – <http://<host name>/passwordvault/auth/windows>
- PKI User Certificate – <http://<host name>/passwordvault/auth/pki>
- CyberArk – <http://<host name>/passwordvault/auth/cyberark>
- Oracle SSO – <http://<host name>/passwordvault/auth/oraclesso>
- RSA SecurID – <http://<host name>/passwordvault/auth/rsa>
- Radius – <http://<host name>/passwordvault/auth/radius>
- LDAP – <http://<host name>/passwordvault/auth/ldap>

The Main PVWA Window

After you have been successfully authenticated to the PVWA, the main PVWA window appears. The different parts of this window and their contents are described below:



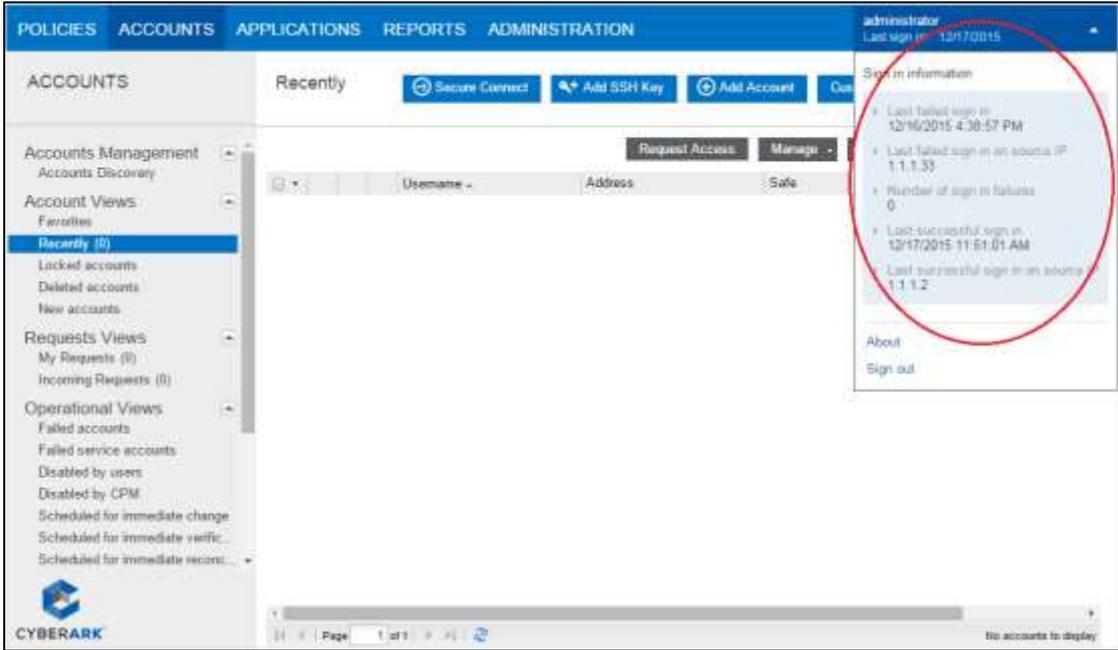
- Sign in information – This area displays the following information:
 - **User name** – The name of the user who signed into the PVWA.
 - **Sign in details** – A link that enables you to display information about the last time your current user signed in or attempted to sign in to PVWA.
 - **About link** – A link that enables you to display information about this version of the PVWA.
 - **Sign out link** – A link that enables you to sign out from the PVWA.
- Navigation list – This list displays the different PVWA pages that enable users to manage the privileged accounts stored in the Vault and configure the Privileged Account Security solution. The PVWA pages in this list are displayed according to the logged on user's authorizations.
- Toolbar – The toolbar displays the buttons that enable users to perform tasks in the PVWA. These buttons are displayed according to the logged on user's authorizations.
- Working Area – The working area displays the contents of the PVWA page selected in the Navigation list. For more information, refer to *Configuring the PVWA*, page 494.

Viewing Sign in Information

The sign in information displays details about the current users most recent sign ins and attempted sign ins to the PVWA from any network area. This enables you to check that you are the only person using this account.

These details include the following information:

- **Failed sign in** – You can see the date and time of the last failed sign in with your user, the IP address of the machine where your user tried to sign in, and the number of failed sign in attempts.
- **Successful sign in** – You can see when your user last signed into PVWA successfully and the IP address of the machine where your user signed in.



The screenshot displays the CyberArk Privileged Account Security interface. The top navigation bar includes 'POLICIES', 'ACCOUNTS', 'APPLICATIONS', 'REPORTS', and 'ADMINISTRATION'. The user 'administrator' is logged in, with a last sign in time of '12/17/2015'. The left sidebar shows the 'ACCOUNTS' section, with 'Recently' selected. The main content area shows a table with columns for 'Username', 'Address', and 'Safe'. A 'Sign in information' panel is open on the right, circled in red, displaying the following details:

- Last failed sign in: 12/16/2015 4:38:57 PM
- Last failed sign in on source IP: 1.1.1.33
- Number of sign in failures: 0
- Last successful sign in: 12/17/2015 11:51:01 AM
- Last successful sign in on source IP: 1.1.1.2

The bottom of the interface shows 'Page 1 of 1' and 'No accounts to display'.

Creating and Managing Locations, Users, and Groups

Managing Locations

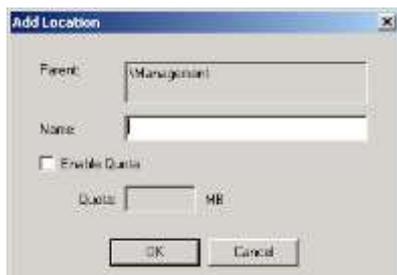
Locations represent the organizational hierarchy. You can organize Safes, Users and Groups by location, facilitating straightforward management. In addition, if you organize your Safes according to location, you can use the Safe filter to view only the Safes in a particular location rather than all of your Safes at once.

You can also set the maximum amount of disk quota that Users in a location can use. This ensures control over the size of the Safe.

This section describes how to manage locations, users and groups through the PrivateArk Client.

To Define a Location

1. From the **Tools** menu, select **Administrative Tools**, then **Locations**; the Locations window appears.
2. Select the position in the existing hierarchy for a new location, then click **Add**; the Add Location window appears.



3. Type the name of the new location.
4. To specify a quota, select **Enable Quota** and enter the size of the quota to allocate to the location, then click **OK**; the Manage Locations window appears and displays the new location.

You can now save Safes, Users and Groups in the new location.

To Move a Location

Locations can be moved in the hierarchy by dragging and dropping them. In this way, locations can be made sub-locations or moved to a higher level on the hierarchy after they have been created.

To Update a Location

After the location has been created, you can update its quota at any time.

1. In the Locations window, select the Location to update, then click **Update**; the Update Location window appears.
2. Modify the quota for the location, then click **OK**.

To Delete a Location

Locations that do not contain Safes, Users or Groups can be deleted.

1. In the Locations window, select the Location to delete, then click **Delete**; a confirmation message appears requiring you to confirm that you want to delete the location.
2. Click **OK**; the location is deleted from the hierarchy.

To Rename a Location

1. In the Locations window, select the location to rename, then click **Rename**.
2. Type the new name for the location, then click **OK**.

Managing Users

As a Vault administrator you are responsible for managing users in the Vault. Users can be created, deleted, updated, etc. These tasks are carried out through the Users and Groups window.

Users are divided into hierarchical levels that mirror the hierarchy in the office environment. Each department can have a User Manager who creates new Users and updates existing Users' properties. The User Managers can manage Users who are in the same hierarchical level and those in lower levels. In this way, User Managers have flexibility to control permissions of Users in other departments that are hierarchically beneath in the same way as their own Manager would.

For example, the Manager of the Engineering department is out of the office for one week. During that week, User permissions for members of that department need to be updated. Using the current hierarchy setup, any Department Manager above the engineering department can alter the permissions of the member of the Engineering department, and enable the Engineering team to continue working. Therefore, they don't have to wait for their own Manager to return to the office to update their permissions.

This feature makes User Management flexible, giving control to a wider group of authorized Users.

User authorizations determine which tasks users can perform in the Vault. Each user is only given the authorizations that they require and no others. This helps to achieve segregation of duties and provides a flexible methodology for controlling user management tasks in the Vault.

Depending on the permissions granted to them, Users of each level can manage other Users who are at the same level or lower than them, giving control and flexibility in user management.

Users who are listed in an LDAP-compliant enterprise directory can also be managed transparently by the Vault. They can be added as Safe members and given security attributes and authorizations depending on their location in the directory. For more information, refer to *Transparent User Management*, page 86.

Types of Users

The CyberArk license defines different types of users that can access the Vault through specific interfaces. The user type is defined when users are added to the Vault and when their properties are updated. All users are assigned a user type, including predefined users and those that are added manually or through an LDAP directory. In addition, Vault users that are used by CyberArk components to access the Vault are assigned a user type.

The following table describes the default user types and the interfaces that are allowed for each user type:

User Type	Description	Allowed Interfaces
EPVUser	EPV end user	PVWA, PrivateArk Client, PrivateArk Webclient, PACLI, NAPI, XAPI, PIMSU
PVWA	Password Vault Web Access component user	PVWA
PSM	Privileged Session Manager component user	PSM
PSMUser	PSM end user Note: This user can be used for PSM workflow only.	PSM, PVWA
PSMPServer	PSMP Server (PSM SSH Proxy)	PSMPAPP
CPM	Central Policy Manager component user	CPM
ENE	Event Notification Engine component user	ENE
AIMAccount	Application Account end user	Application Provider, PVToolkit
AppProvider	Application Password Provider component user	Application Provider
OPMProvider	OPM component user	OPM
OPMUser	OPM end user	PIMSU
PIMProvider	Application Password Provider and OPM component users	Application Provider, OPM
POCAdmin	POC Administrative user – used for v8.0 POC installations only.	PVWA, PrivateArk Client, PrivateArk Webclient, PACLI, PasswordUploadUtility

Each interface is specified by a unique Interface ID, listed in the table below:

Authorized Interface/Component	Interface ID
PrivateArk Client, Webclient	WINCLIENT
PrivateArk Client, Webclient (pre v4.6)	GUI
Password Vault Web Access	PVWA
Password Vault Web Access (application user and gateway user)	PVWAAApp
Central Policy Manager	CPM
Privileged Session Manager	PSM
Event Notification Engine	ENE
Application Password Provider	AppPrv
On-Demand Privileges Manager	PIMSU
PrivateArk CLI (PACLI)	PACLI
.NET API (v4.1 and below)	HTTPGW

You can generate a License Capacity report which enables you to see the maximum number of licenses for each user type or object, and the number of used licenses for each one. For more information about the License Capacity report, refer to *Reporting License Usage*, page 869.

Adding a User to a Vault

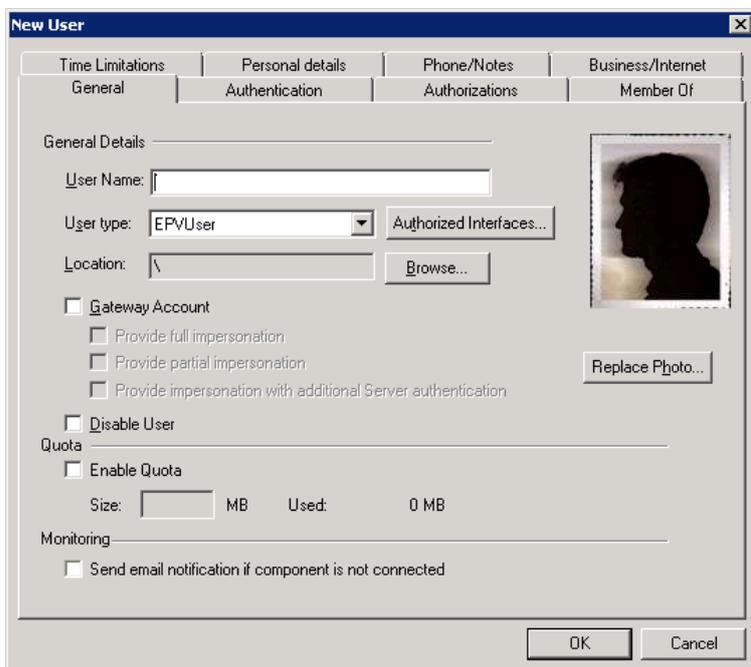
The Vault administrator is responsible for adding new users to the Vault. This process involves assigning a user name and password, defining permissions, and other managerial tasks.

To Add a New User

1. From the **Tools** menu, select **Administrative Tools** and then **Users and Groups**; the Users and Groups window appears.
2. In the hierarchy, select the Location where the user will be, then click **New**, then select **User**; the New User window appears.
3. In the different tabs of the New User window, fill in the information as described below. The **General** and the **Authentication** tabs are mandatory while the other tabs are optional.

General Tab

This tab enables you to create a new User account and to specify if this is a Gateway account. You can insert a picture of the User which appears each time you view any Visual Security displays.



Option	Defines ...
User Name	The name of the user. Note: You can specify up to 128 characters in the username. Make sure that the first 28 characters are unique to the username.
User type	The user type. This determines the interfaces that the user will be able to use to access the Vault. Click Authorized Interfaces to view and modify the interfaces that the selected user type is authorized to use. For more information, refer to <i>Updating User Types and Authorized Interfaces</i> , page 50.
Location	The user's Location inside the organization hierarchy.
Replace Photo	Enables you to select a photograph of the user. This is used in visual security.
Gateway Account	This user account is the Gateway account.
Disable User	The user account is temporarily inaccessible.
Quota	The amount of disk space allocated to the User and currently in use by this user.
Monitoring	Whether or not email notifications will be issued if the user account cannot connect to its authorized interfaces.

Authentication Tab

This tab determines the type of authentication method that the User will use to log onto the PrivateArk Client and access the Vault.

Option	Defines ...
Authentication method	The authentication method that the User will use to log onto the PrivateArk Client.
Require RSA SecurID authentication	The User is required to provide a SecurID passcode as well as the method specified above.
Password/Confirm	The User's password that is created for him to enable him to logon to the Vault the first time.
User Must Change Password ...	The User will change his password after he logs on the first time.
Password Never Expires	The PrivateArk Client will retain the User's password until he decides to change it.
Distinguished Name	Specify the User's distinguished name or select it from a list. (PKI authentication.)

Note: For more information about the various authentication methods, refer to the Privileged Account Security Installation Guide.

Authorizations Tab

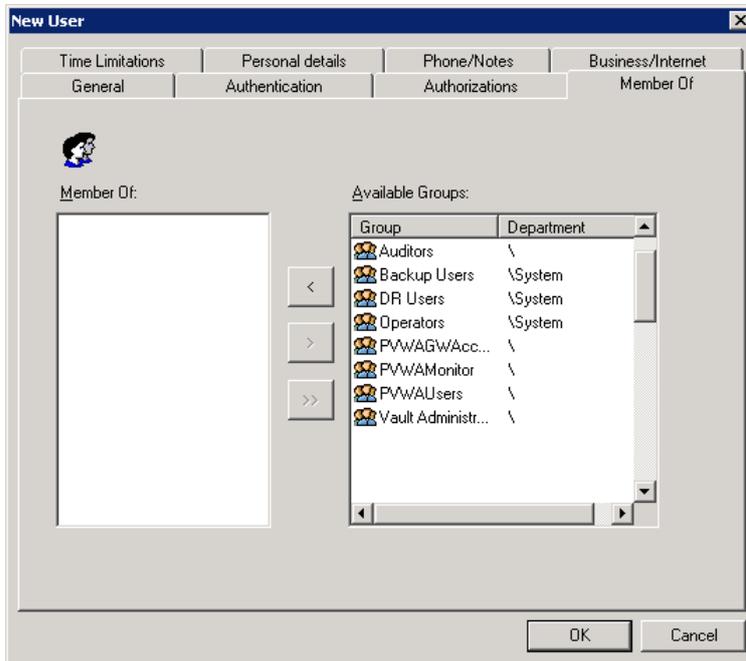
This tab determines the authorizations that the User will have in the Vault.



Section	Enables the user to ...
Add Safes	Add Safes in the Vault.
Audit Users	Track user activities in the Vault.
Add/Update Users	Add and update users, manage network areas, and manage Locations in the same level or lower on the Vault hierarchy.
Reset Users' Passwords	Reset user's passwords and set the "User Must Change Password at Next Logon" for users in the same level or lower on the Vault hierarchy.
Activate Users	Activate or deactivate trusted network areas for users in the same level or lower on the Vault hierarchy.
Add Network Areas	Add, update, and remove network areas in the Vault that specify where the Vault can be accessed.
Manage Directory Mapping	Add, update, and remove directory maps that manage users transparently in the Vault.
Manage Server File Categories	Add, update, and remove file categories in the Vault.
Backup All Safes	Run backup procedures.
Restore All Safes	Run restore procedures.

Member Of Tab

This tab is used to add a member to a group. Defining groups can make it easier to manage Safes since the permissions of more than one user can be modified in a single instance. Refer to *Managing Groups*, page 56 for more information.



Time Limitations Tab

This tab defines the time limitations to be applied to this User's account.



Section	Defines ...
History	The number of days that users' account activity records are stored before they can be deleted. This includes logon, logoff, user management, and other similar tasks. Note: If this parameter is set to zero, user activities in the Vault will not be written in an audit log.
Enable this User to logon at	Whether the User can log on to the Vault during specific hours or whenever he wishes.
Automatically expire User account on	This User account is accessible for either a set period of time or for an indefinite period of time.

Business/Internet Tab

This tab defines the user's contact information.

Section	Defines ...
Business address	The User's postal address.
Internet	E-mail addresses to which E-mails will be delivered.

Remaining Tabs

The following tabs include information that can be used later as reference for the Vault administrator.

- Personal details – the User's first and last names appear on the Owners list and in User Reports to facilitate easy identification.
- Phone/Notes

Updating Users

After a User Account has been created for a User, it can be updated at any time by the Vault administrator. This is also relevant for external Users, although their General Details cannot be modified in the PrivateArk Client, but only in the external directory which supplies their details.

In order to update user accounts, the Vault administrator requires the following authorizations:

- Audit Users
- Add/Update Users

In order to reset user's passwords and activate suspended users, the Vault administrator requires the following authorization:

- Audit Users
- Reset Users' Passwords
- Activate Users

To Update a User's Profile

1. In the Users and Groups window, select a user, then click **Update**; the Update Users window appears.
2. Make the relevant changes in the Update User Window (e.g., change password, update picture, etc.), then click **OK**.

To Rename a User

1. In the Users and Groups window, select the user's name to change, then click **Rename**.
2. Type the new name for the user, then click **OK**.

Deleting Users

When a User will not be using his User account any longer, you can delete the account from the Vault. This is important as it maintains the high level of security for the data in the Vault.

Note: Although you can delete external Users' accounts, the User must be deleted from the external directory to prevent a new User account being created for them when they next try to log on.

To Delete a User Account

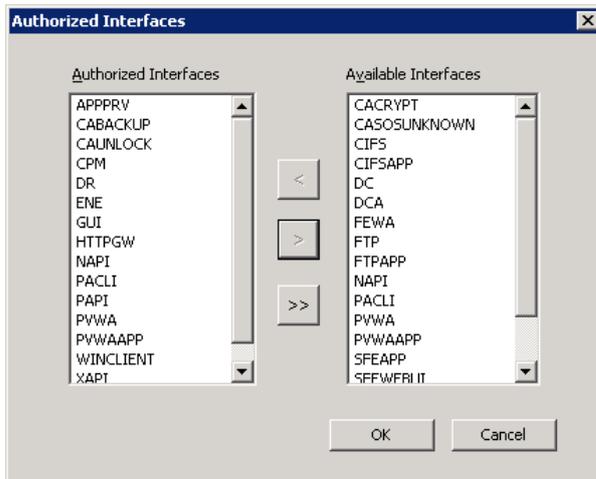
1. In the Users and Groups window, select a User, then click **Delete**; a confirmation box appears.
2. Click **Yes** to remove the User's account, and to prevent him from logging onto the PrivateArk Client.

Updating User Types and Authorized Interfaces

A user's type and authorized interfaces can be updated in the same way as all their other user account properties.

To Update a User's Type

1. In the Users and Groups window, select a user, then click **Update**; the Update Users window appears.
2. In the General tab, from the User type drop-down list, select the user type to apply to the user account.
3. Click **Authorized Interfaces**; the Authorized Interfaces window appears. This window displays all the interfaces that can be accessed by the selected user type, as defined in the license.



To Add Authorized Interfaces to the User Account

1. In the Available Interfaces list, select the authorized interface that the user will be able to use, then click the left-pointing arrow to move it over to the Authorized Interfaces list.
2. When the Authorized Interfaces list contains all the interfaces that the user will be able to access, click **OK**.

To Remove Authorized Interfaces from the User Account

1. In the Authorized Interfaces list, select the interface to disable for this user, then click the right-pointing arrow to move it to the Available Interfaces list.
2. When the Authorized Interfaces list contains the updated list of the interfaces that the user will be able to access, click **OK**.

Familiarization with Other Users in the Vault

In the Vault, users only see other users that they are familiar with. This ensures that users are not aware of users who are owners of other Safes. For example, a user from the IT department should not necessarily be aware that users from the Finance department are also using the Vault.

Familiarization is defined by at least one of the following:

1. The user has the **Audit Users** authorization in the Vault. This user is familiar with all the users in his location and sub-locations in the user hierarchy.
2. All users who share a Safe and have the **View Safe Members** authorization are familiar with each other. This means that they can all see each other in the users' hierarchy.
3. All users who are members of the same group are familiar with each other.

Network Areas

The Network Area consists of all locations from which a Vault can be accessed. This geographical security gives you additional control over the Vault.

Network Areas are defined by IP addresses. You can either define a Mask which gives you a wide range of addresses, or by Range which enables you to define specific addresses.

For example,

If you define an address of 1.1.1.1 with a **mask** of 24, the Network Area will be accessible from any computer whose IP address begins with 1.1.1. If the mask is declared as 16, the Network Area will be accessible from any computer whose IP address begins with 1.1. However, if the mask is declared as 32, the Network Area will be accessible only from the computer whose IP address is 1.1.1.1.

If you define an address with a **range**, you can specify the first IP address in the Network Area and the last IP address. So, you might specify that the Network Area is accessible from terminals whose IP addresses are anywhere in between 1.1.1.123 and 1.1.1.133.

Access to a network area can be restricted to specific users, with extra security measures such as permitted logon hours and violation counts. For more information, refer to *Trusted Network Areas*, page 54.

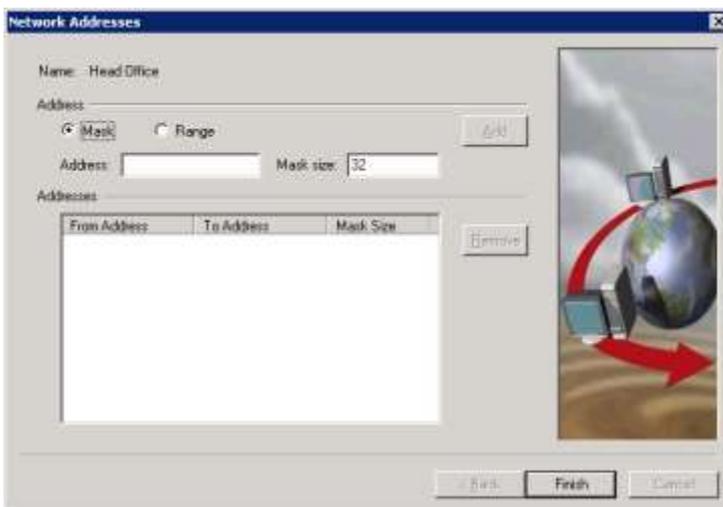
To Add a Network Area

1. From the **Tools** menu, select **Administrative Tools**, then **Network Area**; the Network Areas window appears.
2. In the Areas list, select the location under which the new Network Area location will be added, then click **New**; the New Network Area window appears.



3. In the Name edit box, type the name of the new Network Area.
4. Select the location of the area and its Security Level.

Note: The new Network Area location must have an equal or higher level of security than the existing Network Area.
5. To ensure that users will only be able to log on to web applications from the Network Area that is allocated to their user, select **Enforce Network Areas through Gateway**.
6. Click **Next**; the Network Addresses window appears.

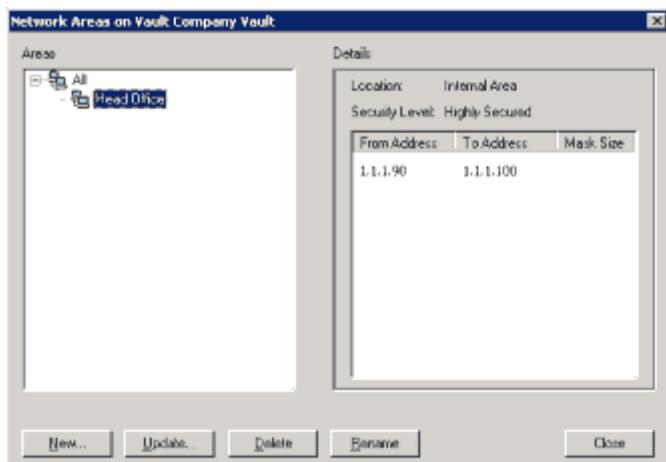


7. Select **Mask** then define the IP address of the Network Area and the Mask size, or,
Select **Range** then define the starting IP address and the final IP address of the Network Area.

- Click **Add**; the Network Addresses appear in the Addresses list.

Note: You can add more than one mask address or range in one Network Area.

- Click **Finish**; the Network Areas window appears listing the Network Area that you have just added with the details of the IP addresses that you specified for that Area.



To Update a Network Area

- In the Network Areas window, select a network area location, then click **Update**; the Update Network Areas window appears. This window displays the name of the Network Area, its location, and its security level.
- Click **Next**; the Network Addresses window appears, in which you can change or remove the IP address mask or range.
- Change the IP address as necessary, then click **Finish**; the modified Network Area and its details appears in the Network Areas window.

To Delete a Network Area

- In the Network Areas window, select a Network Area location, then click **Delete**; a warning box appears, prompting you for confirmation.
- Click **Yes** to delete the Network Area,
or,
Click **No** to retain the selected Network Area.

To Rename a Network Area

- In the Network Areas window, select a Network Area location, then click **Rename**; the name of the Network Area becomes highlighted and you can modify it.

Trusted Network Areas

Trusted Network Areas are the locations on the network from which a user can access the Vault. A Trusted Network Area prevents anyone from logging on to a user account from anywhere other than the specified location(s). The following example shows how a Trusted Network Area can help protect a user account.

A corporate executive has offices in New York, Boston, and London. The executive's Trusted Network Area includes only the Boston office, since he rarely spends time in New York or overseas in the London office and has no need to access the Vault from these other locations. In this scenario, he can only access the Vault from the Boston office. Therefore, by setting up a Trusted Network Area, an extra measure of security has been added.

Setting up a Trusted Network Area

After the Vault administrator has defined Network Areas, authorized users can set up Trusted Network Areas and manage them. These users must have the following authorization in the Vault:

- Add/Update Users
- Audit Users

Users who have the following authorization in the Vault can activate and deactivate trusted network areas:

- Activate Users

In addition, authentication specifications can be set to ensure that user accounts can only be accessed from a Trusted Network Area during certain hours. You can also set a maximum number of violations after which a Trusted Network Area is deactivated. Violations include logging on with an incorrect password or outside the permitted logon hours. The Trusted Network Area must be reactivated by the administrator before users can log on to it again.

To Set up a Trusted Network Area

1. From the **Tools** menu, select **Administrative Tools**, and then **Users and Groups**; the Users and Groups window appears.
2. Select the user whose Trusted Network Areas you will modify, then click **Trusted Net Areas**; the Trusted Net Areas window for the selected user will appear.
3.  Select the Network Area where you will add a Trusted Network Area, then click **Deactivate**; a red 'X' indicates that the network area has been deactivated.
4. Click **Add**; the Add Trusted Network Area window appears; it displays all the locations in the entire network area.
5. Select one or more location(s) to add to the Trusted Network Area.
6. In the Authentication tab, specify the hours during which access to the Vault through this Trusted Network Area will be permitted, and the number of violations after which the Trusted Network Area will be deactivated.
7. Click **OK**; the Add Trusted Network Area window closes and the Trusted Network Areas window displays the Trusted Network Areas where the user can access the Vault.

If you receive an error message, you might be trying to add a location that is already part of the Trusted Network Area. Before you can add a new area, any other areas that it's connected to must be removed or deactivated. Let's look at an example. Suppose that you want to add the Boston office area to a Trusted Network Area. Lisa's workstation is part of the Boston office, and her area is already included in the Trusted Network Area. Before you can add the Boston office to Lisa's Trusted Network Areas, you must first deactivate the Network Area for Lisa's workstation.

Updating a Trusted Network Area

There are two ways to update a Trusted Network Area. You can add a Network Area as described above, or you can activate, deactivate, and remove locations that are already included in the selected Trusted Network Area.

To Activate/Deactivate a Trusted Network Area Location

1. From the **Tools** menu, select **Administrative Tools**, and then **Users and Groups**; the Users and Groups window appears.
2. Select the user whose Trusted Network Areas you will modify, then click **Trusted Net Areas**; the Trusted Net Areas window for the selected user will appear.
3. Click on a Trusted Network Area location, then select **Activate** or **Deactivate**; access to the User Account is either permitted or not from the specified location.

To Remove a Location from a Trusted Network Area

- Select a location in the Trusted Network Area, then click **Remove**; the specified location is removed from the list of Trusted Network Areas.

Trusted Network Area Properties

You can set the hours that the Trusted Network Area can be used to access an account, and the number of illegal attempts that can be made by a User during logon from a Trusted Network Area before it is deactivated automatically.

To Set Trusted Network Area properties

1. Select a location in your Trusted Network Area, then click **Update**; the Update Trusted Network Area dialog box appears.
2. Set the Trusted Network Area properties, then click **OK**; the new authentication specifications are set for the User Account.

Managing Groups

A Group is a collection of Users who have the same authorizations. By defining a Group you can give all the Users in the Group the same authorizations collectively. Likewise, when you update the authorizations of a Group, the authorizations of each member of the Group are affected.

Users who are members of several Groups that own the same Safe, will either have the authorizations of the first group that was added as an Owner to a Safe, or a combination of the authorizations all the groups that they belong to, depending on how the Vault is configured. However, if the user is an independent Owner of the same Safe, his own authorizations will override those of the Group. For more information, refer to *Group Authorizations*, page 59.

Users who are listed in an LDAP-compliant enterprise directory can be added as group members transparently by the Vault, depending on their location in the directory. These users benefit from the same authorizations as group members created directly in the Vault. For more information, refer to *Transparent User Management*, page 86.

During PWWA installation, groups that are required for the PWWA are created automatically. For more information, refer to *The Environment in the Password Vault* in the Privileged Account Security Installation Guide.

To Create a New Group

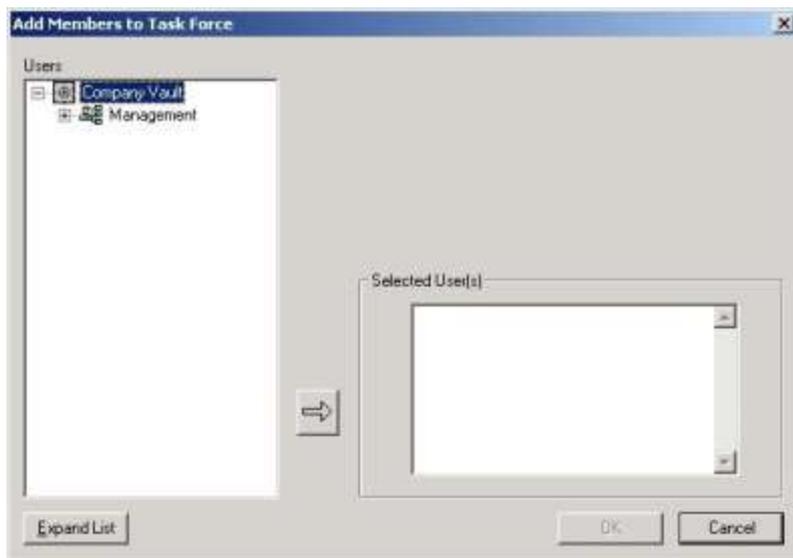
1. From the **Tools** menu, select **Administrative Tools**, and then **Users and Groups**, the Users and Groups window appears.
2. In the hierarchy, select the location where the new group will be created.
3. Click **New**, then select **Group**; the New Group window appears.

4. Enter the Group Name and Description of the group.

Note: You can specify a group name that contains up to 128 characters. Make sure that the first 28 characters are unique to the group name.
5. You can either add users to the group immediately, or click **OK** to create the group and add users later.

To Add a Member to a New Group

1. In the New Group window, click **Add**; the Add Members window appears.



2. Select the User to add, or click **Expand List** to display all users who share Safes with you ('known users'), then click the arrow to move him to the Group Members list. Repeat this process to add each member of the Group.

Note: In the Selected User(s) window, you can also type in the name of a User.

3. Click **OK**; the Group appears in the hierarchy list in the Users and Group window.

To Add a Member to an Existing Group

1. In the Users and Groups window, select the group to update, then click **Update**; the Update Group window appears.
2. Click **Add**, the Add Members to Group window appears.
3. Select the User to add to the group, then click the arrow to move him to the Group Members list. You can add as many Users as you wish.

Note: You can also type in a User Name.

4. Click **OK** to return to the Update Group window and display the members of the group.

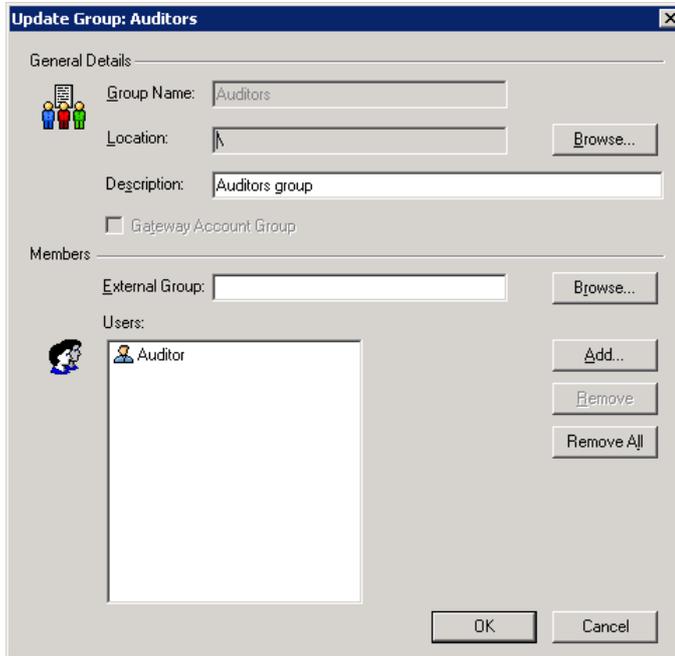
To Remove a Member of a Group

1. In the Users and Groups window, select the Group that the user belong to, then click **Update**; the Update Group window appears.
2. In the Members list, select the user to remove, then click **Remove**; a warning box appears prompting you for confirmation.
3. Click **Yes** to remove the User from the Members list.

Note: Click **Remove All** to remove all the Users from the list.

To Update the Properties of a Group

1. From the **Tools** menu, select **Administrative Tools**, and then **Users and Groups**, the Users and Groups window appears.
2. In the hierarchy, select the Group to update, then click **Update**; the Update Group window appears.



3. Add users to the Group or remove them, then click **OK**.

To Rename a Group

1. In the Users and Groups window, select the group name to change, then click **Rename**.
2. Type the new name for the user, then click **OK**.

To Delete a Group

1. In the Users and Groups window, select a group, then click **Delete**; a warning box appears prompting you for confirmation.
2. Click **Yes** to delete the Group,
or,
Click **No** to leave the Group intact.

Group Authorizations

Users who are members of several Groups that own the same Safe, will either have the authorizations of the first group that the user was added to, or a combination of the authorizations of all the groups that they belong to, depending on the 'GroupMergeAlgorithm' parameter in the DBParm.ini file, as follows:

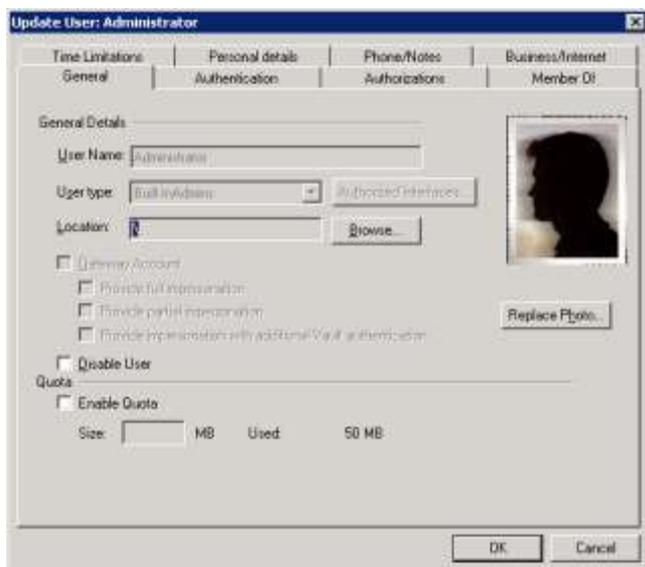
- **DenyOverrides** – users will benefit from a combination of all the authorizations granted to all the groups to which they belong.
- **FirstApplicable** – users will benefit from the authorizations that are specified in the first group that they were added to as a member.

Users that are also independent Owners of the same Safe will benefit from the authorizations specified in their individual user accounts, and not from those specified in the group definitions.

Predefined Users and Groups

The CyberArk Vault automatically creates several users and groups during installation and upgrade. These users are created for administrative tasks, and eliminate the need for specific users to be constantly available to carry out administrative purposes. Most of these users and groups become owners of every Safe in the Vault, both existing and new.

Although the users and groups are created automatically, all these user accounts except Master are disabled and therefore ineffective. In order to activate them, log on as the Master User and in the General tab of the User properties window, clear the 'Disable User' checkbox.



Next, in the Authentication tab, change the default passwords. These users have important permissions, and their passwords must be non-obvious and known only by authorized users.



Predefined Users

Note: To remove predefined users, refer to *Removing Predefined Users*, page 61. Before removing any predefined users, contact your CyberArk support representative.

An explanation of each predefined User follows:

- **Administrator** – The Administrator User appears on the highest level of the User hierarchy and has all the possible permissions. As such, he can create and manage other Users on any level on the Users hierarchy.
- **Auditor** – The Auditor user is a member of the Auditors group. His user appears at the top of the User hierarchy, enabling him to view all the Users in the Safe. The Auditor User can, therefore, produce reports of Safe activities and User activities. This enables him to keep track of activity in the Safe and User requirements.
- **Backup** – The Backup user is a member of the Backup Users group. He has the Backup Safe authorization, and therefore is able to backup all, several, or individual Safes.
- **Batch** – The Batch user is an internal user that cannot be logged onto. This user carries out internal tasks, such as automatically clearing expired user and Safe history.
- **DR** – The DR user is a member of the DR Users group and is specifically for use in Disaster Recovery. This user has the authorization to replicate the Safes in the production Vault to the Disaster Recovery Vault, keeping it continuously up-to-date.

- **Master** – The Master user has all the available Safe member authorizations, except Authorize password requests, and therefore has complete control over the entire system. This user is used to manage a full recovery when necessary. The Master user can only log in with the Master CD, which contains the Private Recovery Key.

In addition, the Master User enables the predefined Users immediately after installation and the initial network areas which enable other Users to begin working with the PrivateArk Client. This user cannot be removed from any Safe.

- **NotificationEngine** – The NotificationEngine user is installed with the Event Notification Engine (ENE). It retrieves information about activities that occur in Safes as well as contact details of recipients so that the ENE can send notifications.

This user is a member of the Notification Engines group.

- **Operator** – The Operator user is a member of the Operators group that has the Manage Safe authorization which enables him to update the Safe properties and carry out other administrative operations, such as compressing the Safe and changing the size of the Safe.

As the Operator user does not have any of the authorizations that would enable him to view the contents of a Safe, when he opens the Safe the Open Safe icon appears but not the Safe contents. In addition, he cannot view Safe logs or the Owners list.

- **POCAdmin** – The POCAdmin user is installed as part of the POC installation for Privileged Account Security solution v8.1. This user is for POC installations only and should not be used in other Privileged Account Security versions.

Removing Predefined Users

Note: Before removing any predefined users, contact your CyberArk support representative.

All predefined users, except the Master user, can be removed from Safes that they were added to automatically during Safe creation. The

PreDefinedUsersOwnerRemoval parameter in DBParm.ini determines whether or not all or none of the predefined users can be removed, or whether only the Auditor and Operator users can be removed.

By default, this parameter is set to 'None', meaning that immediately after installation no predefined users can be removed from Safes.

To Remove Predefined Users

1. In DBParm.ini, add the PreDefinedUsersOwnerRemoval parameter and specify the name of the predefined user to remove.

For example, PreDefinedUsersOwnerRemoval=Batch

2. Restart the Vault server.

For more information about the parameters in DBParm.ini, refer to the Privileged Account Security Reference Guide.

Predefined Groups

Note: To remove predefined groups, refer to *Removing Predefined Groups*, page 63. Before removing any predefined users or groups, contact your CyberArk support representative.

During installation or upgrade, the following predefined groups are added automatically to every Safe in the Vault, and the corresponding predefined user is added as a member. Users who are added to these groups immediately become owners of all the Safes, according to the Group's authorizations in the Safes. These groups can be removed from the Safes according to the Vault configuration.

Note: In Vaults that have been upgraded from previous versions, the predefined groups will only be added to Safes that are currently owned by the corresponding predefined users. These users will become members of a predefined group as well as remaining direct Safe members.

An explanation of each predefined group follows:

- **Vault Admins** – The Vault Admins group is a group of Vault administrators. This group can be added to Safes with all Safe member authorizations. This group is added automatically to the following Safes:
 - System Safe
 - Notification Engine Safe
 - All the Safes that are created during CPM installation and modified during CPM upgrade (<CPM User>, <CPM User>_workspace, and <CPM User>_info)
 - The configuration Safes that are created during PWWA installation (PWWAUserPrefs, PWWAConfig, PWWATicketingSystem, and VaultInternal)
 - The PSM Safe where the PSM user's password is stored, and Recording Safes where session recordings are stored.
- **Auditors** – The Auditors group has the **View audit** and **View Safe Members** authorizations, which enables members to view the contents of the Safe, the activity logs, and the Owners list. The predefined Auditor user is added automatically to this group.
- **Backup Users** – The Backup Users group has the Backup Safe authorization, which enables members to backup all, several, or individual Safes. It is recommended to use members of this group for backup operations and not grant this authorization to individual users. The predefined Backup user is added automatically to this group.
- **DR Users** – The DR Users group has the Backup Safe authorization and is used in Disaster Recovery. It is recommended to use members of this group for replication and not grant this authorization to individual users. The predefined DR user is added automatically to this group.
- **Notification Engines** – The Notification Engines group is a group of NotificationEngine users that are added during ENE installation, and which enable the ENE to send notifications about activities in the Safes. This group has the **View audit** and **View Safe Members** authorizations so that it can monitor activities in the Safe, but does not have access to any information.

- **Operators** – The Operators group has the **Manage Safe** authorization, which enables members to update the Safe properties and carry out other administrative operations, such as compressing the Safe and changing the size of the Safe. The predefined Operator user is added automatically to this group.
- **PVWAGWAccounts** – The PVWAGWAccounts group is a group of gateway accounts that is shared with all Safes that will be accessed through the PVWA.

Removing Predefined Groups

Note: Before removing any predefined groups, contact your CyberArk support representative.

All predefined groups can be removed from Safes that they were added to automatically during Safe creation. The **PreDefinedGroupsOwnerRemoval** parameter in DBParm.ini determines whether or not all or none of the predefined groups can be removed, or whether only the Auditors and Operators users can be removed.

By default, this parameter is set to 'None', meaning that immediately after installation no predefined groups can be removed from Safes.

To Remove A Predefined Group

1. In DBParm.ini, add the PreDefinedGroupsOwnerRemoval parameter and specify the name of the predefined user to remove.

For example, PreDefinedGroupsOwnerRemoval=Operators

2. Restart the Vault server.

For more information about the parameters in DBParm.ini, refer to the Privileged Account Security Reference Guide.

Inspecting User Activity

The Inspect User Activity window displays all your account activity. This includes the dates and times that logon and logoff was performed from your User account, file activity and alerts, and requests that you created or confirmed.

Let's look at the following scenarios:

One week ago, you used a file, but you no longer recall the file name or Safe location. By inspecting your account activity during the previous week, you can find the information that you need. In the User account activity window you will be able to see the names of the files that were used and the Safes where the files were stored. Inspecting your user account is an efficient way to search for a file.

You suspect that someone is using your password to access your User account. You can inspect your User account to check for illegal activity. If your account was used at 12 midnight when you are typically asleep then you know that your account was used by someone else.

To Inspect Activity in your User Account

- From the **Visual Security** menu, select **Inspect**, then **User Account Activity**.

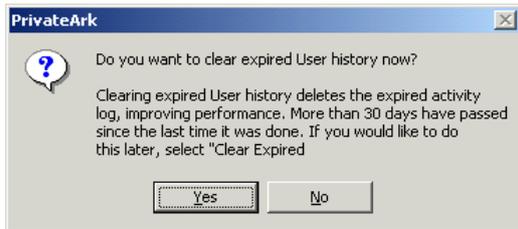
Clearing User History

Periodically, you need to clear the User account history. Only those records that have been held for longer than the time specified in the Safe Properties History window can be deleted.

To Clear User's Safe History

- From the **Tools** menu, select **Clear Expired History**, then **User Account**.

The PrivateArk Client reminds you to clear expired User history on a regular basis when you log on to the Vault, by displaying the following message box.



However, your System Administrator can make this action automatic so that expired history is cleared regularly without displaying this message window.

Creating and Managing Safes and Owners

The Password Vault gives you the flexibility to organize password objects according to individual organizational requirements and store them in different Safes. For example, an organization might decide to organize its password objects according to departments, and would then create a Safe for each department where all the password objects for that department would be stored.

By organizing passwords in different Safes, you can limit access to them. So, using the scenario above, only the administrator of the Windows passwords would have access to the Windows passwords Safe, while only the administrator of the Unix passwords would have access to the Unix passwords Safe.

In addition, only authorized users have access to the password object. As authorizations for each Safe member are given separately, some users will only have access to view a password object, while others will have access to modify its properties.

Throughout the entire password management procedure, the password object benefits from all the security and tracking features of the CyberArk Vault.

Adding and Managing Safes

Users can add Safes in the PWWA and modify their properties, as well as manage Safe members and their authorizations.

Adding Safes in the PVWA

Authorized users can add Safes through the PWWA. The Safes page displays a list of all the Safes they own, and where they can create new Safes. Users require the following authorization in the Vault:

- **Add Safes** – Enables the user to add Safes

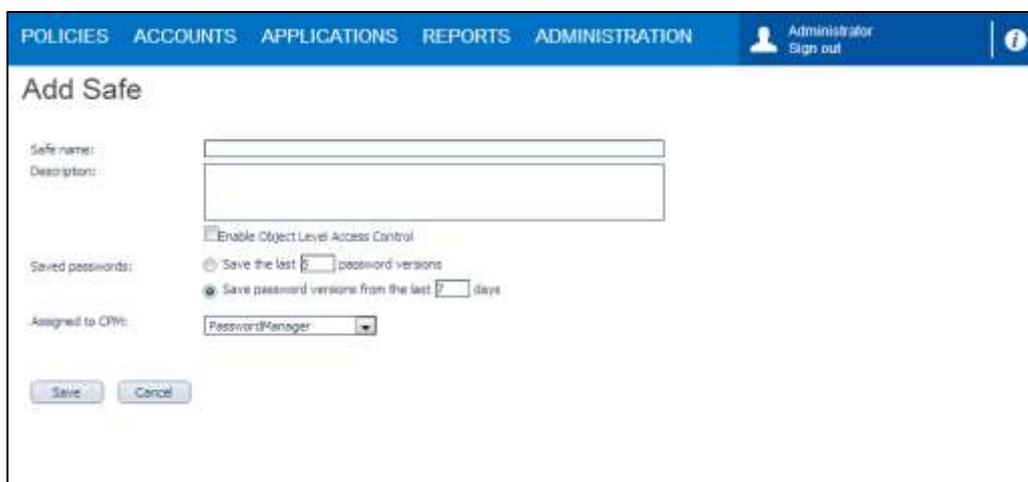
Users who do not have the Add Safe authorization can view the Safes page with either of the following authorizations:

- **Manage Safe** – Enables the user to view the Safes page and manage the properties of existing Safes.
- **Manage Safe Members** – Enables the user to view the Safes page and manage Safe members' authorizations.

Safes that are created in the PWWA are based on properties specified in a Safe template. For more information about creating Safe templates, refer to *Adding Safes*, page 570, in *Configuring the PVWA*.

To Add a New Safe

1. In POLICIES, click **Access Control (Safes)** to display the list of Safes.
2. Click **Add Safe**; the Add Safe page appears.



3. Specify the name of the Safe and a description, if required.
4. To control access to accounts in the Safe, regardless of user authorizations in the Safe, select **Enable Object Level Access Control**. For more information, refer to *Object Level Access Control*, page 76.

5. Specify password version management for the Safe, as follows:
 - **Save previous password versions** – Determines the number of password versions of every password that is stored in the Safe. These versions will be saved in the Safe indefinitely until they are replaced by a newer version.
 - **Save password versions for a time period** – Determines the number of days that password versions are saved in the Safe.

You can display the saved password versions in the Versions tab of the Account Details page. By default, the last five password versions are stored. For more information, refer to *Password Version Control*, page 302.

6. Click **Save**; the Safe will be created in the Vault and the Safe Details page appears.

Note: Reports Safes and PSM Recording Safes are created automatically with the following setting:

- **Auto-purge is enabled** – Files in this Safe will automatically be purged after the Object History Retention Period defined in the Safe properties.

In addition, these Safes cannot be managed by the CPM.

The Members tab displays the Owners of the Safe and their authorizations in the Safe. By default, all predefined users and groups are hidden. To display them, clear **Hide predefined users and groups**.

For more information about Safe members' authorizations, refer to *Adding and Managing Safe Members*, page 68.

Updating Safe Properties in the PVWA

Safes that are created in the PVWA are based on properties specified in a Safe Template. Users who have the Manage Safe permission in the Safe can modify some of the Safe properties that can be updated in the PVWA. Other properties can be changed in the PrivateArk Administrative Client.

For more information about the properties specified in Safe templates, refer to *Adding Safes*, page 570, in *Configuring the PVWA*.

1. In the Safes list, select the Safe to update, then click **Edit Safe**; the Edit Safe page appears.
2. Modify the Safe properties, then click **Save**; the updated Safe properties are saved.

Renaming a Safe

Users who have the Add Safes permission in the Vault can rename a Safe.

1. In the Safes list, select the Safe to rename, then click **Edit Safe**; the Edit Safe page appears.
2. Click **Show advanced section**, then specify the new Safe name.



3. Click **Save**; the updated Safe name is saved.

Deleting a Safe

If you are sure that the contents of a Safe are no longer needed and the Safe can be deleted, it can be deleted by users who have the **Manage Safes** permission in the Vault.

Note: You cannot recover a deleted Safe, so make sure that you will not need any passwords or files that are stored in it.

1. Display the Safe Details page for the Safe to delete, then click **Delete Safe**; the following message appears.



2. Click **OK** to delete the Safe and all its contents,
or,
Click **Cancel** to return to the Safe Details page without deleting the Safe.

Adding and Managing Safe Members

Users who have access to Safes are called Safe members. Each Safe member is given permissions in the Safe that enable them to perform tasks on accounts and files in the Safe. These permissions are given to each Safe member individually and give you flexibility to grant different permissions to different Users. Each Safe member can be given a unique set of permissions that is explicitly for their tasks and is not relevant for any other Safe member.

Below is a list of permissions that can be given to Safe members.

Permissions	Enables the Safe Member to ...
Access	Access accounts in the Safe, including the following tasks:
Use Accounts	<p>Use accounts in the Safe. Users who have this authorization can do the following:</p> <ul style="list-style-type: none"> Log onto a remote machine transparently through a PSM connection from the Accounts List by clicking the Connect with account icon. Log onto a remote machine transparently through a PSM connection from the Account Details page or the Versions tab by clicking the Connect button. <p>Note: To log onto remote machines transparently through a non-PSM connection, users require the 'Retrieve accounts' authorization as well.</p>
Retrieve accounts	<p>Retrieve and view accounts in the Safe.</p> <p>Users who have this authorization can do the following:</p> <ul style="list-style-type: none"> View the password in the Account Details page and the Versions tab by clicking the Show button in the password content panel. If the platform attached to the account doesn't permit users to view the password, the user requires the 'Manage Safe' authorization. Copy the password in the Account Details page by clicking the Copy button. If the platform attached to the account doesn't permit users to view the password, the user requires the 'Manage Safe' authorization. Display the password in the Accounts list by clicking the Show/Copy password icons. If the platform attached to the account doesn't permit users to view the password, the user requires the 'Manage Safe' authorization. Log onto a remote machine transparently through the PVWA. Platforms can be configured not to display the password value to end users, but only allow the transparent connection. Save files by clicking the Save As button in the Files List, File Details and File Versions pages. Open files that are stored in the Password Vault through the Files List, File Details and File Versions pages.
List accounts	<p>View Account lists.</p> <p>Users who have this authorization can do the following:</p> <ul style="list-style-type: none"> View the Accounts or Files list.

Permissions	Enables the Safe Member to ...
Account Management	Perform account management tasks, including the following tasks:
Add accounts	<p>Add accounts in the Safe.</p> <p>Users who are given this authorization in PVWA automatically receive Update password properties as well.</p> <ul style="list-style-type: none"> ▪ Add accounts in the Accounts List and Account Details page by clicking Add Account. ▪ Manage account groups and platforms in the CPM tab of the Account Details page by clicking Add New or Change.
Update password value	<p>Change password values as well as the contents of files.</p> <p>Users who have this authorization can do the following:</p> <ul style="list-style-type: none"> ▪ Change password values manually in the Account Details page by clicking the Change button. ▪ Undelete accounts in the Account Details page of the deleted account by clicking the Undelete button. This is only relevant during the file retention period. ▪ Manage account copies that are linked to accounts and are stored in the same Safe by clicking Add or Edit in the account usage tab. ▪ Upload files to the Password Vault by clicking the Upload button in the Files Details page.
Update password properties	<p>Update existing account properties. This does not include adding new accounts or updating password values.</p> <p>Users who have this authorization can do the following:</p> <ul style="list-style-type: none"> ▪ Update a selected account's properties in the Account Details page by clicking the Edit button. ▪ Manage logon and reconcile accounts in the CPM tab of the Account Details page with the Associate, Add New, and Clear buttons. ▪ Manage account groups and platforms in the CPM tab of the Account Details page. ▪ Save any account property values that are specified in the Remote connection details window for transparent connections when the user connects to a remote machine from the Accounts List, Account Details page, or the Versions tab.
Initiate CPM password management operations	<p>Initiate password management operations through the CPM, such as changing passwords, verifying, and reconciling passwords.</p> <p>Users who have this authorization can initiate CPM password management operations in the Accounts List and the Search results page, as well as the Account Details page by clicking Change, Verify, or Reconcile on the toolbar. In the Change Password window, the 'Manually selected password' option will be enabled if the user has the 'Determine next password value' authorization.</p>

Permissions	Enables the Safe Member to ...
Specify next password value	<p>Specify the password that will be used when the CPM changes the password value. Users who have this authorization can do the following:</p> <ul style="list-style-type: none"> ▪ Specify the next password that will be used as a password value in the Change Password and Immediate Password Change pages. <p>If the user does not have this authorization, the 'Manually selected password' option will be disabled and the CPM will set a new randomly generated password.</p> <p>Note: This authorization can only be given to users to have the Initiate CPM password management operations authorization.</p>
Rename accounts	<p>Rename existing accounts in the Safe in the Advanced section of the Edit Account page.</p>
Delete accounts	<p>Delete existing passwords in the Safe.</p> <p>Users who have this authorization can do the following:</p> <ul style="list-style-type: none"> ▪ Delete the account in the Account Details page by clicking the Delete button. ▪ Delete account copies that are linked to Windows accounts and are stored in the same Safe by clicking Delete in the password usage tab.
Unlock accounts	<p>Unlock accounts that are locked by other users.</p> <p>Users who have this authorization can do the following:</p> <ul style="list-style-type: none"> ▪ Unlock accounts that are locked by other users in the Account Details page by clicking Release on the toolbar, This is only relevant when the Enforce check-in/check-out exclusive access policy rule is configured. ▪ Unlock accounts that are locked by other users in the Advanced section of the Edit Account page by clicking Release. This is only relevant when the Enforce check-in/check-out exclusive access policy rule is configured . ▪ Unlock files that are locked by other users in the File Details page by clicking Unlock on the toolbar.
Safe Management	<p>Perform administrative tasks in the Safe , including the following:</p>
Manage Safe	<ul style="list-style-type: none"> ▪ Update Safe properties ▪ Recover the Safe ▪ Delete the Safe
Manage Safe members	<p>Add and remove Safe members, and update their authorizations in the Safe.</p> <p>Users who have this authorization can also do the following:</p> <ul style="list-style-type: none"> ▪ Modify permissions for accounts stored in Safes configured for Object Level Access Control in the Permissions tab of the Account Details page.
Backup Safe	<p>Create a backup of a Safe and its contents, and store in another location.</p>

Permissions	Enables the Safe Member to ...
Monitor	Monitor Safe members, and account and user activity in the Safe.
View audit log	View account and user activity in the Safe. Users who have this authorization can do the following: <ul style="list-style-type: none"> ▪ View the Activities tab for a selected account or file in the Account Details or File Details page. ▪ Generate the Safe Activities and Active/Non-active Safes reports in the PrivateArk Administrative Client.
View Safe Members	View Safe members' permissions. Users who have this authorization can also do the following: <ul style="list-style-type: none"> ▪ View the Permissions tab for accounts stored in Safes configured for Object Level Access Control in the Account Details page. ▪ Generate the Owners List and Entitlement reports in the PrivateArk Administrative Client.
Workflow	
Authorize password request	Give "confirmation" to a Safe members requesting permission to enter a Safe. Users also require the 'List accounts' authorization to see the Request details of the password requests waiting for their confirmation.
Access Safe without confirmation	Access the Safe without confirmation from authorized users. This overrides the Safe properties that specify that Safe members require confirmation to access the Safe.
Advanced	Perform folder related activities in the Safe, including the following tasks:
Create folders	Create folders in the Safe.
Delete folders	Delete folders from the Safe.
Move accounts/ folders	Move accounts and folders in the Safe to different folders and subfolders.

Adding Safe Members

Users who are authorized to Manage Safe Members in a Safe can add existing Vault users and groups, as well as users in external LDAP directories, as Safe members in the PVWA and specify Safe authorizations.

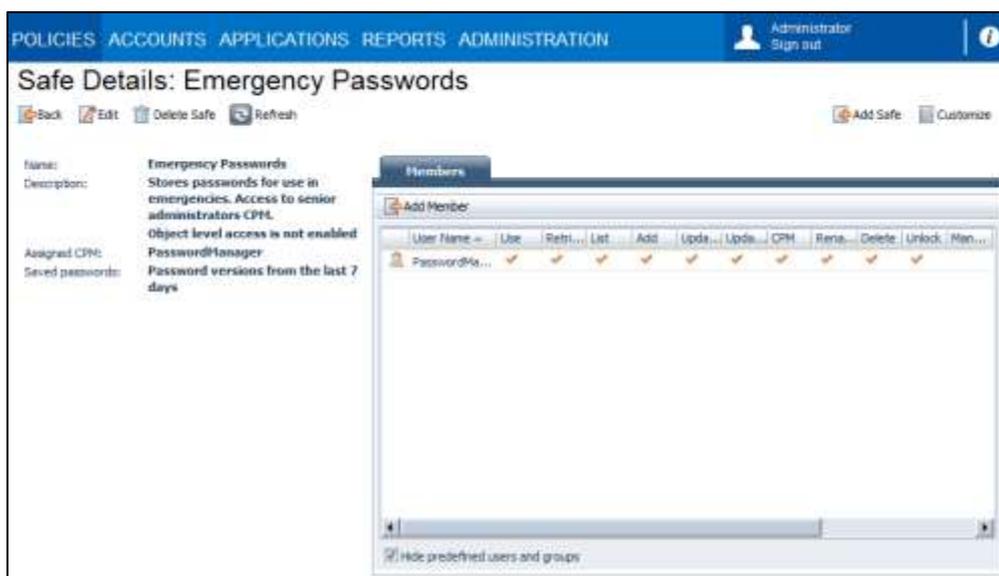
To Add Safe Members in the Current Vault

1. In the Safes list, select the Safe where you will add a Safe member, then click **Members**; the Safe Details page appears.
2. In the Members tab, click **Add Member**; the Add Safe Member window appears.

The default authorizations that will be given to the new Safe Member are selected. These authorizations can be configured in the Default Safe Authorizations in the Web Access Options in the System Configuration page. For more information, refer to *Configuring the System through PVWA*, page 918.

3. In the **Search** edit box, enter either part of the name of the user or group to add as a Safe member or the whole name. You can also leave the Search edit box empty to search for all users.
4. In the **Search In** drop-down box, select **Vault**, then click **Search**; a list of users and groups in the Vault whose names match the specified keyword is displayed.
5. Select the user or group to add as a Safe member, then select the authorizations that they will have in the Safe. Select the checkbox next to the title of the authorizations group to select all the authorizations in that group.
6. Click **Add**; the selected user or group is added and confirmation appears at the bottom of the screen.

- Click **Close**; the Safe Details page appears and displays the new Safe member in the Members list.



Adding Safe Members from LDAP

If the Vault is configured to support transparent user management, users that are configured in an LDAP directory can be added through the PWA.

- Display the Safe Details page for the Safe where you will add a Safe member.
- In the Members tab, click **Add Member**; the Add Safe Member window appears.
- In the Search In drop-down box, select the External Directory where the user that you will add as a Safe member is defined.
- In the **Search** edit box, enter either part of the name of the user or group to add as a Safe member or the whole name. You can also leave the Search edit box empty to search for all users.
- Click **Search**; a list of users in the specified external directory whose names, user ID or email match the keyword and the relevant Vault LDAP mapping rules is displayed.
- Select the user to add as a Safe member, then select the authorizations that they will have in the Safe. Select the checkbox next to the title of the authorizations group to select all the authorizations in that group.
- Click **Add**; the selected user is added and confirmation appears at the bottom of the screen.
- Click **Close**; the Safe Details page appears and displays the new Safe member in the Members list.

For more information about managing users in external directories, refer to *Transparent User Management*, page 86.

Updating Safe Member Authorizations

Users who are authorized to Manage Safe Members can update existing Safe Member authorizations.

1. In the Safe Details page, in the Members tab, click the name of the Safe member to update; the Update Safe Member window appears.

Update Safe Member

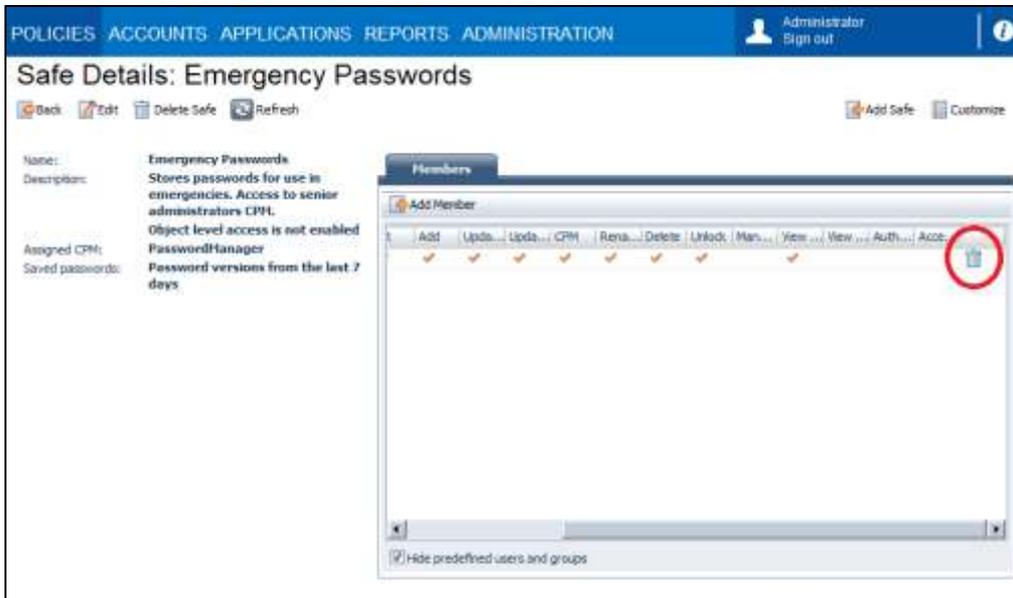
- Access
 - Use passwords
 - Retrieve passwords
 - List passwords
- Password Management
 - Add passwords (includes update properties)
 - Update password value
 - Update password properties
 - Initiate CPM password management operations
 - Specify next password value
 - Rename passwords
 - Delete passwords
 - Unlock passwords
- Safe Management
- Monitor
 - View audit
 - View Safe Members
- Workflow
- Advanced
- Membership expires on date:

Save Close

2. Update the Safe authorizations for this Safe member. Select the checkbox next to the title of the authorizations group to select all the authorizations in that group.
3. Click **Save**; the user's authorizations in the Safe are updated and the Safe Details page is displayed again.

Removing Safe Members

1. In the Safe Details page, in the Members tab, use the horizontal scroll bar to scroll to the end of the Safe Member authorizations; you can see the Remove Member icon.



2. Click the **Remove Member** icon in the row of the user to remove; a message appears prompting you for confirmation.



3. Click **OK** to remove the user from the list of members for this Safe,
or,
Click **Cancel** to return to the Safe Members list without removing the user from it.

Object Level Access Control

The Privileged Account Security solution provides granular access control for passwords and files that are stored in the Vault. Object level access enables you to control who can retrieve and use specific passwords and files in the Safe, regardless of Safe level member authorizations. For example, an external vendor or technician can be given retrieve or use authorizations for a specific password which he will be able to use without being aware of any other passwords or files in the Safe.

When a new password or file is added to a Safe, each Safe member will have their default permissions on that new object, as set in their Safe member authorizations. However, these authorizations can be changed granularly for individual passwords or files.

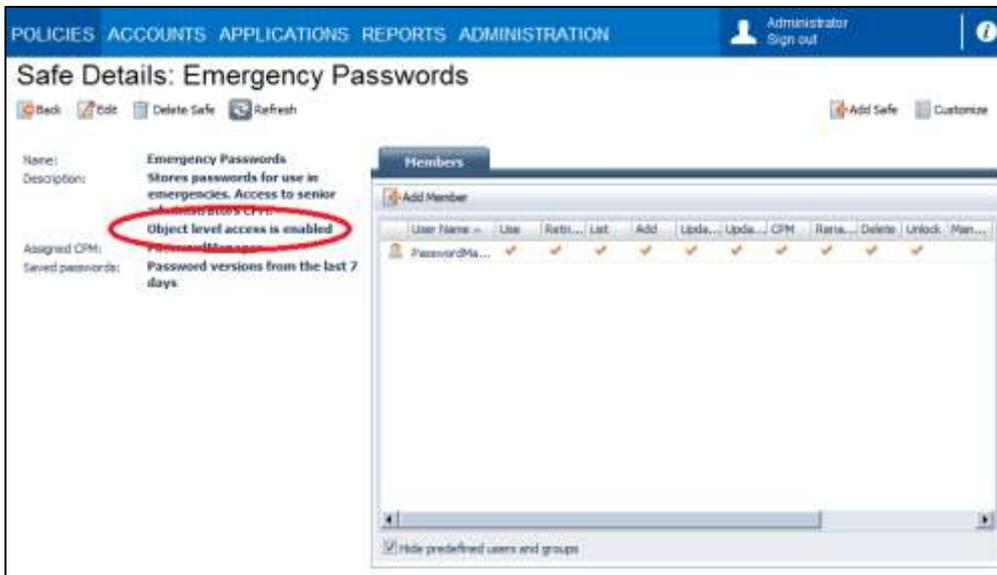
You can see a general summary of each user's access control and authorizations in the Entitlement report.

Configuring the Safe

Object level access control can be configured in the PWWA. It can be set either when the Safe is created or by updating an existing Safe's properties. Once enabled, object level access control cannot be disabled.

To Configure the Safe

1. Display the Safe Details page for the Safe to update, then click **Edit**; the Edit Safe page appears.
2. Select **Enable Object Level Access Control**, then click **Save**; object level access is enabled for this Safe, and the Safe Details page displays the Safe settings.



Configuring User Accounts

Any user who is a Safe member can be given object level access.

To Configure User Accounts

- Add all users who will access passwords or files in the Safe as Safe member of the Safe. The authorization that you select will affect access to objects in the Safe as follows:
 - If the user **has** the **Use accounts** or **Retrieve accounts** authorizations, you can remove these authorizations from individual passwords or files to prevent the user from accessing them.
 - If the user **does not have** either of the above authorizations, you can give them individually on specific passwords and files to enable the user to access them.

Viewing the Safe Members List

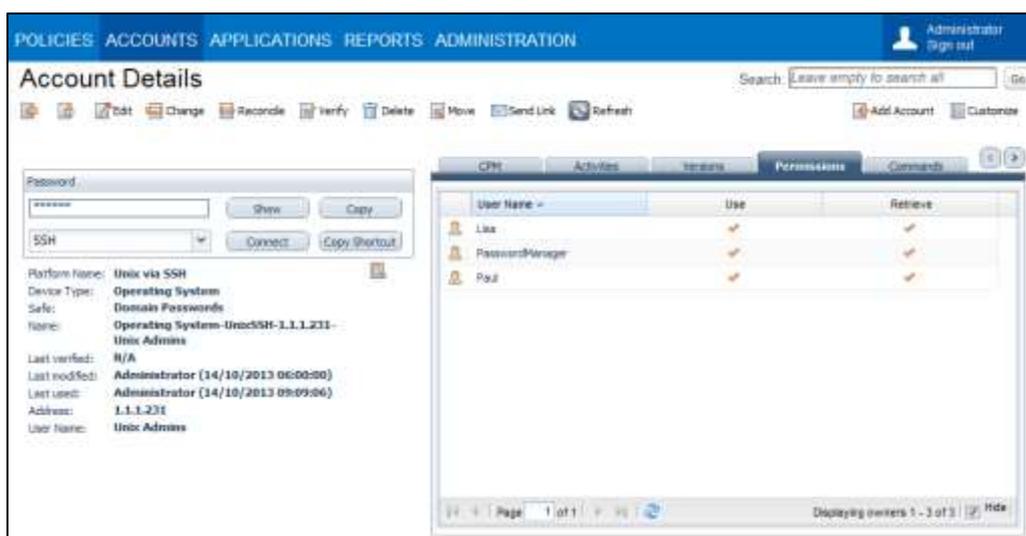
Authorized users can view a list of users who have permission to retrieve a selected account or file in the Object Properties window. Users require the following Safe member authorization in order to view the list of Safe members who are authorized to retrieve a specific account or file:

- View Safe Members

Users who do not have this authorization will not be able to see the Permissions tab in the Account Details window.

To View the Authorized Safe Members List

1. Display the Account Details window for the password for which you want to see who has access.
2. Click the **Permissions** tab; a list of all the Safe Members for this Safe is displayed. You can see which users have the 'Use passwords' authorization for the current account and which have the 'Retrieve passwords' authorization for it.



Managing Object Level Access Control

Authorized users can give use and retrieve permissions on individual passwords or files to Safe members who do not have retrieval permissions in the Safe. These users can also revoke retrieval permissions for specific users on individual passwords or files. Users require the following Safe member authorizations in order to manage Object Level Access Control:

- View Safe Members
- Manage Safe members
- One of the following:
 - Retrieve passwords authorizationor
 - **Use passwords** authorizationor
 - No 'Retrieve passwords' authorization or 'Use passwords' authorization, but has authorization to access the password or file.

Users who do not have all of the above authorizations will not be able to add or remove Safe members to the list of users who are authorized to use or retrieve the specified password or file.

To Manage Access to a Password or File

1. In the Permissions tab, click the name of the user to grant or deny access to the password; the Change Permissions window appears. This window enables you to change the user's access permissions for this password or file.



2. Change the permission, then click **OK**; the user's permission is changed and the current permission is displayed in the Authorized Safe member list.

Advanced Safe Management

Safes that are created in the PWVA are based on properties specified in a Safe Template. Users who have the Manage Safe permission in the Safe can modify some of the Safe properties that can be updated in the PWVA. This section describes the properties that can be changed in the PrivateArk Administrative Client. These tasks described in this section are for advanced management and configuration and are rarely used

Creating Password Safes

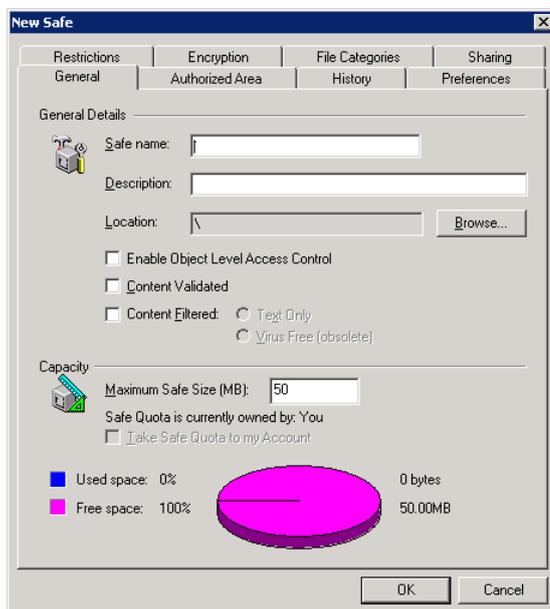
When you create a Safe, you determine how the files in it will be stored, accessed, and categorized. At any time after a Safe has been created, you can modify the Safe property settings.

Different configurations influence the way you can work with the Safe. For example, you can configure a Safe for Sharing, which mean that it can be accessed through a variety of applications through the Password Vault Web Access.

A new Safe is defined in the New Safe window. The first tab is mandatory, although every other tab contains default settings which you can accept as your own.

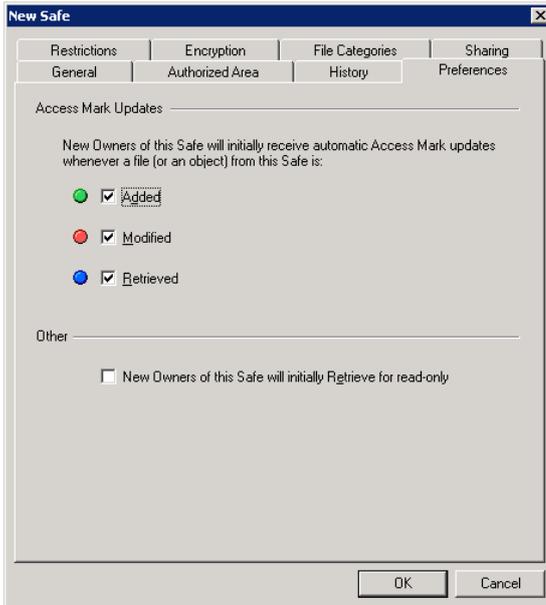
To Create a Password Safe

1. In the PrivateArk client, logon to a Vault.
2. From the **File** menu, select **New**, then **Safe**; the New Safe window appears.
3. In the General tab, specify the name of the Password Safe and any other relevant details.

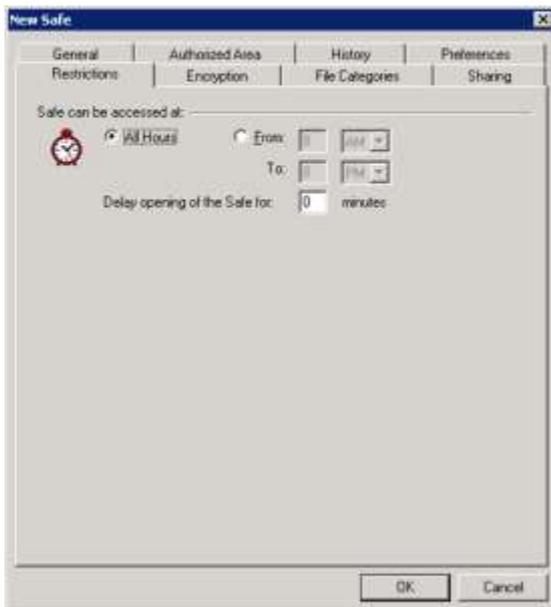


4. If the passwords in the Safe will be managed by the CPM, display the Preferences tab and make sure that **Retrieved** is selected.

This will enable the Safe Owners to track password activity at a glance, and specifically, will enable the CPM user to identify passwords that are marked as **OneTimePasswords** and have been added or retrieved.



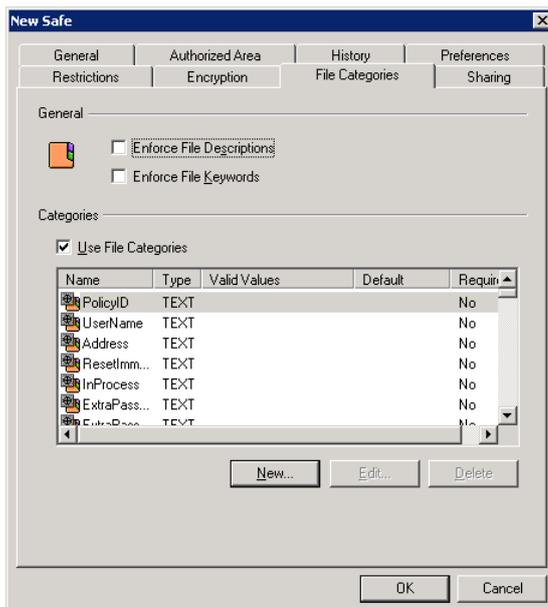
5. In the Restrictions tab, you can determine the hours during which a Safe can be accessed, by selecting one of the following:
 - All Hours – Safes can be accessed at any time.
 - From – Safes can only be accessed between the specified hours.



You can also determine whether or not there will be a delay between when a Safe is opened and when it can be accessed. Specify the length of the delay in minutes.

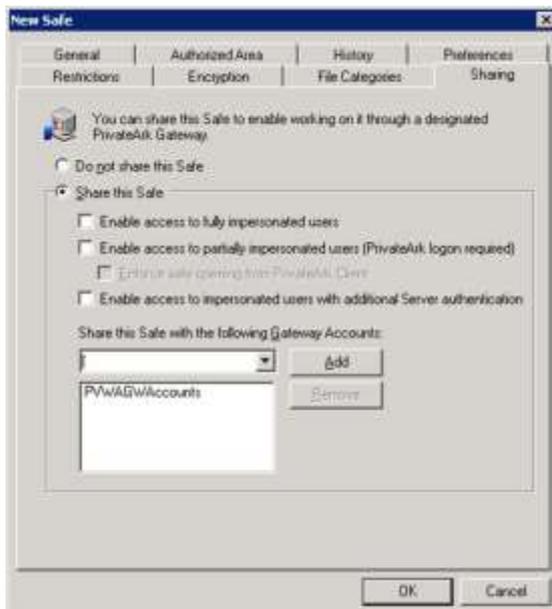
6. In the File Categories tab, check that **Use File Categories** is selected. This enables you to use password properties.

If the passwords in this Safe will be managed by the CPM, this option is required for the CPM user to identify passwords.



To create or add new account properties, refer to *Defining Custom Account Properties*, page 161.

7. In the Sharing tab, select **Share this Safe**, and then select one of the following options, depending on the type of authentication you specified during Password Vault Web Access installation:
 - Enable access to fully impersonated users
 - Enable access to impersonated users with additional Server authentication.
8. From the Gateway Account drop-down list, select the Gateway Account group that was created as part of the Password Vault installation, then click **Add**; the Gateway Account group name is added to the list of Accounts that the Safe is shared with.



9. In the other tabs, specify additional Safe properties.
10. Click **OK**; the new Safe is now ready to store passwords in it, and can also be accessed by authorized users through the Password Vault Web Access.

Updating Safe Properties

If you have the **Manage Safe** authorization, you can modify the properties of an existing Safe.

To Modify Safe Properties

- Open the Safe, then from the **Safe** menu, select **Properties**; the Properties for Safe window appears.

This window is similar to the New Safe window, except that the Safe name cannot be changed. You can modify all the properties in the different tabs as described above, except for those in the Encryption tab.

To Rename a Safe

1. Open the door of the Safe to rename, but remain in the Safe view.
2. From the **File** menu, select **Rename**; the name of the Safe appears within a white text box for you to change.
3. Type the new name of the Safe, then press **Enter**.

A Safe can be renamed even though other users might have files in the workspace. However, when the user logs on or off from the PrivateArk Client, a message box will appear to tell him that the files cannot be returned to the Safe, and to enable him to save the files in a new location.

Deleting a Safe

The Safe and its files can be deleted from a Vault. However, you can only do this after the version retention period has expired for all files.

To Delete a Safe

- Select the Safe to delete, then from the **File** menu, select **Delete**; the Safe and all its contents is deleted.

Note: You cannot recover a deleted Safe, so make sure that you will not need any passwords or files that are stored in it.

For more information about Safe ownership, refer to *Adding and Managing Safe Members*, page 68.

Opening the Safe

After you log onto the Vault, the Safes that you are authorized to access appear in the Working Area. You are a Safe Owner of these Safes.

Safe Owners are users who have the authority to enter the Safe and work with passwords and files in the Safe or make changes to the Safe itself. Authorizations may vary according to the settings of each Safe Owner. For a complete list of Safe Owner rights, refer to *Adding and Managing Safe Members*, page 68.

A Safe Filter box enables the user to filter ‘his’ Safes and display a group of Safes, rather than all of them at the same time. The filter box displays all the locations that the user has access to, in addition to locations that contain Safes which the user owns. By selecting a location, the user can view the Safes in that location and its sublocations.



To begin working with the files in the Safe, open the Safe and then enter to display the files inside.

To Open and Enter a Safe

1. Logon to a Vault. The available Safes are displayed in the Working Area.
2. Select a Safe, then from the **Safe** menu, select **Open and Step Into**,
or,
Double-click the Safe to open.

The contents of the Safe are displayed in the working area.

There are some advanced users who may wish to remain at the “Safe level”. By opening the Safe, but still remaining outside the Safe itself— you can update the Safe properties (e.g., size of the Safe) and perform other administrative tasks.

To Open a Safe without Displaying its Contents



- Click on a Safe, then from the **Safe** menu, select **Open**. The Safe is opened, but you remain outside the Safe itself.

You can configure your PrivateArk Client to open the Safe, but not to display the contents of the Safe.

- From the **Tools** menu, select **Options**; the Options window appears.
- In the **General** tab, clear 'Display Safe contents on double-click/Enter', then click **OK**.

To Display Safes by Location

The Safes you own are displayed automatically when you log onto the Vault. The Safe Filter enables you to display Safes that have been created under a specific location, and therefore reduces the number of Safes that appear on your screen.

- From the Safe Filter drop-down list, select the location where the Safes you want to display were created; the Safes created in that location appear in the Safe view.

Confirmation to Enter the Safe

You might not have the authority to open a particularly secure Safe until you receive "clearance" from one or more of the Safe's Owners. The Requests icons on the side of your screen will indicate when clearance has been requested, and when you have permission to enter the Safe.



Enter the Safe only at Specific Times

Another security feature prevents Safes from being opened except at certain times (e.g., 8 a.m. to 5 p.m.). If you try to enter at a time that has not been designated for access, you will receive a message that informs you that the Safe is unavailable.

Closing a Safe

When you close a Safe, all retrieved files that are in the PrivateArk Workspace are returned automatically to the Safe. It is important to close the Safe after you've finished working with your files to prevent them from being left outside the Vault where they are not secure. Files that are in use by another application cannot be returned until they are closed.

To Close a Safe



- From the **Safe** menu, select **Close**; all retrieved files are returned to the Safe and the Safe is closed.

If any files are in use by another application, a message box appears to inform you. The message box prompts you to retry returning individual files to the Safe, or to skip them and leave them out of the Safe.

- To return files that are in use, toggle to the open file, close it, then click **Retry**.
- To leave the files out of the Safe, click **Skip**.

Inspecting Safe Activity

Whenever necessary, you can view records of all Safe activity. This includes the activities of all Users or each individual User. The Safe activity window displays the names of the Users who have handled files, when, and for what purpose.

For instance, a manager at a large firm can view the activity of everyone who has opened a Safe or the Safe activity for an individual employee.

The Safe activity window can be displayed whether the Safe is opened or closed.

To Inspect Safe Activity

- Select a Safe, then from the **Visual Security** menu, select **Inspect**, then **Safe Activity**,



or,

Select the Safe to inspect, then click **Inspect** on the toolbar.

Clearing Safe History

Periodically, you need to clear the Safe history to avoid unnecessarily long lists and confusion. Only file versions and Safe history logs that have been held for longer than the time specified in the Safe Properties History window can be deleted.

To Clear Safe History

- From the **Tools** menu, select **Clear Expired History**, then **Safe**.

The PrivateArk Client reminds you to clear expired Safe history on a regular basis when you log on to the Safe, by displaying the following message.



However, your System Administrator can make this action automatic so that expired history is cleared regularly without displaying this message window.

Alerts

-  This icon indicates routine Safe activity (e.g., logging on, opening a Safe, etc.)
-  This icon indicates an alert. (E.g., an attempt to log on to your account with an incorrect password or incorrect key).

Transparent User Management

The Privileged Account Security solution transparently supports User Accounts and Groups of users whose details are stored externally in LDAP-compliant directories. In order to maintain the typically high level of security in the Vault, the security attributes of LDAP User Accounts and Groups are managed internally.

An LDAP User Account is created the first time a User is referenced in one of the following situations:

- The user logs on to the Vault
- The user is added as a Safe member
- The user is added as a Group member

LDAP Groups are created when Groups that are defined in one or more external directories are added as Safe Owners or as members of a regular group in the CyberArk Vault.

A Directory Map determines whether a User Account or Group may be created in the Vault, and according to which criteria. Each Map contains a rules list which specifies the users and groups who can access the Vault, and a template which contains the security attributes and authorizations that will be applied when an LDAP User Account is created. During installation, the Privileged Account Security solution creates built-in directory maps for the most common Privileged Account Security solution users. You can use these directory maps immediately, modify them with relevant mapping rules according to your enterprise standards, or create new directory maps.

Security attributes and authorizations of an LDAP User or Group cannot be modified in the same way as a User Account or Group that has been created directly in the Vault. LDAP User Accounts and Groups are based on a template that is created as part of the Directory Map, and any changes must be made there. The LDAP User's Account properties are updated by the Directory Map each time the user logs on to the Vault.

Personal User details are retrieved from the external directory each time the user authenticates to the Vault, and therefore cannot be modified in the Vault, but only in the LDAP directory.



LDAP Users and Groups that have been created in the Vault appear in the Users list, marked with the LDAP User or Groups icon.

Managing Directory Maps

Predefined Directory Maps

During installation, the Privileged Account Security solution creates four built-in directory maps. You can use these directory maps immediately, modify them with relevant mapping rules according to your enterprise standards, or create new directory maps.

The following list describes the built-in directory maps and their details.

Vault Users Mapping

Applies to:	Users
Location:	"/"
User type:	EPVUser
Authentication:	LDAP Auth
Authorizations:	None
Default mapping rules:	-

Vault Groups Mapping

Applies to:	Groups
Location:	"/"
User type:	N/A
Authentication:	N/A
Authorizations:	None
Default mapping rules:	Map to base context

Vault Auditors Mapping

Applies to:	Users
Location:	"/"
User type:	EPVUser
Authentication:	LDAP Auth
Authorizations:	Audit Users
Default mapping rules:	-

Vault Admins Mapping

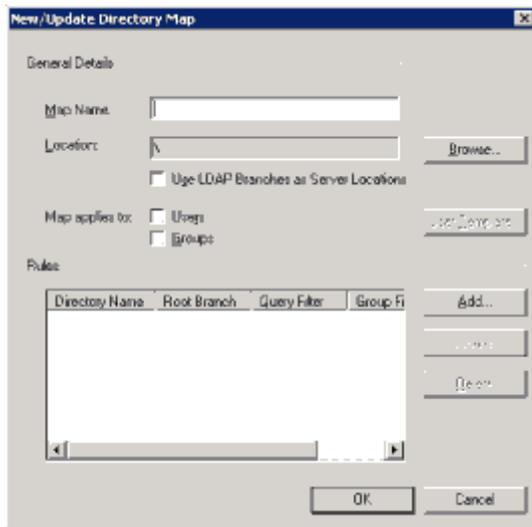
Applies to:	Users
Location:	"/"
User type:	EPVUser
Authentication:	LDAP Auth
Authorizations:	All permissions
Default mapping rules:	-

Creating a Directory Map

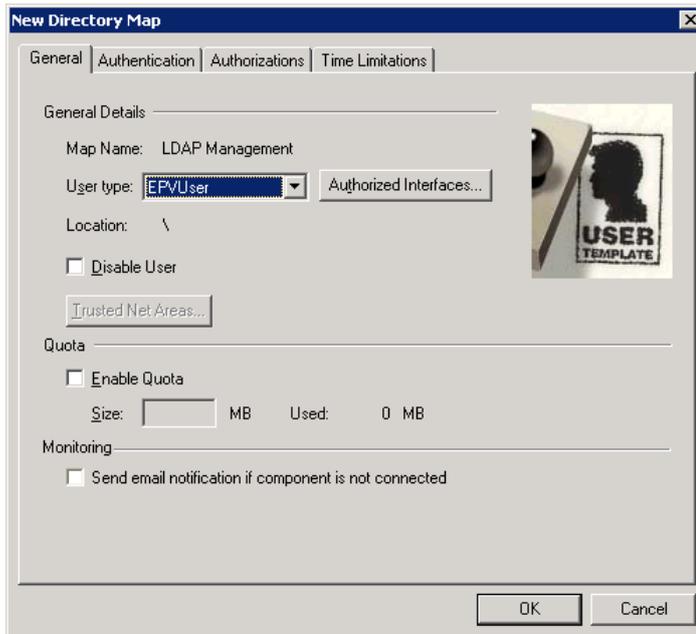
The directory map determines which users in the External Directories that the Vault recognizes will be able to access the Vault. Therefore, only users who have a high level of authority and responsibility in your organization should be given the “Manage Directory Mapping” authorization in the Vault to perform this operation.

To Create a Directory Map

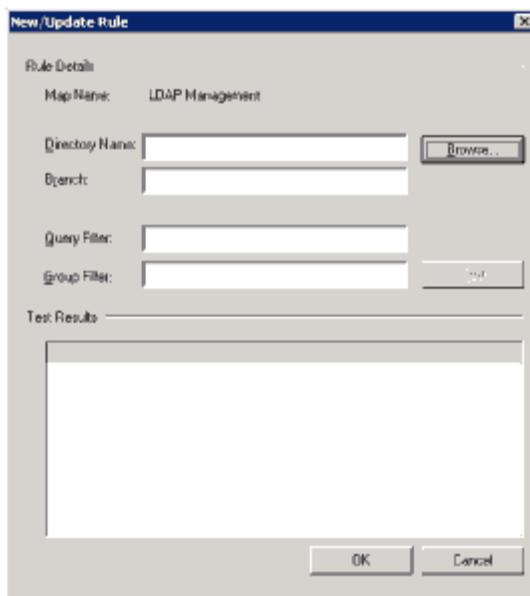
1. Log onto the PrivateArk Administrative Client as a Vault administrator.
2. From the **Tools** menu, select **Administrative Tools**, then **Directory Mapping**; the Directory Mapping for Vault window appears.
3. Click **Add**; the New/Update Directory Map window appears.



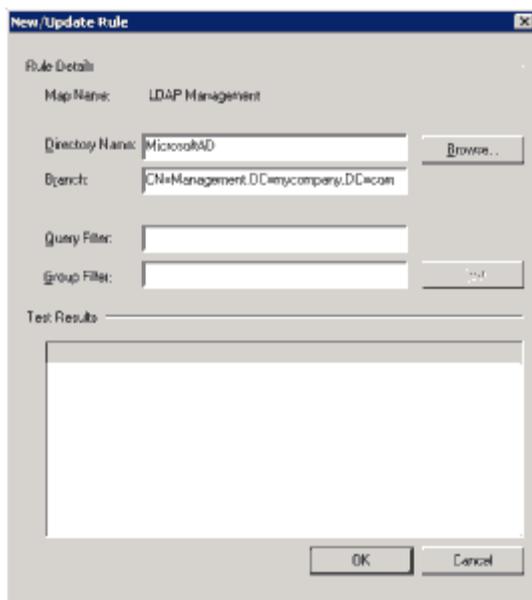
4. In the Map Name edit box, type the name of the Directory Map that will be created.
5. In the Location edit box, specify the location in the Vault hierarchy where the users who will be created according to this Map will be created.
Click **Browse** to display the Vault hierarchy and select the Location.
6. To create locations in the Vault according to the LDAP branches in the External Directories, select **Use LDAP Branches as Vault Locations**.
7. Depending on whether this Map will create users or groups, or both, select **Users**, **Groups**, or both.
8. If you select **Users**, the **User Template** button becomes active. Click **User Template** to display the New Directory Map window and specify the user properties that will be given to the External User Account when it is created.



- i. In the General tab, select the User type, then enable a quota, if necessary. To monitor this user type's activity, select **Send email notification if component is not connected**.
 - ii. In the Authentication tab, specify the authentication method that the user will use to log onto the Vault.
 - iii. In the Authorizations tab, select the Vault authorizations that will be allocated to users created with this Map.
 - iv. In the Time Limitations tab, specify the time allocations that will be allocated to users created with this Map.
9. When you have finished specifying the Directory Map, click **OK**; the new Directory Map is created and New/ Update Directory Map window appears again.
 10. In the Rules section, click **Add**; the New/Update Rule window appears.



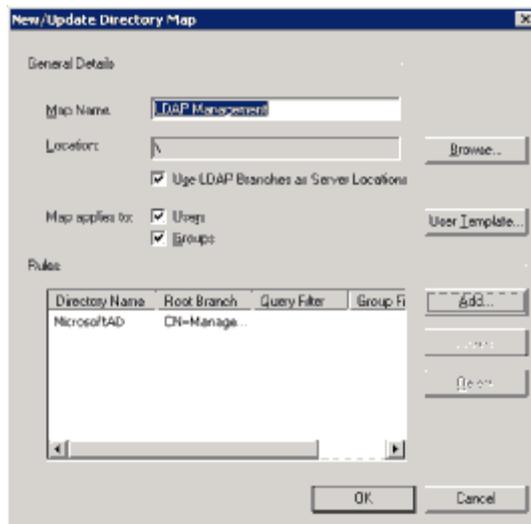
11. In the Directory Name edit box, specify the name of the External Directory whose users will be able to access the Vault with this map.
Click **Browse** to display a list of Directories and Branches that are available to the Vault.
12. Select a directory, then click **LDAP Connection** to connect to the selected External Directory; the Connect to Directory window appears.
13. Select **Login automatically using the current user credentials** to log onto the external directory with the credentials of the current Windows user,
or,
Select **Use other user credentials** to specify the User's DN and Password.
14. Click **Connect**; you are connected to the specified external directory.
15. To log on as a different LDAP user, in the Choose Directory and Branch window, click **LDAP Connection**, then specify different User credentials and click **Connect**; the directories that this user has access to will appear.
16. Select the branch that lists users who will be able to log on with this Map, then click **Select**; the New/Update Rule appears and displays the name of the selected directory and branch.



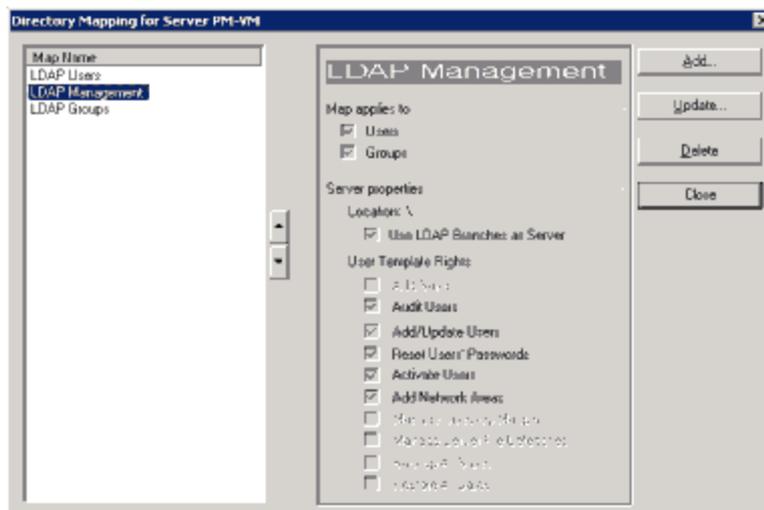
17. In the Query Filter, specify the filter that will be applied to the users in the specified branch to ensure that only certain users will have access to the Vault.
Note: The query filter is optional and should only be used when the location-based filtering is not focused enough to identify users.
18. In the Group Filter, specify the group in the specified branch that will be used to map users according to LDAP group membership. You can specify values to this field in regular expression format.
Note: The group filter is optional and should only be used when the location-based filtering is not focused enough to identify users.
19. Click **Test** to display the users that meet the specified criteria and for whom an External User Account can be created.

20. If the list of users is correct, click **OK** to add the specified directory and branch to the Rules list.

To add another rule to this Map, click **Add** and repeat steps 9-14.



21. When you have finished specifying Rules, click **OK**; the Directory Mapping for Vault window appears.



All the Directory Maps that have been created in the Vault appear in the Map Name list.

Allocating Map Order

The order in which the Maps appear in the Directory Mapping window indicates the order in which the Maps are matched with users and groups from the External Directory when determining if they can be created in the Vault.

The arrows next to the Map Name list enable you to move Maps higher or lower in the list, thus altering their priority.

Updating Directory Maps

You can update Directory Maps and change the rules and properties that will apply to LDAP User Accounts and Groups created with that Map.

In LDAP User Accounts, you can change the security attributes and authorizations. These properties will be applied when a new LDAP User Account is created or when an existing LDAP User logs on again, or is added as a Safe member or a Group member.

In LDAP Groups, you can change the type of locations hierarchy that will be used in the Vault when LDAP Groups are created.

To Update a Directory Map

1. Log onto the PrivateArk Administrative Client as a Vault administrator.
2. In the Directory Mapping for Vault window, select the Map to update, then click **Update**; the New/Update Directory Map window appears.
3. Update the Map rule and template as required.

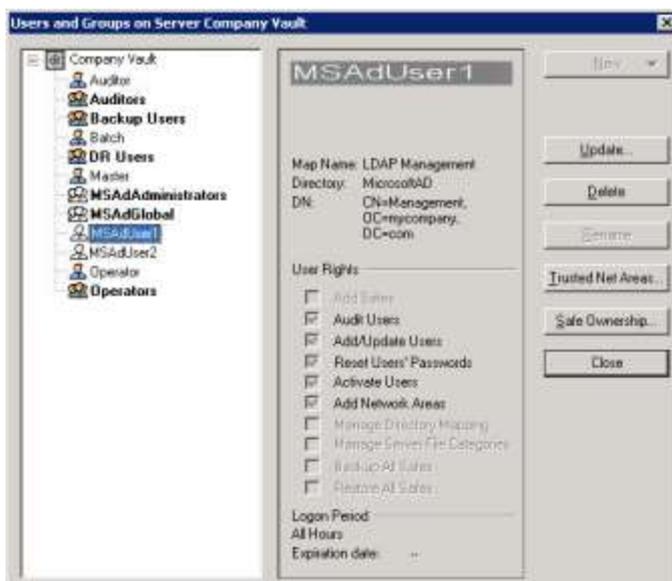
Modifying External User Accounts

After External User Accounts and Groups have been created in the Vault, you can view their properties and modify some of them in the External User Account in the Vault.

To Modify External User Accounts

1. Log onto the PrivateArk Administrative Client as a Vault administrator.
2. From the **Tools** menu, select **Administrative Tools** and then **Users and Groups**; the Users and Groups window appears.

The Users list displays all the users and groups that have been created in the Vault. LDAP users and groups are marked with special icons.



3. Select the External User Account to modify, then click **Update**; the Update Users window appears.

In the Update Users window, you can change the following user properties:

- The method that the user will use to authenticate to the Vault, including the following:
 - Password properties
 - The user certificate DN
 - Whether a user is disabled or not.
4. Update the user account properties as required, then click **OK**; the modifications are applied to the selected External User Account.

Setting Up PKI Authentication when the Certificate Subject Names are different from the Active Directory DNs

If a user's Distinguished Name (DN) in the Active Directory does not match the Subject in their PKI certificate, their user will not be identified and they will not be able to log onto the Vault. However, if at least one element of the DN matches the certificate subject, you can configure the Vault to identify LDAP users according to that specific element.

In the LDAP Profile file for the relevant directory, specify the following parameters:

- **UserNameDNElement** – The DN element of the Certificate Subject that will be used to match the user who is attempting to log on with the given PKI certificate.
- **ObjectCommonName** – Specifies the field in the Active Directory that will be matched with the value of the certificate DN element that is specified in the **UserNameDNElement** parameter.

The following example shows the DN listed in the Active Directory and the corresponding DN listed in the PKI certificate:

DN in Active Directory: CN=User;OU=mycompany;DC=com

Subject in PKI certificate: CN=User;OU=mycompany;DC=eu

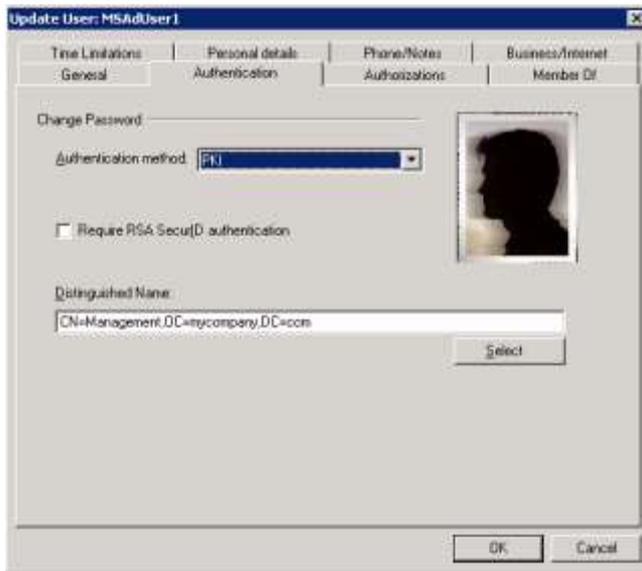
As the values of CN and OU are the same in both DNs, in the Profile set, you could specify either of them in the **UserNameDNElement** parameter. If you decide to specify CN, you would also specify CN in the **ObjectCommonName** parameter to enable the system to search in the Active Directory according to this DN element.

Changing the External User's Certificate

If the certificates that can be used to enable PKI authentication to the Vault are in the external directory, you can change the certificate that is specified in the External User's Account.

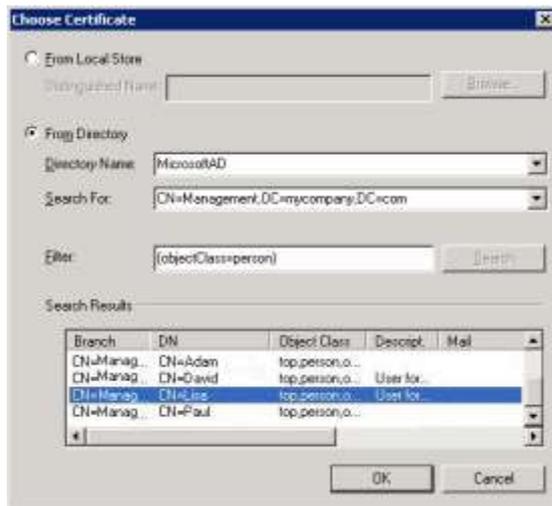
To Change the External User's Certificate

1. Log onto the PrivateArk Administrative Client as a Vault administrator.
2. From the **Tools** menu, select **Administrative Tools** and then **Users and Groups**; the Users and Groups window appears.
3. From the Users list, select the LDAP User Account to modify, then click **Update**; the Update Users window appears.
4. Select the **Authentication** tab; the User's authentication settings appear.



5. Click **Select**; the Choose Certificate window appears.
 6. Select a certificate from a local certificate store:
 - i. Select **From Local Store**, then click **Browse**, and select the certificate from the certificate list; the certificate's Distinguished Name appears in the Choose Certificate window.
 - ii. Click **OK**; the specified certificate's Distinguished Name appears in the authentication tab of the Update User window and can now be used to authenticate the LDAP user to the Vault.
- or,
- Select a certificate from an LDAP directory:
- i. Select **From Directory**, then click **Browse** to display the Choose Directory and Branch window.
 - ii. Select a directory. If you are not logged on to the LDAP directory, the Connect to Directory window appears,
 - or,
 - To log on as a different user and choose a group from a different LDAP directory, click **LDAP Connection** and specify the User credentials that will give you access to those directories.
 - iii. Select the branch that contains the required certificate, then click **Select**; the directory name and branch appear in the Choose Certificate window.

- iv. Specify the Query Filter, then click **Search**; all the certificates that meet the query filter criteria are displayed in the Search Results.



- v. Select the required certificate, then click **OK**; the specified certificate's Distinguished Name appears in the authentication tab of the Update User window and will now be used to authenticate the LDAP User to the Vault.
7. Click **OK**; the LDAP User Account's authentication properties will be updated.

Modifying LDAP Groups' Properties

After LDAP groups have been created in the Vault, their Safe Ownership properties can be altered.

To Modify LDAP Groups

1. Log onto the PrivateArk Administrative Client as a Vault administrator.
2. From the **Tools** menu, select **Administrative Tools** and then **Users and Groups**; the Users and Groups window appears.
3. Select the LDAP group to modify; only the **Delete** and **Safe Ownership** buttons are active, indicating that these are the only activities that can be carried out on these groups.
4. Modify the Safe Ownership properties as required, then click **OK**; the modifications are applied to the selected Group.

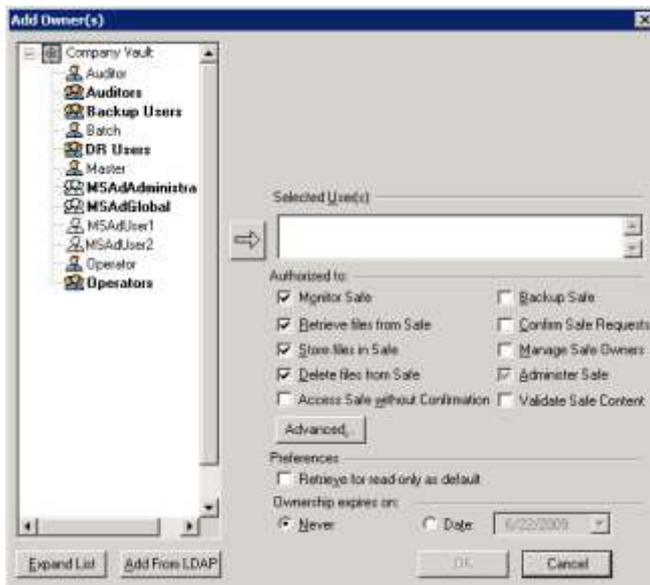
Managing Safe Ownership for LDAP Users and Groups

After LDAP User Accounts and Groups have been created in the Vault according to Directory Maps, they can be given Safe ownership rights in the same way as any other user account in the Vault. Alternatively, LDAP Users and Groups can be added to the Safe owners list directly from the external directory.

Safe ownership for LDAP users and groups can also be managed in the PVWA. For more information, refer to *Adding and Managing Safe Members*, page 68.

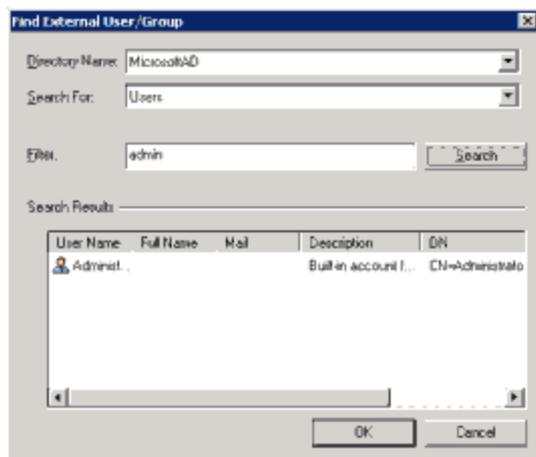
To Grant Safe Ownership to LDAP Users and Groups

1. Log onto the PrivateArk Administrative Client as a Vault administrator.
2. Open the Safe to which to add Owners, then from the **Safe** menu, select **Owners**,
or,
Click **Owners** on the PrivateArk toolbar.
The Owners window appears.
3. Click **Add**; the Add Owners dialog box appears.



4. Select the name of a user or group to add to the Safe Owners list.
If the name of the user to add as a Safe Owner doesn't appear in the list, click **Expand List** to display all users who share Safes with you ('known users')
or,
Add the LDAP user directly from one of the external directories that can be accessed by the Vault:
 - i. Click **Add From LDAP**; the Find External User/Group window appears.
 - ii. Select the directory that contains the user or group to add as a Safe owner.
 - iii. Click **Search For** then select either Users or Groups.

- iv. In the Filter edit box, specify the filter that will identify users or groups to add as external users or groups, then click **Search**; a list of users or groups that meet the specified criteria appears in the Search Results. The filter keywords you enter will be matched against users' first and last names, as well as their email address and actual user or group name in the external directory.



- v. Select an item in the list, then click **OK**; the selected user or group is added to the Add Owners list in the PrivateArk Client.

Note: If you cannot see a user in the hierarchy, you might not be 'familiar' with him. Type his CyberArk Vault username manually in the Selected User(s) field.

5. Select the Safe Owner authorizations and preferences.
6. Click **OK**; the Safe Owner is added to the Safe Owners list with the authorizations that you have set.

For more information about Safe ownership, refer to *Adding and Managing Safe Members*, page 68.

Adding LDAP Users to Vault Groups

You can add External Users to groups in the Vault either after the External User Account has been created in the Vault or through a direct connection to the external directory.

To Add an External User to a Vault Group

1. Log onto the PrivateArk Administrative Client as a Vault administrator.
2. From the **Tools** menu, select **Administrative Tools** and then **Users and Groups**; the Users and Groups window appears.
3. Select the Group to add the External User to, then click **Update**; the Update Group window appears.
4. In the Members section of the window, click **Add**; a drop-down list enables you to add either external users or LDAP groups to the group.
5. Select **User**; the Add Members to 'Group' window appears.

6. If the External User to add appears in this list, select it, then click the arrow to move the user over to the Selected User(s) list,
or,
Select the External Group directly from the external directory:
 - i. Click **Add from LDAP**; the Find External User/Group window appears.
 - ii. Select the directory that contains the user to add as a Safe owner.
 - iii. Click **Search For** and select Users, or leave it as the default value.
 - iv. In the Filter edit box, specify the filter that will be applied to the users in the specified branch, then click **Search**; a list of users that meet the specified criteria appears in the Search Results.
In the Filter edit box, specify the filter that will identify the users to add as external users, and later to add as group members, then click **Search**; a list of users that meet the specified criteria appears in the Search Results.
The filter keywords you enter will be matched against users' first and last name, as well as their email address and actual user name in the external directory.
 - v. Select a user, then click **OK**; the selected user appears in the External Group edit box in the Update Group window.
7. Click **OK** to add the External User as a member of the Vault group.

Adding LDAP Groups to Vault Groups

You can add LDAP Groups to groups in the Vault either after the LDAP Group has been created in the Vault or through a direct connection to the LDAP directory.

To Add an LDAP Group to a Vault Group

1. Log onto the PrivateArk Administrative Client as a Vault administrator.
2. From the **Tools** menu, select **Administrative Tools** and then **Users and Groups**; the Users and Groups window appears.
3. Select the Group to add the LDAP Group to, then click **Update**; the Update Group window appears.
4. In the Members section of the window, click **Add**; a drop-down list enables you to add either external users or LDAP groups to the group.
5. Select **LDAP Group**; the Add External Group to 'Group' window appears.
This window displays all the LDAP Groups that have already been created in the Vault.
6. If the LDAP Group to add appears in this list, select it, then click the arrow to move the group name over to the edit box. You can add as many LDAP groups as necessary.

Or,

Select the LDAP Group directly from the external directory:

- i. Click **Add from LDAP**; the Find External User/Group window appears.
- ii. Select the directory that contains the user or group to add as a Safe owner.
- iii. Click **Search For** then select Groups.
- iv. In the Filter edit box, specify the filter that will identify groups to add as external group, then click **Search**; a list of groups that meet the specified criteria appears in the Search Results.

- v. Select a group, then click **OK**; the selected group appears in the External Group edit box in the Update Group window.
7. Click **OK** to add the LDAP Group as a member of the Vault group.

Deleting LDAP Users and Groups

As LDAP Users and Groups are dependent on the external directory to log onto the Vault, deleting their user accounts does little more than erase any changes that may have been made in the Vault User Accounts after they were created. If the User's name and details appear in the LDAP directory, he will be able to log onto the Vault and create a new User account at any time.

After a User has been removed from the LDAP directory, he can no longer log onto the Vault. However, his User name still appears in the Vault's User hierarchy until the next time that the Vault hierarchy is synchronized with the LDAP directory, when his User Account in the Vault will be erased.

After a Group has been removed from an external directory, the corresponding Group in the Vault is erased the next time that the Vault hierarchy is synchronized with the external directory.

For additional information and installation instructions, contact CyberArk support.

Configuring LDAP Connectivity for Users' Logon

When the Vault is configured to work with LDAP directories, it can be configured to work in the following ways:

- **Requires connectivity with LDAP** – The Vault server will only start working if it can connect to the configured LDAP directories. If the Vault server is already running, but the connection to configured LDAP directories is not active, LDAP users will not be able to log on and the offline synchronization process between the LDAP directory and the Vault server will fail.

This configuration ensures that the Vault will not operate if it cannot connect to the LDAP directory to synchronize details. It provides the highest level of security and ensures that users can access the Vault according to their real-time configurations in the LDAP directory.

- **Does not require connectivity with LDAP** – The Vault server will start working regardless of the LDAP connectivity status, although a warning message will be displayed. All LDAP users (except those authenticating with LDAP authentication) will be able to log onto the Vault and their definitions will not be updated from the directory. In addition, the offline synchronization process between the LDAP directory and the Vault server will fail.

This configuration provides full Vault availability for LDAP users who can log onto the Vault at any time, regardless of the connectivity status of the LDAP server.

In `DBParm.ini`, add the **RequireLDAPConnectivity** parameter and set it according to your implementation needs. This parameter enables you to configure whether or not the Vault will start up when there is no active connection with the configured LDAP directory. By default, this parameter is set to **No**, which means that the Vault will start running, regardless of an active LDAP connection.

Managing Users and Groups who are Listed in Multiple Directories

In order to prevent naming collisions of users and groups with the same name that reside in multiple directories, you can provision and authenticate LDAP users and groups who are listed in multiple LDAP directories by adding the LDAP domain name to the name of the corresponding user or group that is created in the Vault.

For example, PaulBlack who is listed in an LDAP domain called Company.com will have a Vault user called **PaulBlack@Company.com**. If this user is also listed in an LDAP domain called Branch.Company.com, a different Vault user called **PaulBlack@Branch.Company.com** will also be created. The same process is applied to groups that are provisioned in the Vault.

This name structure is configured separately for each directory that is recognized by the Vault.

Note: Currently, this can only be configured for Microsoft Active Directory.

The CyberArk solution provisions non-unique user names across multiple directories for the following authentication methods:

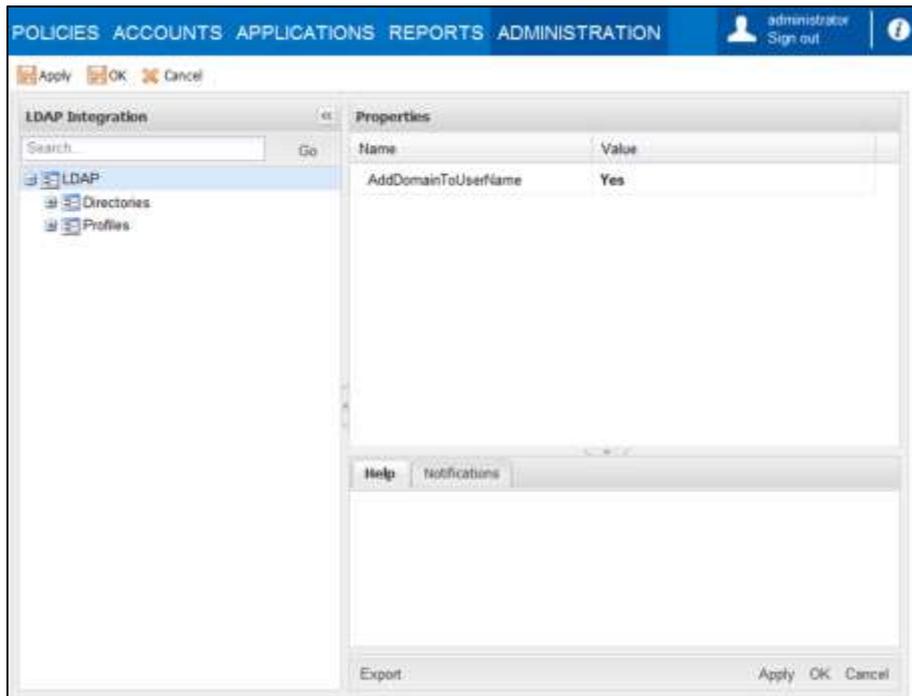
- PVWA Windows authentication
- LDAP or RADIUS authentication which requires the users to specify the full user name in the following format: USER@DOMAIN.

Existing LDAP users whose name does not include the domain name can still authenticate to the Vault with their current user name.

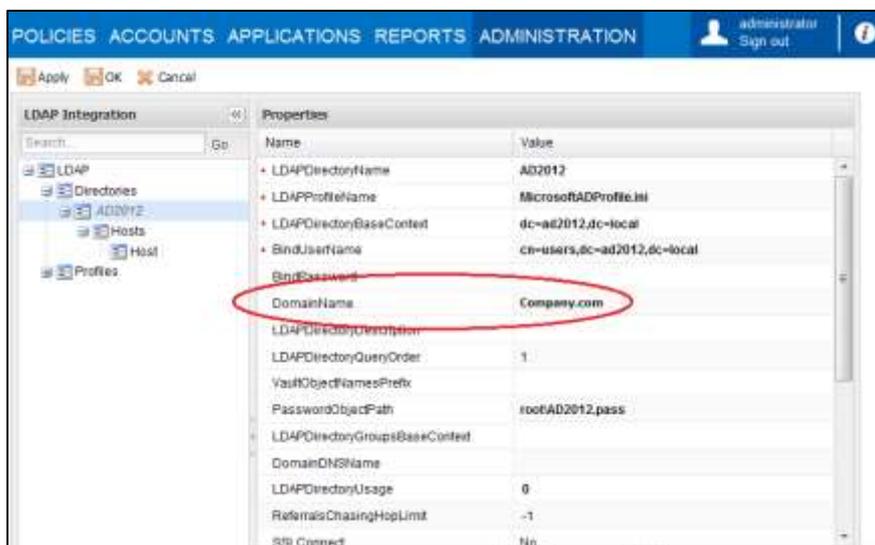
Note: If the domain name changes after you configure the Vault to recognize it, contact your CyberArk Support representative.

To Append the LDAP Domain Name to a Vault User or Group Name

1. Log onto the PWWA as an administrator user. Make sure that this user belongs to the **Vault Admins** group so that you have the required permissions to configure LDAP integration.
2. Click **ADMINISTRATION** to display the **System Configuration** page, then click **LDAP Integration**; the LDAP Integration page appears.
3. Select **LDAP**; the LDAP properties are displayed in the Properties pane.
4. Specify the following value:
 - **AddDomainToUserName** – Whether or not the Domain name will be added to Vault user or group names when a corresponding user account is created. Specify **Yes**.



5. Expand **Directories** and select the Directory whose name will be added to the Vault user or group name; the Directory properties are displayed.
6. Specify the following value:
 - **DomainName** – The Domain name that will be added to the name of the user or group that is created in the Vault for users or groups listed in this LDAP directory. Specify the pre-Windows 2000 Domain name of the Microsoft Active Directory, as it is defined in the Active Directory Domains and Trusts mms snap-in.



7. Make sure that the **VaultObjectNamesPrefix** property does not specify a value.
8. Click **Save** to save the new configurations.
 - These changes will be applied within the PWA refresh period, which could be up to one hour.
 - To apply these changes immediately, run `iisreset`.

The Master Policy

The Master Policy offers a centralized overview of the security and compliance policy of privileged accounts in your organization while allowing you to configure compliance driven rules that are defined as the baseline for your enterprise. It is configured out-of-the-box and can be used immediately after implementation, providing an intuitive, simplified user experience and enhanced bottom-line insight for administrators, IT personnel, managers and auditors.

The Privileged Account Security solution separates higher-level and compliance driven policy rules such as privileged access workflows, account management and session monitoring requirements from technical settings that determine how the policy will be carried out on each platform.

The Master Policy groups together sets of rules and offers better visibility and control over policy configurations and enforcement. Each policy rule has basic settings and, sometimes, advanced settings that are displayed when you select the rule, as well as context-sensitive help that explains each rule and its interdependency on other rules.

Although the Privileged Account Security solution's Master Policy can be applied to most privileged accounts in your organization, you can create rule exceptions to manage specific workflows. For example, you can define a dual control workflow for highly sensitive accounts on a specific platform that require permission from authorized users before they can be used, while access to other accounts in the organization does not require such confirmation.

The Master Policy defines basic system behavior for the entire lifecycle of privilege account management and access.

The Master Policy includes the following main concepts:

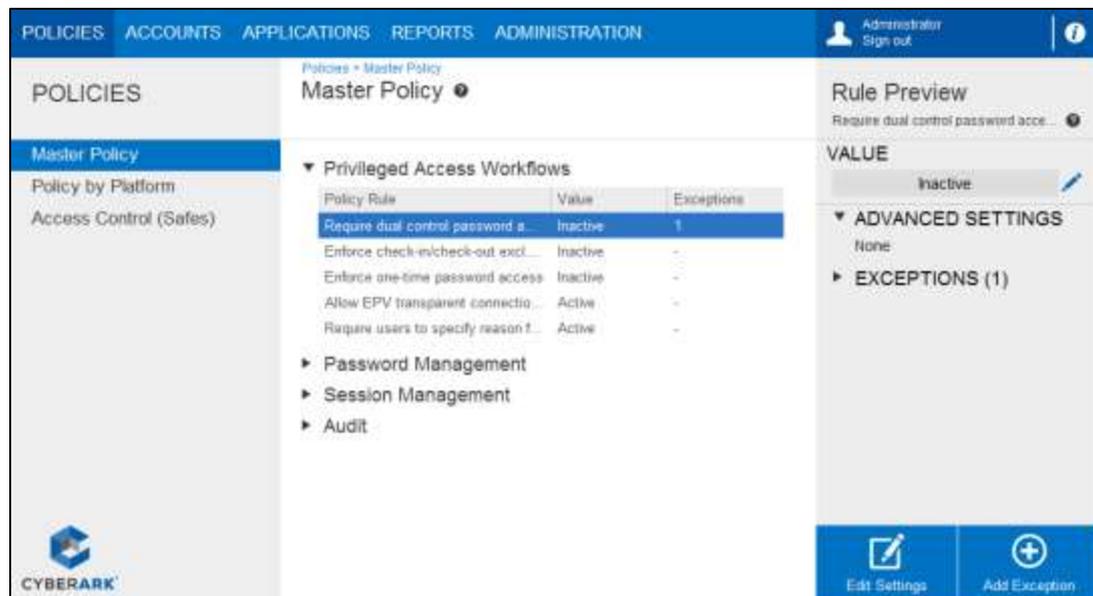
- Basic policy rules allow you to define specific aspects of privileged account management. These rules include several groups of policy rules for the access workflow, management of passwords, session monitoring and auditing.
- In addition, some policy rules have related advanced settings. For example, in the basic policy rules you can determine whether users will be allowed to transparently connect to target systems using 'Click to Connect'. In the related advanced settings, you can determine whether users will also be able to view passwords.
- The new Master Policy model introduces the ability to define Exceptions. These are policy rules that differ from the overall Master Policy for a specific scope of accounts, for example accounts associated with a specific platform. Each exception contains the basic policy rule as well as its related advanced settings. For example, the Master Policy may define that Dual Control is disabled in the organization. However, the Windows PCI production servers require Dual Control to be enabled because of their higher sensitivity. You can make this allowance by creating an exception to the Dual Control rule that enables Dual Control enforcement on the scope of Windows PCI production servers platform.

In the Platform Management settings, the IT administrator can configure technical settings defined by your organization's environment and security policies to control how the system manages accounts on various platforms. Most of these settings have default values that do not need to be changed, but certain specific features need to be set according to your organizational requirements.

Working with Master Policy Rules

The Master Policy enables you to define a baseline for how you manage accounts in your organization. You can define and view these rules in the Master Policy page. Click a section title to display the policy rules defined in that section, as well as the status of each rule and possible exceptions to it.

In order to display this page, users must be members of the Vault Admins group.



The Master Policy Page enables you to set Master Policy rules for privileged access workflows, password management, and session management. These are all described below:

- **Privileged Access Workflows** – This enables you to view the main policy rules and settings that define how you manage access to privileged accounts in your organization.
 - **Require dual control password access approval** - Users must receive approval from authorized users before they can access passwords. This enables you to see who wants to access passwords, when, and for what purpose. By default, this rule is **inactive**.

Advanced settings enable you to determine the following workflows:

- Whether requests for privileged accounts require approval from multiple levels of users.
- Whether requests for privileged accounts must be approved by a direct manager.
- The number of authorized users required to confirm requests.

For more information, refer to *Dual Control*, page 251.

- **Enforce check-in/check-out exclusive access** – Users can check out an account and lock it so that no other users can retrieve it at the same time. After the user has used the password, they check the password back into the Vault. Together with enforcing one-time password access, this restricts access to a single user, ensuring exclusive usage of the privileged account and guaranteeing accountability. By default, this rule is **inactive**.

For more information, refer to *Accounts Check-out and Check-in*, page 246.

- **Enforce one-time password access** – Accounts can be retrieved for one-time use only, and the password stored inside must be changed after each use before the account is released and can be used again. Passwords can be changed automatically by the Privileged Account Security solution's password management capability. By default, this rule is **inactive**.

For more information, refer to *Accounts Check-out and Check-in*, page 246.

- **Allow EPV transparent connections ('Click to connect')** – Users can connect to remote devices without needing to know or specify the required password. This prevents the password from being exposed to the user and maintains productivity as the user does not have to open a login session and then copy and paste the password credentials into it. In addition, advanced settings define whether or not users are permitted to view passwords. This enforces strong authentication for accessing managed devices and restricts user access to passwords according to granular access control. By default, this rule is **active**.

For more information, refer to *Configuring Transparent Connections*, page 527.

- **Require users to specify reason for access** – Users can only retrieve accounts after they specify a reason that explains why they want to retrieve them. By default, this rule is **active**.

An advanced setting determines whether users will be able to specify a free text reason in the Reason edit box or will be required to select one of the predefined reasons.

- **Password Management** – This enables you to view the rules that determine how passwords are managed.
 - **Require password change every X days** – The Master Policy determines how frequently passwords must be changed. By default, passwords are changed every **90** days. You can see when password changes are planned in the Compliance Report.
 - **Require password verification every X days** – You can define a Master Policy to verify passwords after the timeframe specified in the previous rule. Passwords can be changed manually or replaced by a unique and highly secure password that is randomly generated by the Password Vault. By default, passwords are verified every **7** days.
- **Session Management** – This enables you to view the rules that determine whether or not privileged sessions are recorded, and how they are monitored.
 - **Require privileged session monitoring and isolation** – You can define a Master Policy to monitor and isolate all IT administrator privileged sessions on remote machines. By default, this rule is **inactive**.
 - **Record and save session activity** – You can define a Master Policy to record all the activities in each privileged session in text and/or video format, and stored them in the Vault, compressed, for future auditing. These recordings are transparent to users and cannot be bypassed. By default, this rule is **active**.
- **Audit** – This rule enables you to determine how Safe audits are retained.
 - **Activities audit retention period** – The Master Policy controls the number of days that Safe activities audits are retained. By default, audits of activities are kept for **90** days.

Note: If this parameter is set to zero, activities in the Safe will not be written in an audit log.

Exceptions

After setting a Master Policy that determines how accounts will be managed in the entire organization, you can create exceptions to add granularity as needed and set different behavior for specific platforms that will override the corresponding rules set by the Master Policy. Exceptions can be set for a scope of accounts associated with a specific platform. The Master Policy, together with the exceptions defined on each platform, determine the resultant behavior of the system on each account, based on its Platform.

To define more granularity for a specific scope of accounts, such as the Windows PCI accounts, after you define the Master Policy, you can duplicate a Windows platform in Platform Management and define an exception that contains specific rules that are relevant to Windows PCI only. The unique combination of the Master Policy rules together with the exception ensures that each platform is managed exactly according to your needs, with minimum configuration.

Initially, when a user adds an exception, it inherits all values from the Master Policy and these values still adopt any changes made in the Master Policy. However, if a user changes the value of any setting in the exception, either basic or advanced, the new value overrides the value that was inherited from the Master Policy and disconnects the setting value from the Master Policy. To emphasize this, a broken chain icon is displayed next to the 'disconnected' setting.

In addition, any changes made in a Master Policy after an exception is created do not affect any settings in the exception that override the Master Policy; they only affect the settings in the exception that inherit directly from Master Policy. This is especially relevant when a rule contains several basic and advanced settings, and some of the exception settings may inherit values from the Master Policy and some override it.

For example, an enterprise decides that users can connect directly to target systems ("Click to Connect") but can still view passwords when needed (i.e. utilize the "Show" or "Copy" functions). However, the Windows PCI accounts cannot be viewed by users and can only be accessed through the 'Connect' button. In this case, an exception will be created for the rule that defines that users can connect directly to target systems on a Windows PCI platform. The basic setting remains without changes (meaning that it inherits from the Master Policy), while the advanced setting that determines that users can view passwords will be disabled, overriding the Master Policy.

Example

The following scenario describes a typical workflow using the Master Policy and Platform Management technical settings.

In a large enterprise that manages multiple accounts on local and remote machines, a Risk Manager has issued a security policy defining that all passwords in the organization must be changed every 90 days. In response, the IT/IS Group Manager informed him that passwords for the Windows PCI systems in the organization's US offices only need to be changed once a year. In addition, the Vault Administrator has suggested using a different port to manage Windows_US PCI systems.

To ensure compliance with enterprise and standard policies, the Compliance Auditor emphasizes the importance of compliancy and wants to know how to verify that all accounts comply with the Master Policy.

The new Master Policy enables all the above users to get what they want:

- The Master Policy defines password changes for all privileged accounts every 90 days.
- An exception is created within the Master Policy to change passwords on Windows PCI systems in the US offices once a year.
- The Compliance Auditor can see the effective accounts policy that is enforced throughout the organization in the Master Policy, and can view compliancy to it in the standard compliance report.
- Finally, technical settings that are set in the new Platform Management page enable the Vault administrator to set a different port or any other technical settings for all accounts that are managed on the Windows_US PCI system.

Auditing Master Policy Activity

Authorized users can view a list of changes that have been made in the Master Policy by generating an Activities report. This includes changes made in Master Policy rules, as well as adding, modifying, and deleting exceptions.

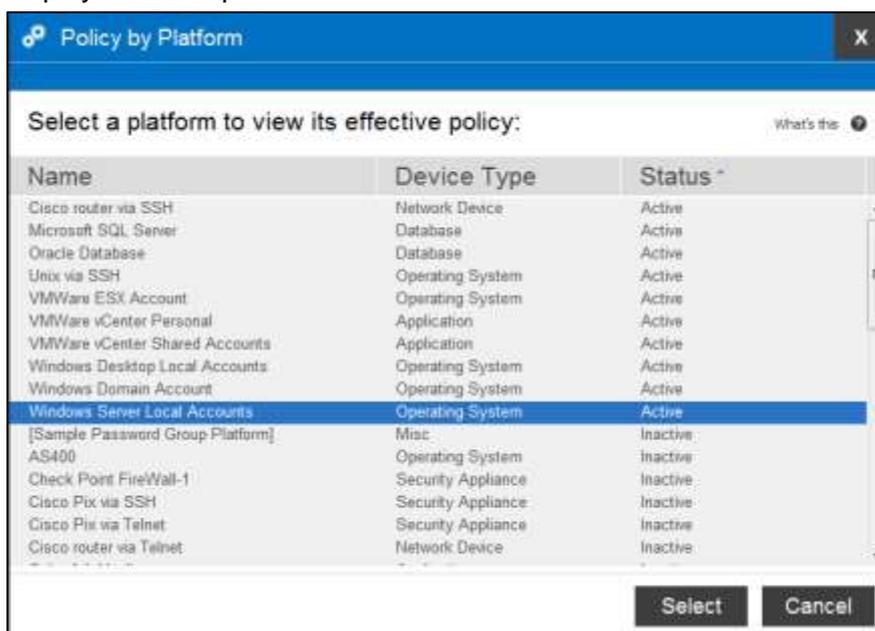
For more information about generating reports, refer to *Generating Reports in the PVWA*, page 341.

Viewing the Effective Policy by Platform

The Policy by Platform view displays the effective policy settings per platform based on the Master Policy, together with any exceptions that were defined for this platform. When a selected rule has an exception, the Rule Preview pane specifies the value for this rule and indicates that the basic value and/or Advanced values are disconnected from the Master Policy.

To View the Effective Policy for a Specific Platform

1. In the POLICIES page, click **Policy by Platform**; the Policy by Platform window displays a list of platforms.



2. Select a platform, then click **Select**; the effective policy settings for the selected platform are displayed.

This page displays the following information for each rule:

- **Value** – This column indicates the effective value of this rule based on the Master Policy value together with the value of any exceptions defined for this platform.
- **Exception** – This column indicates whether or not an exception to this rule exists for the selected platform. The Rule Preview pane indicates whether the exception exists in the basic value and/or the Advanced value.

The following example shows the effective policy for Windows Server Local Accounts.

The screenshot shows the CyberArk Policy by Platform interface. The main content area displays the effective policy for 'Windows PCI'. The policy is categorized under 'Privileged Access Workflows' and includes several rules. The 'Require dual control password access approval' rule is highlighted, showing a value of 'Inactive' and an exception of 'Yes'. Other rules include 'Enforce check-in/check-out exclusive access', 'Enforce one-time password access', 'Allow (IPV) transparent connections (Click to connect)', and 'Require users to specify reason for access'. The 'Password Management' section includes 'Require password change every X days' (30 days, Yes exception) and 'Require password verification every X days' (7 days, No exception). The 'Session Management' section includes 'Require privileged session monitoring and isolation' (Active, No exception) and 'Record and save session activity' (Active, No exception). The 'Audit' section includes 'Activities audit retention period' (90 days, No exception). The right-hand pane shows the 'Rule Preview' for 'Require dual control password access approval', indicating it is 'Inactive' and 'Disconnected from Master Policy'. The 'ADVANCED SETTINGS' section is currently 'None'.

To Change the Platform

1. In the Policy by Platform page, click **Change Platform**; the list of platforms appears again.
2. Select the platform whose effective policy you want to display, then click **Select**; the effective policy settings for the selected platform are displayed.

Managing Target/Service Account Platforms

Target account platforms and service account platforms define the technical settings which determine how the system manages accounts on different platforms. All the platforms supported by the Privileged Account Security solution are configured out-of-the-box with default values for most of the settings. The technical settings under each platform include settings that determine how password management operations take place, transparent connections, PSM connections, etc. Some settings, such as logon or reconcile accounts, do not have defaults and require setting up when needed. You can manage target account platforms as well as service account platforms.

The Privileged Account Security solution supports remote password management and change on the following target platforms:

- Operating Systems
- Databases
- Security Appliances
- Network Devices
- Directories
- Applications

Predefined platform settings for each platform determine the following:

- How frequently a password will be changed and/or verified, the password management timeframe, notification settings, and a variety of other management criteria and capabilities. For more information, refer to *Changing Passwords*, page 197.
- The rules that must be applied when a new random password is generated. These rules must match the password rules on the remote machine where the password will be used, so that the password will be accepted during the password change operation as well as during logon.
- Additional information common to all accounts associated with this platform.

In order for users to be able to add passwords to the Vault through the Password Vault Web Access, the supported platform must be specified. A list of platforms can be accessed and configured in the Platform Management page. You can use the default parameters as they are or you can add/edit mandatory or optional password properties. In this way, you can customize the system to meet your own organization's policy requirements.

When the user adds a password in the Password Vault Web Access, he will be able to allocate any predefined supported platform. The required and optional properties for the selected platform will appear automatically, so that he can specify the required information.

The following options in the platform settings page enable you to customize account management on supported target platforms:

- **UI & Workflows** – Customizes account management workflows on target accounts, such as ticketing systems and associated logon and verification accounts.
- **Automatic Password Management** – Defines how passwords are managed in the Privileged Account Security solution.

These configurations can be viewed and modified by default by users with membership in the following group:

- Vault Admins

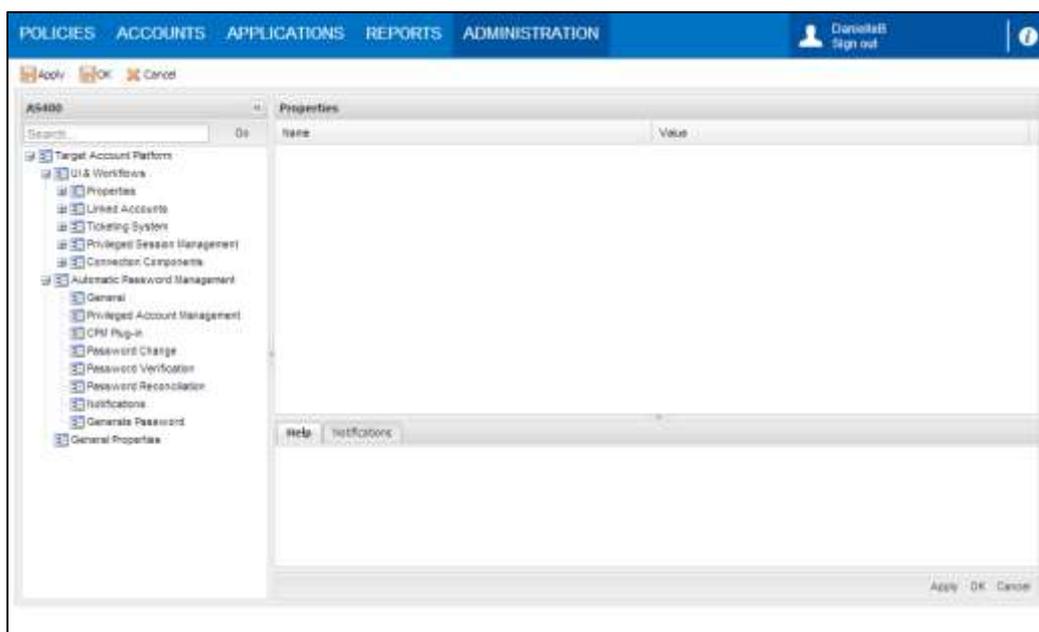
Note: By default, platform configurations are applied to all Safes. From a security aspect, it is recommended to use the AllowedSafe parameter to enable platforms for specific Safes.

Modifying Target Account Platforms

You can customize target account platforms to define technical settings for specific platforms and password management workflows.

To Customize a Target Account Platform

1. In the list of supported target account platforms, select the platform to modify, then click **Edit**; the platform settings page appears.



2. Change existing parameter values and/or add new values to customize the platform.
3. Click **Apply** to save the new configurations and apply them immediately, or,
Click **Save** to save the new configurations and apply them after the period of time specified in the **RefreshPeriod** parameter.

For more information about the above parameters, refer to *Configuring the PVWA*, page 494.

Adding New Platforms

You can add new platforms in either of the following ways:

- **Duplicate existing platforms** - Duplicate existing platforms and customize them to support accounts on target accounts according to very specific standards. For more information, refer to *Duplicating an Existing Platform*, page 110.
- **Import new platforms** - Import new platforms that are not included in the default installation directly into the Privileged Account Security solution. For more information, refer to *Importing New Platforms*, page 111.

Defining Account Properties in the Vault

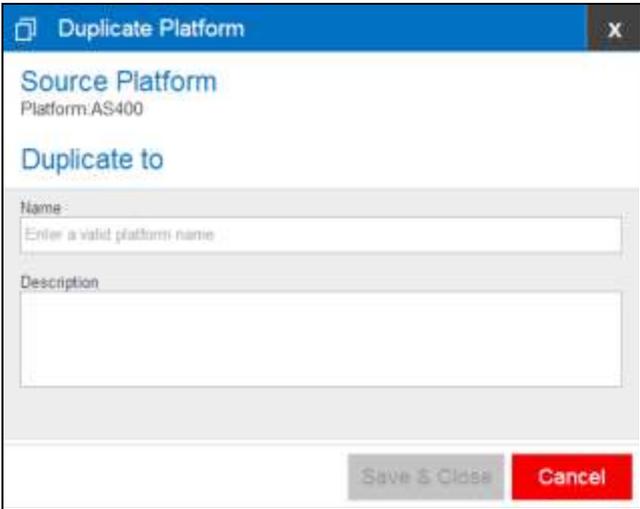
Before you can set new custom properties in a platform as optional or required, they must be defined in as password properties in the Vault.

- In the PrivateArk Administrative Client, define the new password properties that will be added to the target account platform.

For more information and specific instructions about adding password properties in the Vault, refer to *Defining Custom Account Properties*, page 161.

Duplicating an Existing Platform

1. Click **ADMINISTRATION** to display the **System Configuration** page, then click Platform Management to display a list of supported target account platforms.
2. Select an existing platform that is similar to the new target account platform, then click **Duplicate**; the Duplicate Platform window appears.



The screenshot shows a 'Duplicate Platform' dialog box. The title bar is blue with the text 'Duplicate Platform' and a close button (X). The main content area is white and contains the following elements: 'Source Platform' with the value 'Platform:AS400', 'Duplicate to' section with a 'Name' field containing the placeholder text 'Enter a valid platform name' and a 'Description' field. At the bottom, there are two buttons: 'Save & Close' (grey) and 'Cancel' (red).

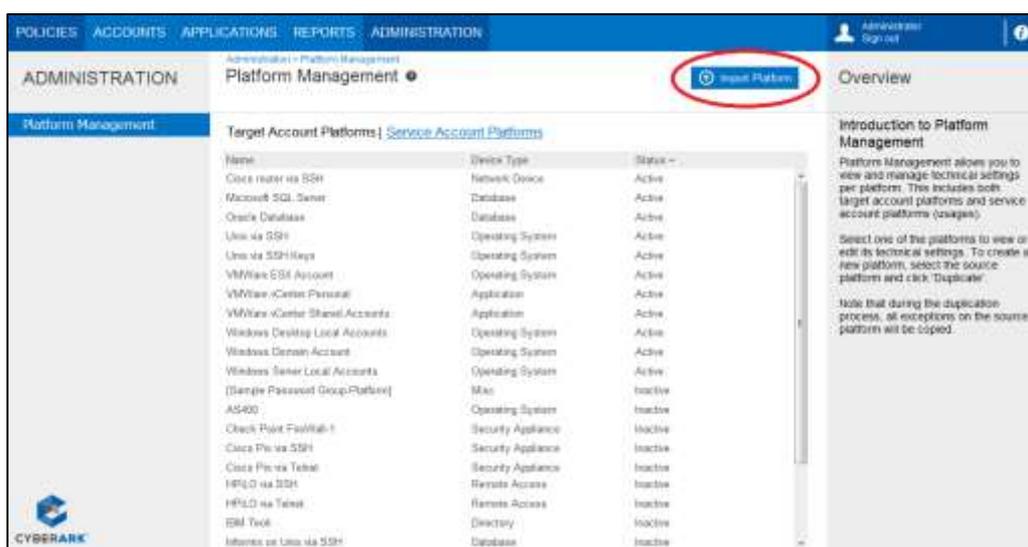
3. Type the name and a description of the new platform, then click **Save & Close** to create the new platform.
4. Select the new target account platform, and then click **Edit**; the configuration page for the selected platform appears.
5. Change existing parameter values and/or add new values to define the new platform.

- Click **Apply** to save the new configurations and apply them immediately, or, Click **Save** to save the new configurations and apply them after the period of time specified in the **RefreshPeriod** parameter.

Importing New Platforms

You will receive a platform package from your CyberArk support representative. The following procedure describes how to import the new platforms in this package.

- In the Platform Management page, click **Import Platform**; the Open window appears.



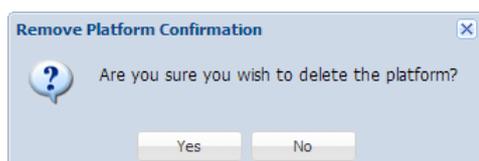
- Navigate to the location of the platform package to import and select it, then click **Open**; the Privileged Account Security solution imports the new platform and applies the new platform settings to all the relevant components. The new supported platform appears in the list of Target Account Platforms.

Deleting Platforms

If you are sure that you will not support accounts on a certain target platform, you can delete the platform.

To Delete a Platform

- In the list of Target Account Platforms, select the platform to delete, then click **Delete**; the following confirmation message appears.



- Click **Yes** to delete this platform, or, Click **No** to close this message and return to the Target Account Platforms list.

Activating and Deactivating Platforms

The Privileged Account Security solution supports over 50 built-in platforms out-of-the-box and lists them all in the Platform Name drop-down list that is displayed when users create new accounts. To simplify implementations, your Vault administrator can mark the platforms that are not currently relevant to your implementation as 'Inactive', hiding them from end-users and reducing the size of the list. As these platforms are only marked as 'Inactive' and have not been deleted, they can be marked 'Active' at any time and will be included in the list of supported platforms again. This also facilitates step-by-step implementations during which platforms can be made active at different phases.

The Target Account Platforms list displays all the supported platforms and their status. In addition, the status of each platform is displayed in the Platform Preview pane.

The CPM does not manage inactive platforms and they cannot be assigned to new or modified accounts.

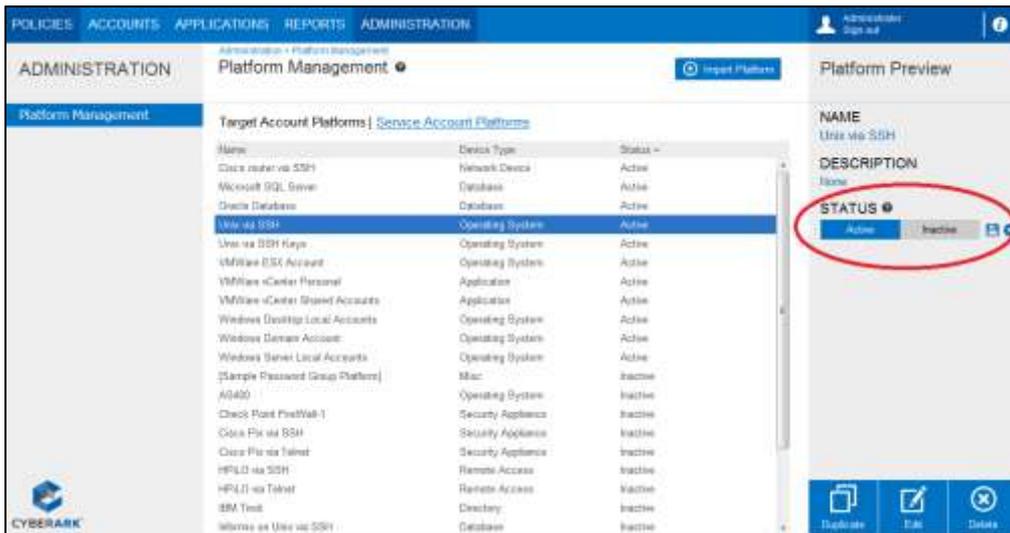
Note: This is only relevant to Target Account Platforms and not to Service Account Platforms.

The screenshot displays the 'Platform Management' interface. On the left, a table lists 'Target Account Platforms' with columns for Name, Device Type, and Status. The 'User via SSH' platform is selected. On the right, the 'Platform Preview' pane shows details for the selected platform, including a 'STATUS' field which is circled in red and displays 'Active'.

Name	Device Type	Status
Cisco router via SSH	Network Device	Active
Microsoft SQL Server	Database	Active
Oracle Database	Database	Active
User via SSH	Operating System	Active
User via SSH Keys	Operating System	Active
VMWare ESX Account	Operating System	Active
VMWare vCenter Personal	Application	Active
VMWare vCenter Shared Accounts	Application	Active
Windows Desktop Local Accounts	Operating System	Active
Windows Domain Account	Operating System	Active
Windows Server Local Accounts	Operating System	Active
[Sample Password Group Platform]	Mac	Inactive
AS400	Operating System	Inactive
Check Point Firewall-1	Security Appliance	Inactive
Data Phone SSH	Security Appliance	Inactive
GoGo Piv via Telnet	Security Appliance	Inactive
HPLO via SSH	Remote Access	Inactive
HPLO via Telnet	Remote Access	Inactive
IBM Tivoli	Directory	Inactive
Informatica User via SSH	Database	Inactive

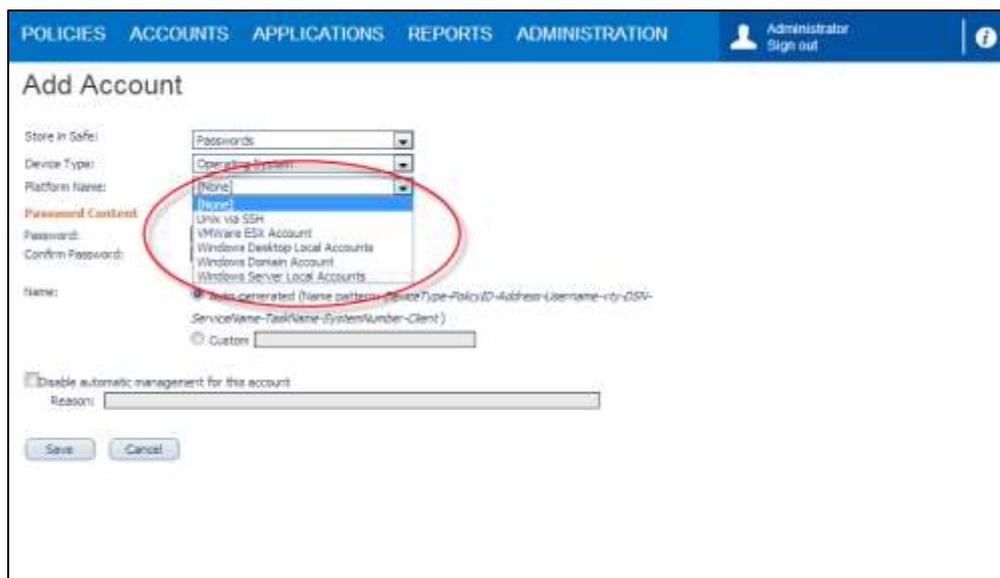
To Set the Status of a Platform

1. In the list of Target Account Platforms, select the platform to mark.
2. In the Platform Preview pane, you can see the current status of the selected platform. To change its status, click the **Edit** icon; the following options appear:
 - **Active** – The platform will be included in the Platform Name drop-down list that is displayed when users create new accounts.
 - **Inactive** – The platform will not be included in the Platform Name drop-down list that is displayed when users create new accounts, and will not be managed by the CPM.



3. Select **Active**, then click the **Save** icon to save the new platform status. The status is also updated in the list of Target Account Platforms.

When users add new accounts, the list of Platform Names for each Device Type will only display the active platforms for that device that can be applied to passwords stored in the selected Safe, as shown in the following example.

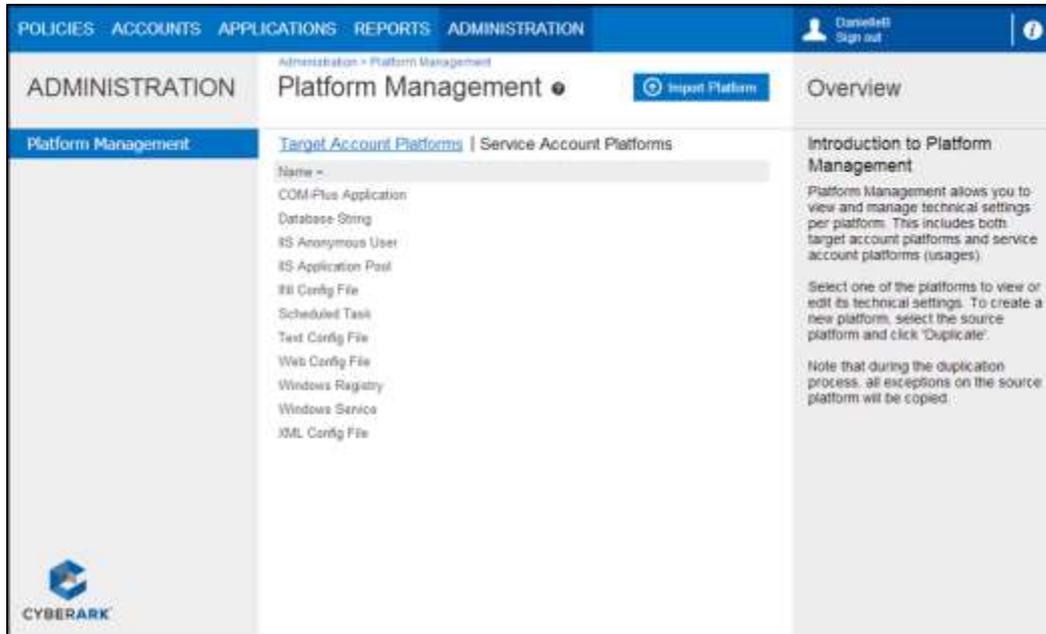


For more information about limiting platforms to passwords stored in specific Safes, refer to *Limit Platforms to Specific Safes*, page 119.

Service Account Platforms

The Service Account platforms define additional service accounts that are required for use in different resources, such as Windows services or Windows scheduled tasks. These service accounts are referred to in the platform parameters. Required and optional properties, and linked accounts are set for each usage.

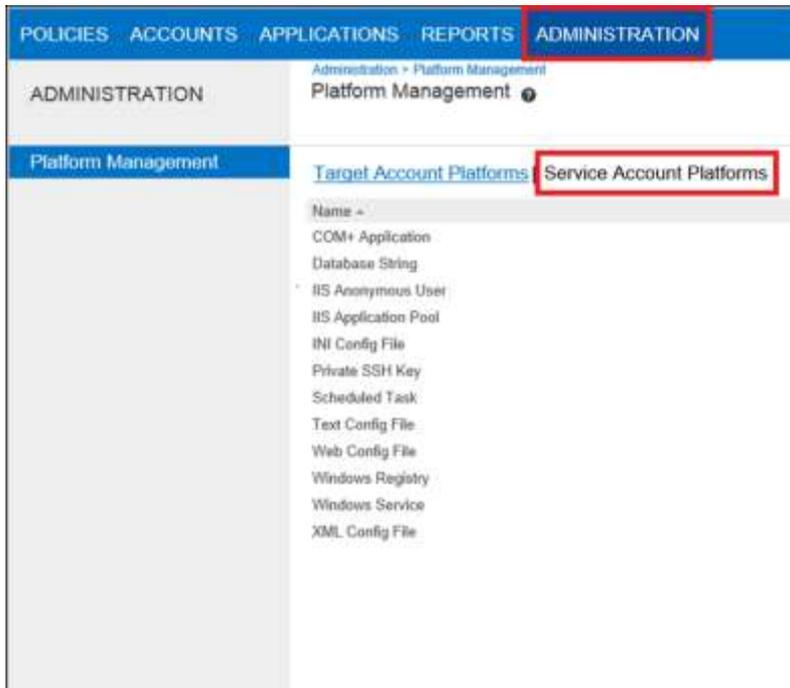
The following example shows the main Service Account Platform page.



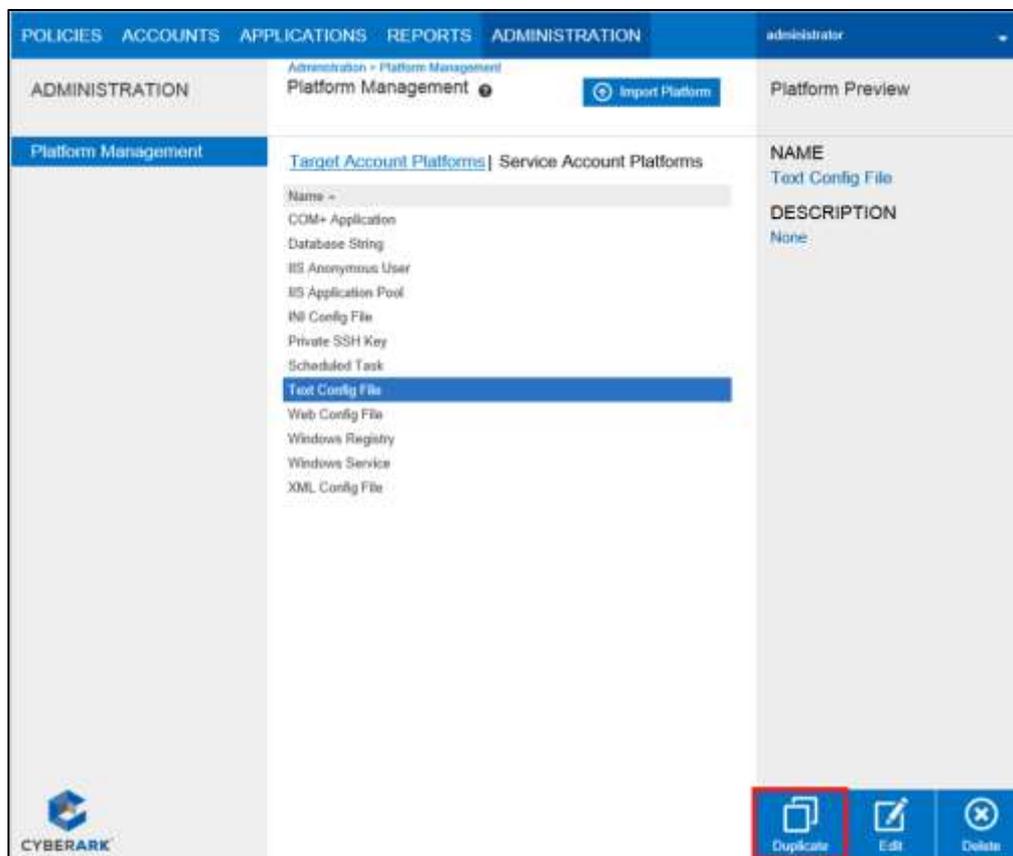
For more information about service accounts, refer to *Managing Service Accounts*, page 151.

Duplicating an Existing Service Account Platform

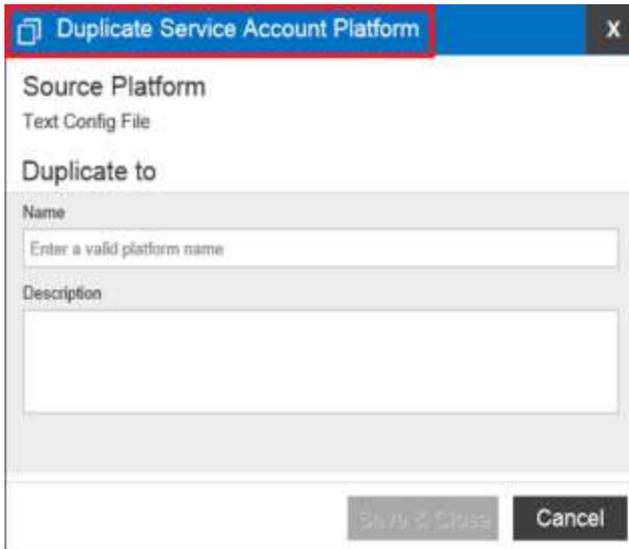
1. Click **ADMINISTRATION** to display the System Configuration page, then click **Platform Management** to display a list of supported service account platforms.



2. Select an existing platform that is similar to the new service account platform, then click **Duplicate**; the Duplicate service account platform window appears.

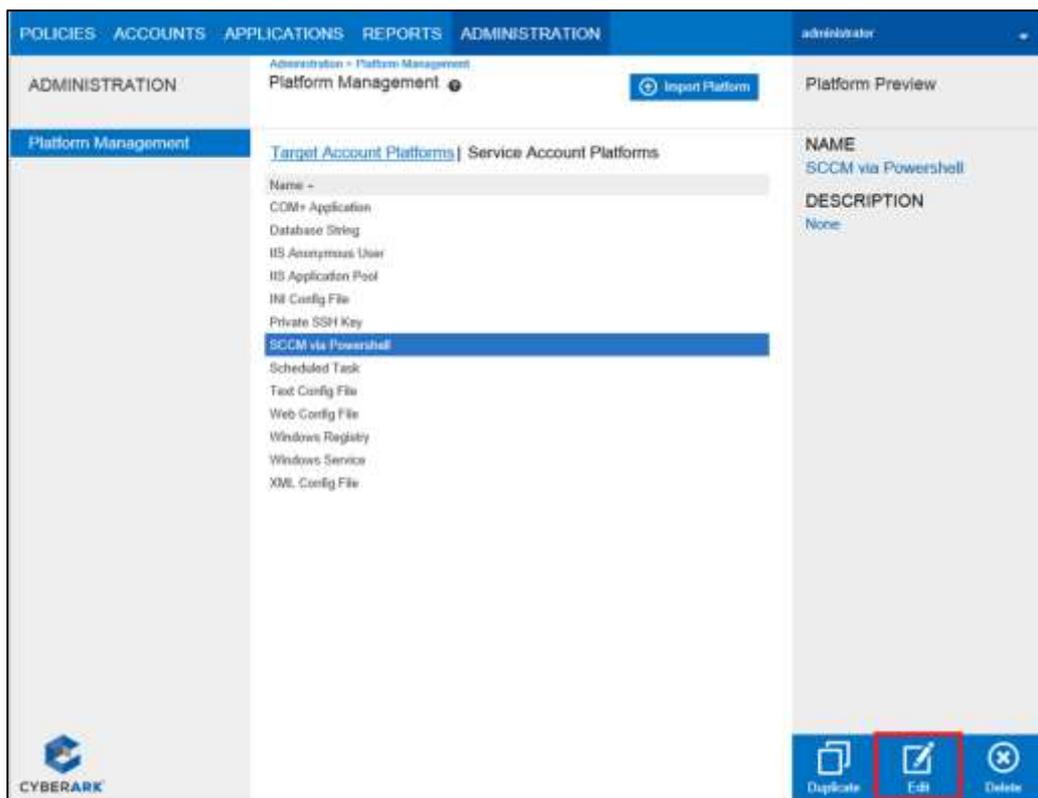


3. Type the name and a description of the new platform, then click **Save & Close** to create the new platform.



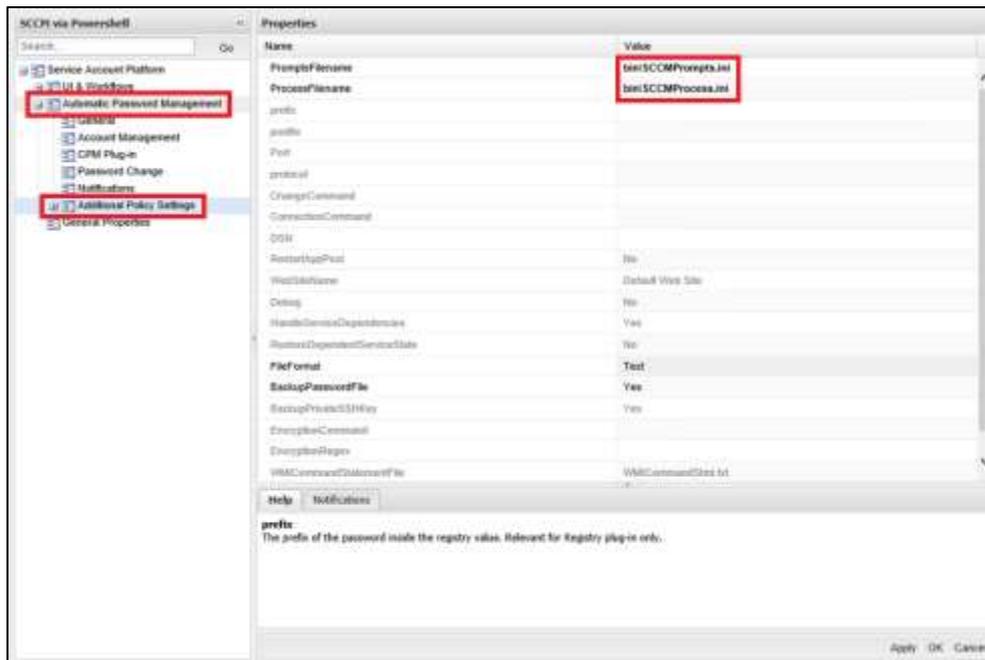
The screenshot shows a dialog box titled "Duplicate Service Account Platform" with a close button (X) in the top right corner. The dialog is divided into sections: "Source Platform" (Text Config File), "Duplicate to", "Name" (with a text input field containing "Enter a valid platform name"), and "Description" (with a larger text area). At the bottom, there are two buttons: "Save & Close" and "Cancel".

4. Select the new service account platform, then click **Edit**; the configuration page for the selected platform appears.



The screenshot displays the "Platform Management" interface. The top navigation bar includes "POLICIES", "ACCOUNTS", "APPLICATIONS", "REPORTS", and "ADMINISTRATION", with the user "administrator" logged in. The main content area is titled "Platform Management" and features a list of "Service Account Platforms". The list includes items like "COM+ Application", "Database String", "IIS Anonymous User", "IIS Application Pool", "INI Config File", "Private SSH Key", "SCCM via Powershell" (which is highlighted), "Scheduled Task", "Text Config File", "Web Config File", "Windows Registry", "Windows Service", and "XML Config File". A "Platform Preview" sidebar on the right shows the details for the selected "SCCM via Powershell" platform, including its "NAME" and "DESCRIPTION". At the bottom right, there are three action buttons: "Duplicate", "Edit" (highlighted with a red box), and "Delete". The CyberArk logo is visible in the bottom left corner.

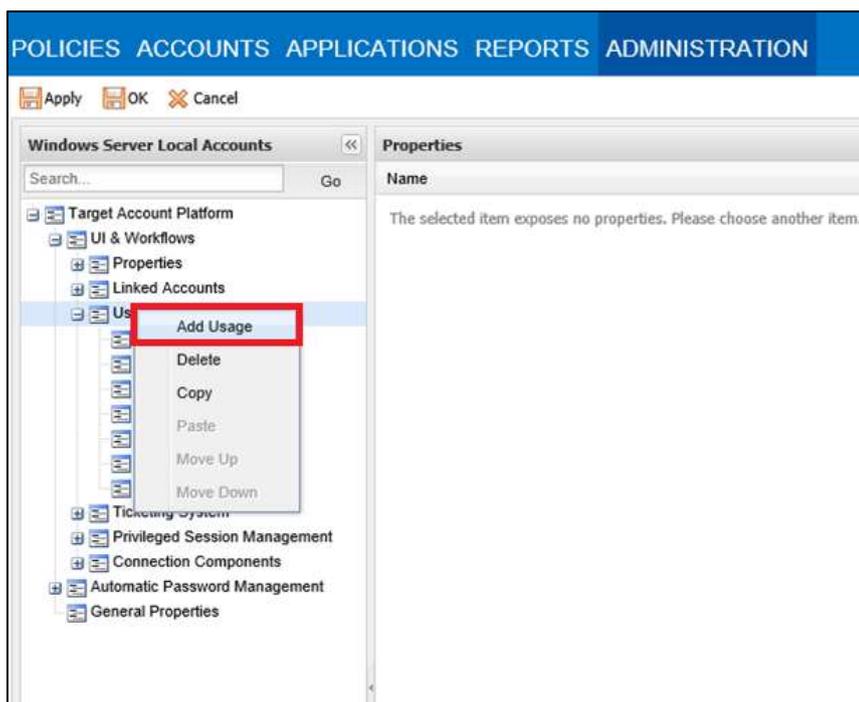
- Change existing parameter values and/or add new values to define the new platform. For example, the names of the process and prompts files.



- Copy the usage plugin files to the Password Manager\bin folder.

Configuring Usage Integration

- Click **ADMINISTRATION** to display the System Configuration page, then click **Platform Management** to display a list of supported target account platforms.
- Select the relevant target account platform, and then click **Edit**; the configuration page for the selected platform appears.
- Expand **UI & Workflow**, right-click **Usages**, then select **Add Usage**; the new usage is now created.



4. In the Name field, specify the name of the policy without white spaces.
5. Click **OK** to save.

Supported Built-in Platforms

The following table lists the platforms that are supported by the Privileged Account Security solution, and their default status:

Active Platforms	Inactive Platforms
Cisco router via SSH Microsoft SQL Server Oracle Database Unix via SSH VMWare ESX Account VMWare vCenter Shared Accounts Windows Desktop Local Accounts Windows Domain Account Windows Server Local Accounts	AS400 Check Point FireWall-1 Cisco Pix via SSH Cisco Pix via Telnet Cisco router via Telnet CyberArk Vault DB2 on Unix via SSH DB2 on Unix via Telnet DB2 on Windows DRAC via SSH Facebook via https HPiLO via SSH HPiLO via Telnet IBM Tivoli Informix on Unix via SSH Informix on Unix via Telnet Informix on Windows MSActiveDirectory MySQL Server NetScreen via SSH NetScreen via Telnet Novell eDirectory server Oracle Internet Directory OS390 via FTP OS390 via SSH OS390 via Telnet PSMSecureConnect SAP SunOne Directory SunOne directory via SSL Sybase ASE Unix Subnet via SSH Unix via SSH Keys Unix via Telnet VMWare ESX Account API VMWarevCenterPersonal Windows Local Accounts WMI

Limit Platforms to Specific Safes

Target account platforms can be restricted to accounts stored in specific Safes. This feature is especially relevant if you implement the reconciliation functionality to prevent automatic reconciliation being performed on every Safe and giving unauthorized users access to passwords.

In large-scale environments, it is very important to enable the CPM to focus its search operations on specific Safes, instead of scanning all Safes it is allowed to see in the Vault.

To Limit a Platform to a Specific Safe

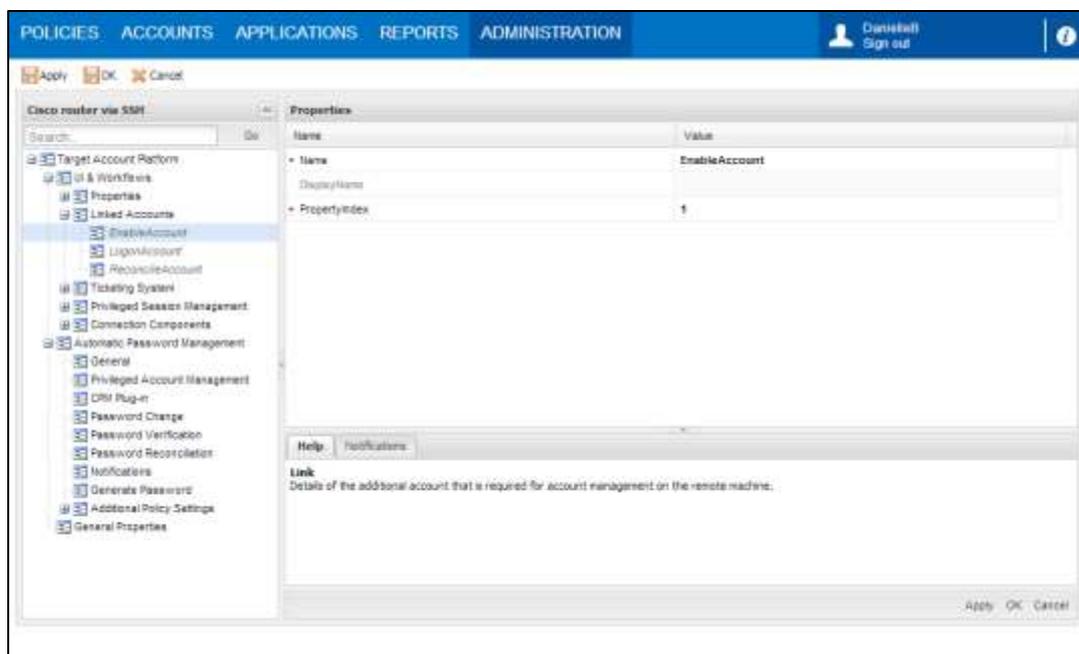
1. In the list of supported target account platforms, select the platform to modify, then click **Edit**; the Target Account platform settings page appears.
2. Expand **Automatic Password Management**, then select **General**; the list of General properties is displayed.
3. In the **AllowedSafes** parameter, specify the name(s) of the Safe(s) where this platform will be used. The default value is `.*`, which allows the platform to be applied in all Safes.
 - For example, to limit this platform to Safes called 'LinuxPasswords' and 'AIXPasswords', specify
`AllowedSafes=(LinuxPasswords) | (AIXPasswords)`
4. Click **Apply** to save the new configurations and apply them immediately,
or,
Click **Save** to save the new configurations and apply them after the period of time specified in the **RefreshPeriod** parameter.

Linked Accounts

The linked accounts feature enables you to specify extra accounts that are required by the CPM to log on to a remote device or to log on as a privileged user to change the original password. For example, a Cisco user can log on to a remote machine as a 'regular' user, and then use the 'enable' account to acquire the authorization required to change the password on the remote machine. The Password Vault supports up to three extra accounts.

Target platform definitions specify the parameters for its linked account(s) in either the Target Account or Service Account that requires them. When the linked accounts are specified in the Target Account platform, they appear in the CPM pane of the Account Details page. When they are specified in the Service Account platform, they appear in the CPM pane of the Service Account Details page.

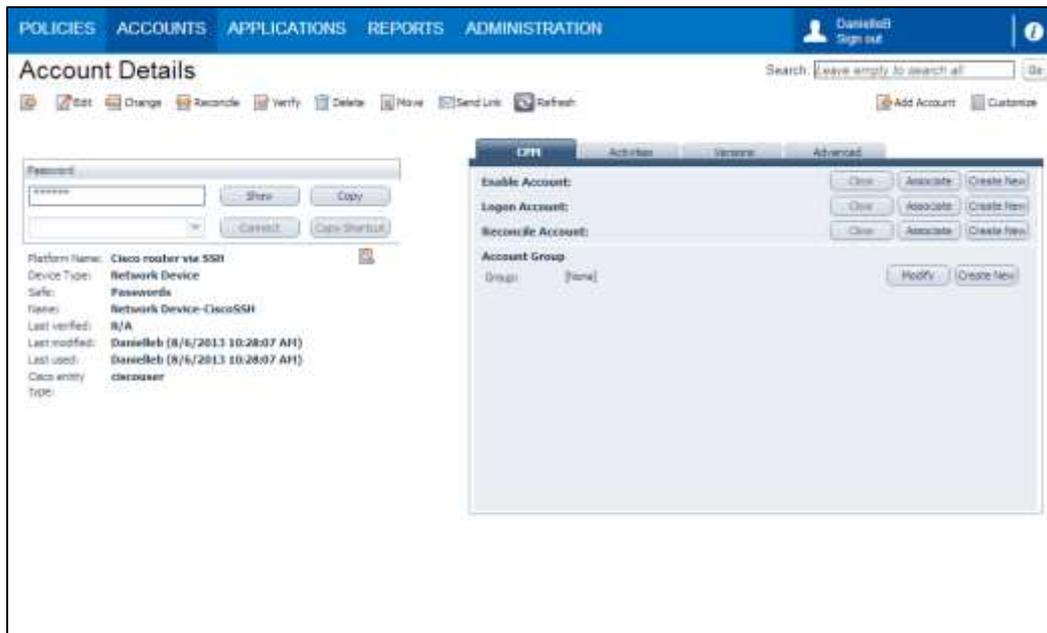
The following example shows the Linked Accounts section in the platform settings page for Cisco SSH passwords.



The Linked Accounts section is in the Target Account platform, indicating that the extra account can be linked from the Account Details page. The platform in the above example offers three types of linked accounts – an Enable account, a Logon account, and a Reconcile account.

Each linked account is defined by three properties, as shown in the Properties list in the above example – the Name of the linked account, the DisplayName, which determines the text that will appear in the PVWA, and the PropertyIndex, which is for internal processing.

As a result of the above example, the PVWA will display the following pane when the user adds a Cisco account.



For more information about adding a linked account to new and existing accounts, refer to *Linked Accounts*, page 220.

Creating Accounts

Adding Accounts

Accounts can be added to the Password Vault in any of the following ways:

- **PVWA** – You can provision accounts individually in the Vault in the Add Accounts page of the PVWA. For more information, refer to *To Add an Account*, below.
- **Accounts Feed** – You can configure the CPM to scan an organizational network and retrieve a list of accounts and their dependencies. For more information, refer to *Accounts Feed*, page 168.

Note: This will replace the auto-detection, which will become obsolete.

- **Provisioning Accounts Automatically (auto-detection)** – You can detect and provision accounts automatically providing a full life-cycle automatic management system for privileged accounts and their dependencies. For more information, refer to *Provisioning Accounts Automatically*, page 154.

Note: This auto-detection functionality will be phased out over time and will be replaced by the Accounts Feed, as described in *Accounts Feed*, page 168.

- **Web Service** – You can provision accounts using the AddAccount web service. For more information, refer to the *Privileged Account Security Web Services SDK Implementation Guide*.
- **Bulk upload** – You can provision multiple accounts with the Password Upload utility. For more information, refer to *Implementing the Password Upload Utility*, page 793.

To Add an Account in the PVWA

1. Click ACCOUNTS to display the Accounts page.
2. Click **Add Account**; the Add Account page appears.

Note: This button will only be displayed if you have the Add accounts, Update password value, or update password properties authorization in at least one Safe.
3. From the Safe drop-down list, select the Safe where the account will be stored.
4. From the Device drop-down list, select the platform on which the new password is used.
5. From the Platform Name drop-down list, select an active target platform.

Required or optional properties for the type of account that you have selected will appear automatically, according to the definitions in the target platform configurations.
6. Specify the required account properties and, if necessary, the optional account properties.

The tables below list the following information for each device supported by the PVWA:

CyberArk Accounts

- **Properties** – The following properties are required to define CyberArk Application accounts:

Parameter	Description
Required Properties	
Platform Name	The platform name that is relevant for this account, and is specified in the platform. The default platform name for CyberArk accounts is CyberArk .
Address	The IP/DNS address, Windows domain or machine name, or TNS name of the remote machine where the password will be used.
User Name	The name of the user on the remote machine who the password belongs to.
Optional Properties	
Port	The Vault IP port. The default port number is 1858.
Timeout	The number of seconds to wait for a Vault to respond to a command before a timeout message is displayed. The default timeout is 30 seconds.
ReconnectPeriod	The number of seconds to wait before the sessions with the Vault is re-established. The default is 60 seconds.
ProxyType	The type of proxy through which the Vault is accessed. Options are HTTP, HTTPS, SOCKS4, SOCKS5, NOPROXY. Default value: NOPROXY.
ProxyAddress	The proxy server's IP/DNS address. This is mandatory when using a proxy server.
ProxyPort	The Proxy server IP port.

Parameter	Description
ProxyAuthDomain	The domain for the Proxy server if NTLM authentication is required.
ProxyUser	User for Proxy server if NTLM authentication is required.
ProxyPassword	The password for Proxy server if NTLM authentication is required.
BehindFirewall	Whether or not the Vault is accessed via a Firewall. Default value: No.
UseOnlyHTTP1	Whether or not to use only HTTP 1.0 protocol. Valid either with proxy settings or with BehindFirewall. Default value: No.

For more information, refer to *CyberArk Vault Accounts*, page 465, in *Setting Up Supported Platforms*.

SAP Accounts

- **Properties** – The following properties are required to define SAP accounts:

Parameter	Description
Required Properties	
Platform Name	The platform name that is relevant for this account, and is specified in the platform. The default platform name for SAP accounts is SAP .
Address	The address of the remote machine where the password will be used.
User Name	The name of the user on the remote machine who the password belongs to.
SAP System Number	The SAP system number.
SAP Client	The SAP Client

For more information, refer to *SAP Applications Accounts*, page 465, in *Setting Up Supported Platforms*.

VMWare vCenter Personal Accounts

This platform is used to define access to vCenter machines by PSM for Virtualization. VMWare vCenter can be configured to work with personal accounts (i.e. end-users' Windows accounts). The accounts defined by this platform describe the vCenter machine.

- **Properties** – The following properties are required to define personal accounts for VMWare vCenter accounts:

Parameter	Description
Required Properties	
Platform Name	The platform name that is relevant for this account, and is specified in the platform. The default platform name for personal accounts for VMWare vCenter environments is VMWarevCenterPersonal .
Address	The address of the remote machine where the password will be used.
Machine Type	The type of machine where the password will be used. By default, a vCenterMachine type is already defined, although you can define new machine types.
Password	Specify a fictitious password when creating this account. Users will be prompted for their actual password when they attempt to log onto the vCenter machine.
Disable automatic management for this account	Select this option to ensure that the CPM does not manage this account.

For more information, refer to *Configuring PSM Connections*, page 621.

VMWare vCenter Shared Accounts

- **Properties** – The following properties are required to define shared accounts for VMWare vCenter accounts:

Parameter	Description
Required Properties	
Platform Name	The platform name that is relevant for this account, and is specified in the platform. The default platform name for personal accounts for VMWare vCenter environments is VMWarevCenterShared .
Address	The address of the remote machine where the password will be used.
User Name	The name of the user on the remote machine who the password belongs to.

Parameter	Description
Optional Properties	
Logon To	The domain of the account. To connect to the remote machine with transparent connection, specify the NETBIOS name of the domain that the logon user belongs to. For example, a domain whose full name is mycompany.com might have the NETBIOS name mycompany_dom , which users would specify in this property. To try to resolve the remote machine's domain automatically, click Resolve ; if the PVWA can identify the remote machine's domain automatically, the domain name will appear in the 'Logon To' field. If not, a message will appear prompting you to specify it manually.
User DN	User's distinguished name.
Port	The port that will be used to access the remote machine.

For more information, refer to *Configuring PSM Connections*, page 621.

DB2 Unix SSH Accounts

- **Properties** – The following properties are required to define DB2 Unix SSH accounts:

Parameter	Description
Required Properties	
Platform Name	The platform name that is relevant for this account, and is specified in the platform. The default platform name for DB2 Unix SSH accounts is DB2 on Unix via SSH .
Address	The address of the remote machine where the password will be used.
User Name	The name of the user on the remote machine who the password belongs to.

- **Additional accounts** – The following additional accounts can be specified for DB2 Unix SSH accounts:

Account	Description
Logon account	An extra account that contains the password that is required to log onto the remote machine. For more information, refer to <i>Linked Accounts</i> , page 220.

For more information, refer to *DB2 Accounts*, page 452, in *Setting Up Supported Platforms*.

DB2 Unix Telnet Accounts

- **Properties** – The following properties are required to define DB2 Unix Telnet accounts:

Parameter	Description
Required Properties	
Platform Name	The platform name that is relevant for this account, and is specified in the platform. The default platform name for DB2 Unix Telnet accounts is DB2UnixTelnet .
Address	The address of the remote machine where the password will be used.
User Name	The name of the user on the remote machine who the password belongs to.

- **Additional accounts** – The following additional accounts can be specified for DB2 Unix SSH accounts:

Account	Description
Logon account	An extra account that contains the password that is required to log onto the remote machine. For more information, refer to <i>Linked Accounts</i> , page 220.

For more information, refer to *DB2 Accounts*, page 452, in *Setting Up Supported Platforms*.

DB2 Windows Accounts

- **Properties** – The following properties are required to define DB2 Windows accounts:

Parameter	Description
Required Properties	
Platform Name	The platform name that is relevant for this account, and is specified in the platform. The default platform name for DB2 Windows accounts is DB2 on Windows .
Address	The address of the remote machine where the password will be used.
User Name	The name of the user on the remote machine who the password belongs to.

For more information, refer to *DB2 Accounts*, page 452, in *Setting Up Supported Platforms*.

Informix Unix SSH Accounts

- **Properties** – The following properties are required to define Informix Unix SSH accounts:

Parameter	Description
Required Properties	
Platform Name	The platform name that is relevant for this account, and is specified in the platform. The default platform name for Informix Unix SSH accounts is Informix on Unix via SSH .
Address	The address of the remote machine where the password will be used.
User Name	The name of the user on the remote machine who the password belongs to.

- **Additional accounts** – The following additional accounts can be specified for Informix Unix SSH accounts:

Account	Description
Logon account	An extra account that contains the password that is required to log onto the remote machine. For more information, refer to <i>Linked Accounts</i> , page 220.

For more information, refer to *Informix Accounts*, page 453, in *Setting Up Supported Platforms*.

Informix Unix Telnet Accounts

- **Properties** – The following properties are required to define Informix Unix Telnet accounts:

Parameter	Description
Required Properties	
Platform Name	The platform name that is relevant for this account, and is specified in the platform. The default platform name for Informix Unix Telnet accounts is Informix on Unix via Telnet .
Address	The address of the remote machine where the password will be used.
User Name	The name of the user on the remote machine who the password belongs to.

- **Additional accounts** – The following additional accounts can be specified for Informix Unix Telnet accounts:

Account	Description
Logon account	An extra account that contains the password that is required to log onto the remote machine. For more information, refer to <i>Linked Accounts</i> , page 220.

For more information, refer to *Informix Accounts*, page 453, in *Setting Up Supported Platforms*.

Informix Windows Accounts

- **Properties** – The following properties are required to define Informix Unix Windows accounts:

Parameter	Description
Required Properties	
Platform Name	The platform name that is relevant for this account, and is specified in the platform. The default platform name for Informix Windows accounts is InformixWindows .
Address	The address of the remote machine where the password will be used.
User Name	The name of the user on the remote machine who the password belongs to.

For more information, refer to *Informix Accounts*, page 453, in *Setting Up Supported Platforms*.

MSSql Accounts

- **Properties** – The following properties are required to define Microsoft SQL Server accounts:

Parameter	Description
Required Properties	
Platform Name	The platform name that is relevant for this password, and is specified in the platform. The default platform name for Microsoft SQL Server passwords is MSSql .
User Name	The name of the user on the remote machine.
Optional Properties	
DSN	The name of the DSN connection that will be used. Use either this parameter or 'ConnectionStringFile'.
Address	The IP address of the remote machine where the password will be used.
Port	The port that will be used to access the remote machine.
Database	The name of the database where the account will be used.

- **Additional accounts** – The following additional accounts can be specified for Microsoft SQL Server accounts:

Account	Description
Reconcile account	An extra account that contains the password used in reconciliation processes. For more information, refer to <i>Reconciling Passwords</i> , page 207.
Windows reconcile account	Whether the reconcile account is a Microsoft Windows account or an SQL account.

For more information, refer to *Microsoft SQL Server Accounts*, page 437, in *Setting Up Supported Platforms*.

Oracle Accounts

- **Properties** – The following properties are required to define Oracle accounts:

Parameter	Description
Required Properties	
Platform Name	The platform name that is relevant for this password, and is specified in the platform. The default platform name for Oracle passwords is Oracle .
User Name	The name of the user on the remote machine.
Optional Properties	
DSN	The name of the DSN connection that will be used. Use either this parameter or 'ConnectionStringFile'.
Address	The IP address of the remote machine where the password will be used.
Port	The port that will be used to access the remote machine.
Database	The name of the database where the account will be used.

- **Additional accounts** – The following additional accounts can be specified for Oracle accounts:

Account	Description
Reconcile account	An extra account that contains the password used in reconciliation processes. For more information, refer to <i>Reconciling Passwords</i> , page 207.

For more information, refer to *Oracle Database Accounts*, page 431, in *Setting Up Supported Platforms*.

Sybase Accounts

- **Properties** – The following properties are required to define Sybase accounts:

Parameter	Description
Required Properties	
Platform Name	The platform name that is relevant for this password, and is specified in the platform. The default platform name for Sybase passwords is Sybase .
User Name	The name of the user on the remote machine.
Optional Properties	
DSN	The name of the DSN connection that will be used. Use either this parameter or 'ConnectionStringFile'.
Address	The IP address of the remote machine where the password will be used.
Port	The port that will be used to access the remote machine.
Database	The name of the database where the account will be used.

- **Additional accounts** – The following additional accounts can be specified for Sybase accounts:

Account	Description
Reconcile account	An extra account that contains the password used in reconciliation processes. For more information, refer to <i>Reconciling Passwords</i> , page 207.

For more information, refer to *Sybase Database Accounts*, page 442, in *Setting Up Supported Platforms*.

Novell eDirectory Accounts

- **Properties** – The following properties are required to define Novell eDirectory accounts:

Parameter	Description
Required Properties	
Platform Name	The platform name that is relevant for this password, and is specified in the platform. The default platform name for Novell eDirectory passwords is Novell-eDirectory .
Address	The IP address of the remote machine where the password will be used.
Optional Properties	
Port	The port that will be used to access the remote machine.
UserDN	The distinguished name of the user.

- **Additional accounts** – The following additional accounts can be specified for Novell eDirectory accounts:

Account	Description
Reconcile account	An extra account that contains the password used in reconciliation processes. For more information, refer to <i>Reconciling Passwords</i> , page 207.

For more information, refer to *Novell eDirectory Accounts*, page 463, in *Setting Up Supported Platforms*.

SunOne Directory Accounts

- **Properties** – The following properties are required to define SunOne Directory accounts:

Parameter	Description
Required Properties	
Platform Name	The platform name that is relevant for this password, and is specified in the platform. The default platform name for SunOne Directory passwords is SunOneDirectory .
Address	The IP address of the remote machine where the password will be used.
Optional Properties	
Port	The port that will be used to access the remote machine.
UserDN	The distinguished name of the user.

- **Additional accounts** – The following additional accounts can be specified for SunOne Directory accounts:

Account	Description
Reconcile account	An extra account that contains the password used in reconciliation processes. For more information, refer to <i>Reconciling Passwords</i> , page 207.

For more information, refer to *SunOne Directory Accounts*, page 464, in *Setting Up Supported Platforms*.

SunOne Directory SSL Accounts

- **Properties** – The following properties are required to define SunOne Directory SSL accounts:

Parameter	Description
Required Properties	
Platform Name	The platform name that is relevant for this password, and is specified in the platform. The default platform name for SunOne Directory SSL passwords is SunOneDirectorySSL .
Address	The IP address of the remote machine where the password will be used.
Optional Properties	
Port	The port that will be used to access the remote machine.
UserDN	The distinguished name of the user.

- **Additional accounts** – The following additional accounts can be specified for SunOne Directory SSL accounts:

Account	Description
Reconcile account	An extra account that contains the password used in reconciliation processes. For more information, refer to <i>Reconciling Passwords</i> , page 207.

For more information, refer to *SunOne Directory Accounts*, page 464, in *Setting Up Supported Platforms*.

Cisco SSH Accounts

- **Properties** – The following properties are required to define Cisco SSH accounts:

Parameter	Description
Required Properties	
Platform Name	The platform name that is relevant for this password, and is specified in the platform. The default platform name for Cisco SSH passwords is CiscoSSH .
Type	The type of password to use. Specify one of the following: <ul style="list-style-type: none"> ▪ ciscouser ▪ ciscoenable ▪ ciscoterminal
Optional Properties	
User Name	The name of the user on the router that this password belongs to. Specify one of the following: <ul style="list-style-type: none"> ▪ ciscouser – the name of the user on the PIX machine. ▪ ciscoenable – nothing
Address	The IP address of the remote machine where the password will be used.
Port	The port that will be used to access the router.
vt	The virtual terminal line that will connect to the router.

- **Additional accounts** – The following additional accounts can be specified for Cisco SSH accounts:

Account	Description
Enable account	An extra account that contains the password that will enable the CPM to switch to 'enable' mode and change the password on the remote machine.
Logon account	An extra account that contains the password that contains logon information that will enable the CPM to log onto the remote machine where the password will be changed.

For more information, refer to *Cisco Router Accounts*, page 459, in *Setting Up Supported Platforms*.

Cisco Telnet Accounts

- **Properties** – The following properties are required to define Cisco Telnet accounts:

Parameter	Description
Required Properties	
Platform Name	The platform name that is relevant for this password, and is specified in the platform. The default platform name for Cisco Telnet passwords is CiscoTelnet .
Type	The type of password to use. Specify one of the following: <ul style="list-style-type: none"> ◆ ciscouser ◆ ciscoenable ◆ ciscoterminal
Optional Properties	
User Name	The name of the user on the router that this password belongs to. Specify one of the following: <ul style="list-style-type: none"> ◆ ciscouser – the name of the user on the PIX machine. ◆ ciscoenable – nothing
Address	The IP address of the remote machine where the password will be used.
Port	The port that will be used to access the router.
vt	The virtual terminal line that will connect to the router.

- **Additional accounts** – The following additional accounts can be specified for Cisco SSH accounts:

Account	Description
Enable account	An extra account that contains the password that will enable the CPM to switch to 'enable' mode and change the password on the remote machine.
Logon account	An extra account that contains the password that contains logon information that will enable the CPM to log onto the remote machine where the password will be changed.

For more information, refer to *Cisco Router Accounts*, page 459, in *Setting Up Supported Platforms*.

AS400 (iSeries) Accounts

- **Properties** – The following properties are required to define AS400 (iSeries) accounts:

Parameter	Description
Required Properties	
Platform Name	The platform name that is relevant for this password, and is specified in the platform. The default platform name for as400 passwords is as400 .
Address	The IP address of the remote machine where the password will be used.
User Name	The name of the user on the remote machine who this password belongs to.
Optional Properties	
AS400 Account Type	The type of the AS400 (iSeries) account. Specify one of the following: <ul style="list-style-type: none"> ♦ RegularUserProfile – An account type for regular OS users (default). ♦ ServiceToolUser – An account type for either Dedicated Service Tools (DST) users or System Service Tools (SST) users.

- **Additional accounts** – The following additional accounts can be specified for AS400 (iSeries) accounts:

Account	Description
Logon account	An extra account that contains the password that is required to log onto the remote machine for Service Tools accounts. For more information, refer to <i>Linked Accounts</i> , page 220. This account must be defined as a RegularUserProfile type account.
Reconcile account	An extra account that contains the password used in reconciliation processes for Service Tools accounts. For more information, refer to <i>Reconciling Passwords</i> , page 207. This account must be defined as the same type as the main account type.

For more information, refer to *AS400 (iSeries) Accounts*, page 420, in *Setting Up Supported Platforms*.

OS/390 (Z/OS) FTP Accounts

- **Properties** – The following properties are required to define OS/390 (Z/OS) FTP accounts:

Parameter	Description
Required Properties	
Platform Name	The platform name that is relevant for this password, and is specified in the platform. The default platform name for OS/390 (Z/OS) FTP passwords is OS390FTP .
Address	The IP address of the remote machine where the password will be used.
User Name	The name of the user on the remote machine who this password belongs to.

For more information, refer to *OS/390 (Z/OS) Accounts*, page 423, in *Setting Up Supported Platforms*.

OS/390 (Z/OS) SSH Accounts

- **Properties** – The following properties are required to define OS/390 (Z/OS) SSH accounts:

Parameter	Description
Required Properties	
Platform Name	The platform name that is relevant for this password, and is specified in the platform. The default platform name for OS/390 (Z/OS) SSH passwords is OS390SSH .
Address	The IP address of the remote machine where the password will be used.
User Name	The name of the user on the remote machine who this password belongs to.

- **Additional accounts** – The following additional accounts can be specified for OS/390 (Z/OS) SSH accounts:

Account	Description
Logon account	An extra account that contains the password that is required to log onto the remote machine. For more information, refer to <i>Linked Accounts</i> , page 220.

For more information, refer to *OS/390 (Z/OS) Accounts*, page 423, in *Setting Up Supported Platforms*.

OS/390 (Z/OS) Telnet Accounts

- **Properties** – The following properties are required to define OS/390 (Z/OS) Telnet accounts:

Parameter	Description
Required Properties	
Platform Name	The platform name that is relevant for this password, and is specified in the platform. The default platform name for OS/390 (Z/OS) Telnet passwords is OS390Telnet .
Address	The IP address of the remote machine where the password will be used.
User Name	The name of the user on the remote machine who this password belongs to.

- **Additional accounts** – The following additional accounts can be specified for OS/390 (Z/OS) Telnet accounts:

Account	Description
Logon account	An extra account that contains the password that is required to log onto the remote machine. For more information, refer to <i>Linked Accounts</i> , page 220.

For more information, refer to *OS/390 (Z/OS) Accounts*, page 423, in *Setting Up Supported Platforms*.

ESX/i Accounts

- **Properties** – The following properties are required to define ESX/i accounts:

Parameter	Description
Required Properties	
Platform Name	The platform name that is relevant for this account, and is specified in the platform. The default platform name for ESX/i accounts is VMWareESX-API .
Address	The address of the remote machine where the password will be used.
User Name	The name of the user on the remote machine who this password belongs to. Specify a local ESX/ESX/i account or 'root'.

- **Additional accounts** – The following additional accounts can be specified for OS/390 (Z/OS) Telnet accounts:

Account	Description
Logon account	An extra account that contains the password that is required to log onto the remote machine. This must also be an ESX/i local or root account. For more information, refer to <i>Linked Accounts</i> , page 220.
Reconcile account	An extra account that contains the password used in reconciliation processes. This must also be an ESX/i local or root account. For more information, refer to <i>Reconciling Passwords</i> , page 207.

For more information, refer to *ESX/i Accounts*, page 424, in *Setting Up Supported Platforms*.

Unix SSH Accounts

- **Properties** – The following properties are required to define Unix SSH accounts:

Parameter	Description
Required Properties	
Platform Name	The platform name that is relevant for this password, and is specified in the platform. The default platform name for Unix SSH passwords is Unix via SSH .
Address	The IP address of the remote machine where the password will be used.
User Name	The name of the user on the remote machine who this password belongs to.

- **Additional accounts** – The following additional accounts can be specified for Unix SSH accounts:

Account	Description
Logon account	An extra account that contains the password that is required to log onto the remote machine. For more information, refer to <i>Linked Accounts</i> , page 220.
Reconcile account	An extra account that contains the password used in reconciliation processes. For more information, refer to <i>Reconciling Passwords</i> , page 207.

For more information, refer to *Unix Accounts*, page 419, in *Setting Up Supported Platforms*.

Unix Accounts with SSH Keys

- **Properties** – The following properties are required to define Unix accounts with SSH keys:

Parameter	Description
Required Properties	
Platform Name	The platform name that is relevant for this account, and is specified in the platform. The default platform name for Unix accounts with SSH Keys is Unix via SSH Keys .
Address	The IP address of the remote machine where the private SSH key will be used together with a public SSH key stored on that machine.
User Name	The name of the user on the remote machine who is authorized to use the private SSH key.
SSH Key	The content of the private SSH key. This can be specified as either a key file or as the actual key content.

- **Additional accounts** – The following additional accounts can be specified for Unix accounts with SSH keys:

Account	Description
Reconcile account	An extra account that contains the password or SSH Key used in reconciliation processes. For more information, refer to <i>Reconciling Passwords</i> , page 207.

You can also add accounts using the AccountUploader utility. For more information, refer to *Appendix E: Adding Accounts with SSH Keys using the AccountUploader Utility*, page 984.

For more information, refer to *Unix Accounts*, page 419, in *Setting Up Supported Platforms*.

Unix Telnet Accounts

- **Properties** – The following properties are required to define Unix Telnet accounts:

Parameter	Description
Required Properties	
Platform Name	The platform name that is relevant for this password, and is specified in the platform. The default platform name for Unix Telnet passwords is UnixTelnet .
Address	The IP address of the remote machine where the password will be used.
User Name	The name of the user on the remote machine who this password belongs to.

For more information, refer to *Unix Accounts*, page 419, in *Setting Up Supported Platforms*.

Unix Domain/NIS Accounts

- **Properties** – The following properties are required to define Unix Domain/NIS accounts:

Parameter	Description
Required Properties	
Platform Name	The platform name that is relevant for this password, and is specified in the platform. This platform is not predefined and must be configured manually. For more information about using these accounts in PSM connections, refer to <i>UNIX Domain/NIS</i> , page 632. For more information about using these accounts in PSMP connections, refer to <i>Configuring UNIX Domain/NIS Accounts</i> , page 716.
Address	The domain name of the machine where the password will be used. This can either be specified as an IP address or as a Fully Qualified Domain Name (FQDN). For example, mycompany.com.
User Name	The name of the domain user who can access the machine where the password will be used.

VMWare ESX Accounts

- **Properties** – The following properties are required to define for ESX accounts:

Parameter	Description
Required Properties	
Platform Name	The platform name that is relevant for this account, and is specified in the platform. The default platform name for personal accounts for VMWare ESX environments is VMWareESX .
Address	The address of the remote machine where the password will be used.
User Name	The name of the user on the remote machine who the password belongs to.

For more information, refer to *Configuring PSM Connections*, page 621.

Windows Domain Accounts

- **Properties** – The following properties are required to define Windows Domain accounts:

Parameter	Description
Required Properties	
Platform Name	The platform name that is relevant for this password, and is specified in the platform. The default platform name for Windows Domain Accounts is WinDomain .
Address	The Windows domain name of the remote machine where the password will be used. This can be specified as a Fully Qualified Domain Name (FQDN). For example, mycompany.com.
User Name	The name of the user on the remote machine.
Optional Properties	
Logon To	The domain where the account will be used. Note: To connect to a remote machine with a transparent connection, specify the FQDN name of the domain that the logon user belongs to. For example, mycompany.com. This replaces the domain's NETBIOS name.
User DN	User's distinguished name.
Port	The port that will be used to access the remote machine.

For more information, refer to *Windows Domain Accounts*, page 384, in *Setting Up Supported Platforms*.

- **Additional accounts** – The following additional accounts can be specified for Windows Domain accounts:

Account	Description
Logon account	An extra account that contains the password that is required to log onto the remote machine. For more information, refer to <i>Linked Accounts</i> , page 220.
Reconcile account	An extra account that contains the password used in reconciliation processes. For more information, refer to <i>Reconciling Passwords</i> , page 207.

- **Multiple copies of accounts** – Multiple copies of Windows domain accounts can be synchronized and used in the following different resources.

Parameter	Description
Windows Services	A Windows domain account password can be synchronized with multiple copies of the same password used in different services, after it has been changed successfully.
Windows Scheduled Tasks	A Windows domain accounts can be synchronized with other occurrences of the same password in different Windows scheduled tasks, after it has been changed successfully.
Windows IIS Pools	A Windows domain account password can be synchronized with multiple copies of the same password used in Windows IIS Application Pools, after it has been changed successfully.
Windows COM+ Applications	A Windows domain account password can be synchronized with multiple copies of the same password used in Windows COM+ applications, after it has been changed successfully.
Windows IIS Directory Security (Anonymous Access)	A Windows domain account password can be synchronized with multiple copies of the same password used in IIS Directory Security with Anonymous Access definition, after it has been changed successfully.

For more information, refer to *Managing Service Accounts*, page 151.

Windows Local Accounts

- **Properties** – The following properties are required to define Windows Local accounts:

Parameter	Description
Required Properties	
Platform Name	The platform name that is relevant for this password, and is specified in the platform. The default platform name for Windows local accounts is WinServerLocal .
Address	The network name or IP address of the remote machine where the password will be used.
User Name	The name of the user on the remote machine who this password belongs to.

Parameter	Description
Optional Properties	
Logon To	The domain of the account. If you intend to connect to the remote machine with transparent connection, specify the NETBIOS name of the domain that the logon user belongs to. For example, a domain whose full name is mycompany.com might have the NETBIOS name mycompany_dom , which users would specify in this property. To try to resolve the remote machine's domain automatically, click Resolve ; if the PVWA can identify the remote machine's domain automatically, the domain name will appear in the 'Logon To' field. If not, a message will appear prompting you to specify it manually.
User DN	User's distinguished name.
Port	The port that will be used to access the remote machine.

For more information, refer to *Windows Local Accounts*, page 386, in *Setting Up Supported Platforms*.

- **Additional accounts** – The following additional accounts can be specified for Windows local accounts:

Account	Description
Logon account	An extra account that contains the password that is required to log onto the remote machine. For more information, refer to <i>Linked Accounts</i> , page 220.
Reconcile account	An extra account that contains the password used in reconciliation processes. For more information, refer to <i>Reconciling Passwords</i> , page 207.

- **Multiple copies of accounts** – Multiple copies of Windows local accounts can be synchronized and used in the following different resources.

Parameter	Description
Windows Services	A Windows local account password can be synchronized with multiple copies of the same password used in different services, after it has been changed successfully.
Windows Scheduled Tasks	A Windows local account can be synchronized with other occurrences of the same password in different Windows scheduled tasks, after it has been changed successfully.
Windows IIS Pools	A Windows local account password can be synchronized with multiple copies of the same password used in Windows IIS Application Pools, after it has been changed successfully.
Windows COM+ Applications	A Windows local account password can be synchronized with multiple copies of the same password used in Windows COM+ applications, after it has been changed successfully.
Windows IIS Directory Security (Anonymous Access)	A Windows local account password can be synchronized with multiple copies of the same password used in IIS Directory Security with Anonymous Access definition, after it has been changed successfully.

For more information, refer to *Managing Service Accounts*, page 151.

Windows Local Accounts with WMI

- **Properties** – The following properties are required to define Windows Local accounts with WMI:

Parameter	Description
Required Properties	
Platform Name	The platform name that is relevant for this password, and is specified in the platform. The default platform name for Windows Local Accounts with WMI is WinLocalWMI .
Address	The IP/DNS address, Windows domain or machine name, or TNS name of the remote machine where the password will be used.
User Name	The name of the user on the remote machine.
Optional Properties	
LogonDomain	The domain where the account will be used.
Location	The physical location of the Windows machine.
OwnerName	The full name of the desktop owner.

For more information, refer to *Windows Local Accounts with WMI*, page 388, in *Setting Up Supported Platforms*.

- **Additional accounts** – The following additional accounts can be specified for Windows Local Desktop accounts:

Account	Description
Logon account	An extra account that contains the password that is required to log onto the remote machine. For more information, refer to <i>Linked Accounts</i> , page 220.
Reconcile account	An extra account that contains the password used in reconciliation processes. For more information, refer to <i>Reconciling Passwords</i> , page 207. By default, a platform reconcile account is configured, although you can override it by associating a different reconcile account in the Account Details page.

- **Multiple copies of accounts** – Multiple copies of Windows local desktop accounts can be synchronized and used in the following different resources.

Parameter	Description
Windows Services	A Windows local account password can be synchronized with multiple copies of the same password used in different services, after it has been changed successfully.
Windows Scheduled Tasks	A Windows local account password can be synchronized with other occurrences of the same password in different Windows scheduled tasks, after it has been changed successfully.
Windows IIS Pools	A Windows local account password can be synchronized with multiple copies of the same password used in Windows IIS Application Pools, after it has been changed successfully.

Parameter	Description
Windows Registry	A Windows local account password can be synchronized with multiple copies of the same password used in different registries, after it has been changed successfully.
Windows COM+ Applications	A Windows local account password can be synchronized with multiple copies of the same password used in Windows COM+ applications, after it has been changed successfully.
Windows IIS Directory Security (Anonymous Access)	A Windows local account password can be synchronized with multiple copies of the same password used in IIS Directory Security with Anonymous Access definition, after it has been changed successfully.

For more information, refer to *Managing Service Accounts*, page 151.

Windows Local Desktop Accounts

- **Properties** – The following properties are required to define Windows Local Desktop accounts:

Parameter	Description
Required Properties	
Platform Name	The platform name that is relevant for this password, and is specified in the platform. The default platform name for Windows Local Desktop Accounts is WinDesktopLocal .
Address	The IP/DNS address, Windows domain or machine name, or TNS name of the remote machine where the password will be used.
User Name	The name of the user on the remote machine.
Optional Properties	
LogonDomain	The domain where the account will be used.
Location	The physical location of the Windows machine.
OwnerName	The full name of the desktop owner.

For more information, refer to *Windows Local Accounts*, page 386, in *Setting Up Supported Platforms*.

- **Additional accounts** – The following additional accounts can be specified for Windows Local Desktop accounts:

Account	Description
Logon account	An extra account that contains the password that is required to log onto the remote machine. For more information, refer to <i>Linked Accounts</i> , page 220.
Reconcile account	An extra account that contains the password used in reconciliation processes. For more information, refer to <i>Reconciling Passwords</i> , page 207.

- **Multiple copies of accounts** – Multiple copies of Windows local desktop accounts can be synchronized and used in the following different resources.

Parameter	Description
Windows Services	A Windows local desktop account password can be synchronized with multiple copies of the same password used in different services, after it has been changed successfully.
Windows Scheduled Tasks	A Windows local desktop account password can be synchronized with other occurrences of the same password in different Windows scheduled tasks, after it has been changed successfully.
Windows IIS Pools	A Windows local desktop account password can be synchronized with multiple copies of the same password used in Windows IIS Application Pools, after it has been changed successfully.
Windows COM+ Applications	A Windows local desktop account password can be synchronized with multiple copies of the same password used in Windows COM+ applications, after it has been changed successfully.
Windows IIS Directory Security (Anonymous Access)	A Windows local desktop account password can be synchronized with multiple copies of the same password used in IIS Directory Security with Anonymous Access definition, after it has been changed successfully.

For more information, refer to *Managing Service Accounts*, page 151.

Cisco PIX SSH Accounts

- **Properties** – The following properties are required to define Cisco PIX SSH accounts:

Parameter	Description
Required Properties	
platform name	The platform name that is relevant for this password, and is specified in the platform. The default platform name for Cisco PIX SSH passwords is CiscoPixSSH .
Address	The IP address of the remote machine where the password will be used.
Type	The type of password to use. Specify one of the following: <ul style="list-style-type: none"> ◆ ciscouser ◆ ciscoenable ◆ ciscoterminal
Optional Properties	
User Name	The name of the user on the router that this password belongs to. Specify one of the following: <ul style="list-style-type: none"> ◆ ciscouser – the name of the user on the PIX machine. ◆ ciscoenable – nothing
Port	The port that will be used to access the router.

- **Additional accounts** – The following additional accounts can be specified for Cisco PIX SSH accounts:

Account	Description
Enable account	An extra account that contains the password that will enable the CPM to switch to 'enable' mode and change the password on the remote machine.
Logon account	An extra account that contains the password that contains logon information that will enable the CPM to log onto the remote machine where the password will be changed.

For more information, refer to *Cisco PIX Accounts*, page 461, in *Setting Up Supported Platforms*.

Cisco PIX Telnet Accounts

- **Properties** – The following properties are required to define Cisco PIX Telnet accounts:

Parameter	Description
Required Properties	
Platform Name	The platform name that is relevant for this password, and is specified in the platform. The default platform name for Cisco PIX Telnet passwords is CiscoPixTelnet .
Address	The IP address of the remote machine where the password will be used.
Type	The type of password to use. Specify one of the following: <ul style="list-style-type: none"> ◆ ciscouser ◆ ciscoenable ◆ ciscoterminal
Optional Properties	
User Name	The name of the user on the router that this password belongs to. Specify one of the following: <ul style="list-style-type: none"> ◆ ciscouser – the name of the user on the PIX machine. ◆ ciscoenable – nothing
Port	The port that will be used to access the router.

- **Additional accounts** – The following additional accounts can be specified for Cisco PIX Telnet accounts:

Account	Description
Enable account	An extra account that contains the password that will enable the CPM to switch to 'enable' mode and change the password on the remote machine.
Logon account	An extra account that contains the password that contains logon information that will enable the CPM to log onto the remote machine where the password will be changed.

For more information, refer to *Cisco PIX Accounts*, page 461, in *Setting Up Supported Platforms*.

CheckPoint Firewall-1 Accounts

- **Properties** – The following properties are required to define CheckPoint Firewall-1 accounts:

Parameter	Description
Required Properties	
Platform Name	The platform name that is relevant for this password, and is specified in the platform. The default platform name for CheckPoint Firewall-1 passwords is Firewall1 .
Address	The IP address of the remote machine where the password will be used.
User Name	The name of the user on the remote machine who this password belongs to.
ClientDN	The distinguished name of the client entity.
ServerDN	The distinguished name of the SmartCenter module.
Optional Properties	
SicCertFile	The path and name of the sic certification file. Default: opsec.p12 which should be placed in the Password Manager Bin directory.
Port	The port that will be used to access the router.

For more information, refer to *CheckPoint Firewall-1 NG Accounts*, page 456, in *Setting Up Supported Platforms*.

NetScreen SSH Accounts

- **Properties** – The following properties are required to define NetScreen SSH accounts:

Parameter	Description
Required Properties	
Platform Name	The platform name that is relevant for this password, and is specified in the platform. The default platform name for NetScreen SSH passwords is NetScreenSSH .
Address	The IP address of the remote machine where the password will be used.
User Name	The name of the user on the remote machine who this password belongs to.

For more information, refer to *NetScreen Firewall Accounts*, page 458, in *Setting Up Supported Platforms*.

NetScreen Telnet Accounts

- **Properties** – The following properties are required to define OS/390 (Z/OS) Telnet accounts:

Parameter	Description
Required Properties	
Platform Name	The platform name that is relevant for this password, and is specified in the platform. The default platform name for NetScreen Telnet passwords is NetScreenTelnet .
Address	The IP address of the remote machine where the password will be used.
User Name	The name of the user on the remote machine who this password belongs to.

For more information, refer to *NetScreen Firewall Accounts*, page 458, in *Setting Up Supported Platforms*.

Facebook Accounts

- **Properties** – The following properties are required to define Facebook accounts:

Parameter	Description
Required Properties	
Platform Name	The platform name that is relevant for this account, and is specified in the platform. The default platform name for Facebook accounts is Facebook .
Address	The address of Facebook's website, www.facebook.com. This address appears by default.
User Name	The name of the Facebook user who the password belongs to.

For more information, refer to *Facebook Accounts*, page 479, in *Setting Up Supported Platforms*.

Amazon Web Services (AWS) Accounts

- **Properties** – The following properties are required to define Amazon Web Services (AWS) accounts:

Parameter	Description
Required Properties	
Platform Name	The platform name that is relevant for this account, and is specified in the platform. The default platform name for Amazon Web Services (AWS) accounts is Amazon Web Services (AWS) .
Address	The address of the Amazon Web Services (AWS) website, www.AWS.com. This address appears by default.
User Name	The name of the AWS user who the password belongs to.

For more information, refer to *Amazon Web Services (AWS) Accounts*, page 491, in *Setting Up Supported Platforms*.

7. In the **Password** field, specify the password. Make sure this password meets your enterprise password policy requirements.
8. In the **Confirm Password** field, specify the password again.
9. To generate a password name automatically, select **Auto-generated**. For more information about naming passwords automatically, refer to *Identifying Accounts*, page 508.

To specify a password name, enter the name in the **Custom** field.

10. To disable automatic password management by the CPM for this password so that it will be managed manually, select **Disable automatic management for the account**. You can also enter a reason for doing this.

Note: The CPM user must be an owner of the Safe where the password will be stored and a platform name of an active target account platform must be specified in order for the password to be managed by the CPM.

11. Click **Save**; the new account is added.

If the PWA is configured to automatically change or verify passwords when they are added, this will be done now. For more information about configuring this feature, refer to *Adding Accounts*, page 570, in *Configuring the PVWA*.

The account is now created in the specified Safe and the new account details are displayed in the Account Details page. If the specified password contains leading and/or trailing white space character(s), a message appears in the Account Details page indicating that they will automatically be removed.

The screenshot displays the 'Account Details' page. The top navigation bar includes 'POLICIES', 'ACCOUNTS', 'APPLICATIONS', 'REPORTS', and 'ADMINISTRATION'. The user 'DanielleB' is logged in. The page title is 'Account Details'. Below the title are several action buttons: Edit, Change, Records, Verify, Delete, Move, Send Link, Refresh, Add Account, and Customize. The main content area is divided into two sections. The left section shows the account details form with fields for Password (masked), Platform Name (Cisco router via SSH), Device Type (Network Device), Safe (Passwords), Name (Network Device-CiscoSSH), Last verified (N/A), Last modified (DanielleB (8/6/2013 10:28:07 AM)), Last used (DanielleB (8/6/2013 10:28:07 AM)), and Cisco entry Type (ciscoexec). The right section shows a table of activities with columns for Time, User, Action, Client ID, More info, and Reason. The table contains four rows of activity logs.

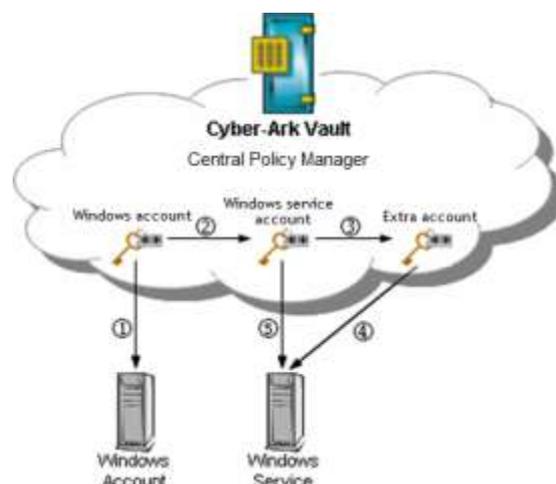
Time	User	Action	Client ID	More info	Reason
8/6/2013 10:28:07	DanielleB	Add File Category	PWA	DeviceType	Value=[Network D
8/6/2013 10:28:07	DanielleB	Add File Category	PWA	PolicyID	Value=[CiscoSSH]
8/6/2013 10:28:07	DanielleB	Add File Category	PWA	Type	Value=[ciscoexec]
8/6/2013 10:28:07	DanielleB	Store password	PWA		

12. Some platforms require additional information. You can specify this information in the tabs in the Account Details page.

Managing Service Accounts

The CPM can synchronize multiple copies of accounts that contain a password that has been changed and is used in different resources. These copies are also known as service accounts.

The following diagram shows the procedure that is carried out when the CPM changes and synchronizes passwords in accounts on Windows services. A full description of the procedure appears after the diagram.



After the CPM has replaced a Windows Account password successfully, it searches for Windows Services where the same password is used. If a logon password is required to log onto the Windows Service machine, the CPM searches for the associated account that is linked to the Windows service account and uses it to log on. Finally, the CPM replaces the password in the Windows Service account.

The following table lists the multiple service service accounts supported by the CPM, by default.

- Windows Services
- Windows Scheduled Tasks
- Windows Registry
- Windows IIS Application Pools
- Windows COM+ Applications
- Windows IIS Directory Security (Anonymous Access)
- Configuration Files
- Database String
- Web Application Accounts

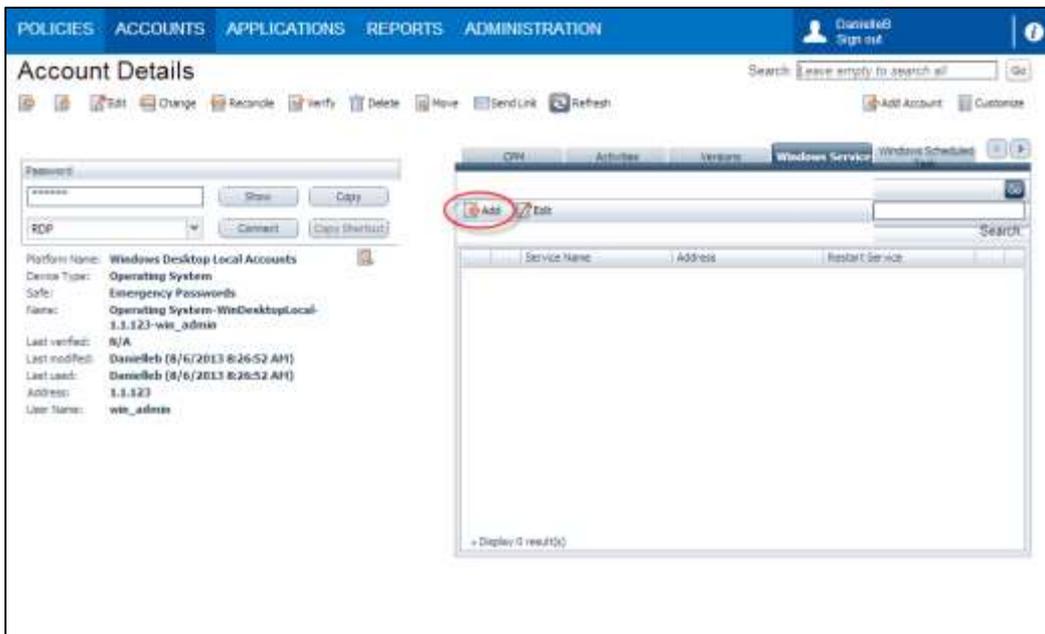
For more information about customized support for additional platforms, contact your CyberArk support representative.

To Create a Service Account

1. In the Accounts List, click an existing account; the Account Details window appears.

According to the PVWA environment, different Service Account tabs are displayed which enable you to work with accounts in the Password Vault in a variety of ways.

2. In the relevant service account pane (eg., Windows Services), click **Add**.

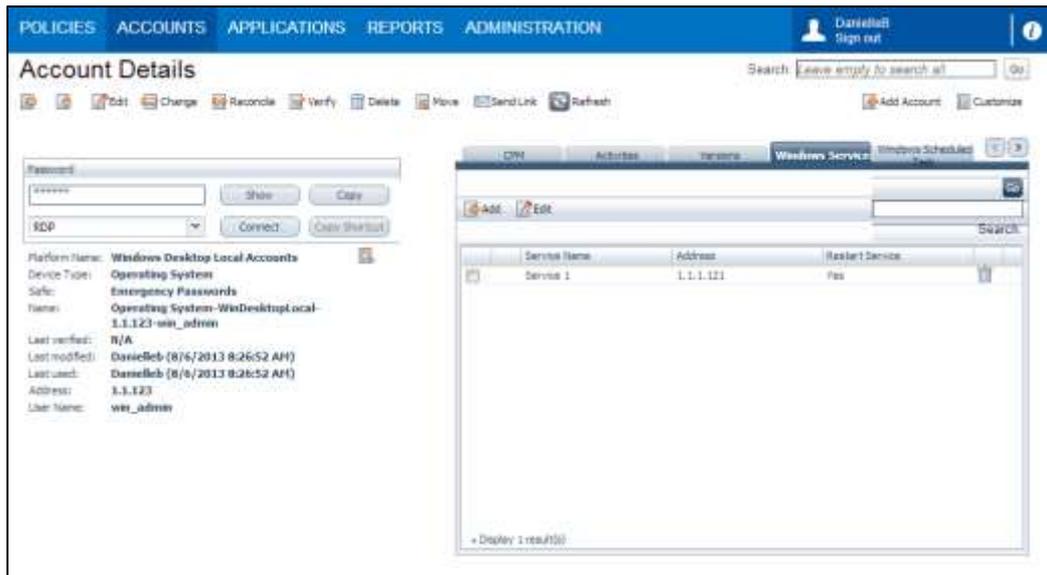


The Add Service Account page appears.



3. Specify the required information, then click **Save**; the service accounts that use the displayed account appear in the Service Accounts list.

The following example shows a list for service accounts of Windows Desktop Local accounts.



To Modify a Service Account

1. In the service account tab, select the service account to modify then click **Edit**; the Edit page appears.
2. Modify the account properties as necessary, then click **Save**; the Account Details page appears with the details of the modified service account.

To Delete a Service Account

1. In the service account tab, click the **Delete the Password** icon in the account row; a confirmation message appears.
2. Click **Yes** to delete the service account,
or,
Click **No** to leave the service account and return to the Account Details window.

Provisioning Accounts Automatically

The Accounts Feed, which is the next generation workflow for discovering and provisioning privileged accounts, replaces auto-detection functionality. For more information, refer to *Accounts Feed*, page 168.

The Privileged Account Security solution facilitates automatic full life-cycle management for Windows accounts and their service accounts in your enterprise, such as Windows Services, Scheduled Tasks, etc. This ranges from provisioning to removal or archiving, and includes all management tasks in between, ensuring complete control and secure management. This capability provides the following features:

- **Automatic provisioning for Windows local accounts** – Your enterprise's external directory can be integrated with the Password Vault to create, update, and remove privileged accounts automatically in the Vault for Windows machines in Windows domains.
- **Automatic provisioning for VMware Unix/Linux guest machines Root accounts** – Your enterprise's vCenter directory can be integrated with the Password Vault to create, update, and remove privileged accounts automatically in the Vault for Root or local accounts in VMware Unix/Linux guest machines.
- **Automatic provisioning for VMware ESX host Root accounts** – Your enterprise's vCenter directory can be integrated with the Password Vault to create, update, and remove privileged accounts automatically in the Vault for Root accounts in VMware ESX host machines. This enables you to maintain the organization password policy across your vCenter environment.
- **Automatic provisioning for usages** – All local and domain service accounts can be detected and provisioned automatically in the Password Vault, where they benefit from all of CyberArk's standard account life-cycle management features. This greatly reduces administration overhead required by the IT personnel when machines are added or updated, or when existing machines are removed from the external directory.
- **Flag domain or accounts used in Windows Services, Schedules tasks, etc, and are not currently managed by the Privileged Account Security solution** – Accounts that have not been used for a while and/or are not currently managed in the Password Vault can be automatically identified and flagged. This prompts the PIM to notify and/or automatically start managing any potential shared/privileged domain/local account that is used in a Windows Service or other Windows usage, and is not currently managed by the Privileged Account Security solution.
- **On demand automatic detection and reporting** – Users can initiate specific automatic detection processes for local and domain service accounts and generate a report of all the detected service accounts, with or without provisioning them in the Vault.
- **Auditing Automatic Detection Activities** – A record of all automatic detection activities is maintained in the Vault, and a report can be generated at any time with all these details, providing a full audit of every account and usage that is detected and/or provisioned in the Vault.

Provisioning Accounts and their Services

Due to the large number of Windows accounts that are used in enterprises today, it is a major challenge to keep track of each account, where it is used and when. As a result, many organizations avoid centrally managing and periodically changing the passwords of these accounts, as required by regulations and to maintain a higher level of security, in order to prevent failures of services, scheduled tasks, IIS settings, and other service accounts due to unsynchronized credentials. Maintaining accounts and their linked service accounts is tedious and time-consuming, and is almost impossible to manage manually in large enterprises, and is frequently neglected. In addition, accounts that are not used on a regular basis can be forgotten, and their services become unusable due to a lack of management and tracking.

The Privileged Account Security solution can detect accounts and their services in domain machines that are defined in the corporate AD and provision them automatically in the Vault, providing a full life-cycle automatic management system for Windows accounts and their services. This enables the Privileged Account Security solution to automatically manage Windows privileged accounts in environments that contain large numbers of Windows desktops and laptops, and transparently handle frequent machine moves, additions, and changes. This automatic detection and provisioning ensures that every usage in your enterprise for services, scheduled tasks, application pools, and every other usage supported in the Vault is identified and managed securely, according to enterprise policies.

According to predefined processes, the Central Policy Manager scans the enterprise Active Directory to detect all relevant machines, including new and removed ones, and can provision local accounts for each of those machines based on predefined templates. Default properties are assigned to these accounts so that the relevant platform can be applied to them. The CPM can also scan each of these machines for different types of service accounts for each account and provision them in the Vault

Once provisioned, depending on how the process is configured, the CPM can immediately initiate automatic management for accounts and service accounts, based on the associated platforms. After an account's password is changed, all service accounts associated with it are automatically synchronized. In addition, accounts and service accounts in the Vault are updated and removed to reflect their status in the domain, for example, when a machine is added or removed from the domain. Enterprises that do not wish the CPM to start managing accounts on remote devices as soon as the machine is detected can specify a delay between the time that the CPM detects a new machine in the domain and when the CPM will begin to manage accounts. This is particularly useful for enterprises that have a build process when new machines are added to the network.

While the CPM is scanning domain machines for service accounts, it can detect domain or local accounts that are not managed in the Privileged Account Security solution and either add them automatically or notify users of their existence, with a view to provisioning them automatically.

The flexibility of auto-detection processes provides multiple options for managing accounts and service accounts, as shown in the following examples:

- In one enterprise, a domain account that was created for a specific antivirus client is used to run the antivirus Windows service on multiple Windows machines, such as servers, desktops, and laptops. This enterprise has configured the Privileged Account Security solution to automatically detect all the service accounts for this domain account and, in addition, while the system is scanning domain machines for service accounts, it will also automatically detect domain accounts that are not currently managed in the Privileged Account Security solution and will immediately create them in the Vault without any human intervention.
- In another implementation, an enterprise wishes to centrally manage the local Windows administrators or all end users' workstations, desktops and laptops. The implementation can be configured to provision these local accounts automatically according to a preconfigured template, and to automatically detect any additional service accounts such as Windows Services on these machine, where the local administrator is used to run the service. Each time a usage is detected, if it is associated with the relevant provisioned local account, the Central Policy Manager automatically manages these accounts and their service accounts based on the associated predefined platform, including automatically changing and verifying their passwords, as well as keeping the service accounts synchronized accordingly.

The CPM process that defines auto-detection specifies the machines to detect and scan, as well as the types of accounts and service accounts to provision in the Vault. Users determine how frequently machines are detected or scanned according to intervals and time frames. For more information about configuring the CPM to detect Windows accounts and service accounts automatically, refer to *Configuring Automatic Provisioning*, page 390.

The CPM can be configured to manage accounts that are auto-detected as members of a single group. For more information, refer to *Managing Auto-Detected Accounts with Groups*, page 417.

Invoking Automatic Provisioning Manually in the PVWA

Users can initiate detection processes manually in the PVWA, using the **Detect Now** option in the Accounts List, according to preconfigured auto-detection processes. When selecting an auto-detection process, users can specify any combination of the following three processes:

- **Detect machines** – The process will detect any new or removed machines in preconfigured domains.
- **Scan newly detected machines** – The process will scan new machines that have just been detected for accounts and service accounts. This option is selected together with 'Detect machines'.
- **Scan Machines** – The process will scan previously detected machines for accounts and service accounts.
- All other configurations will be taken from the selected process configuration.

In order to initiate an auto-detection process manually, users must be members of the group that is authorized to configure the PVWA. By default, this is the Vault Admins group.

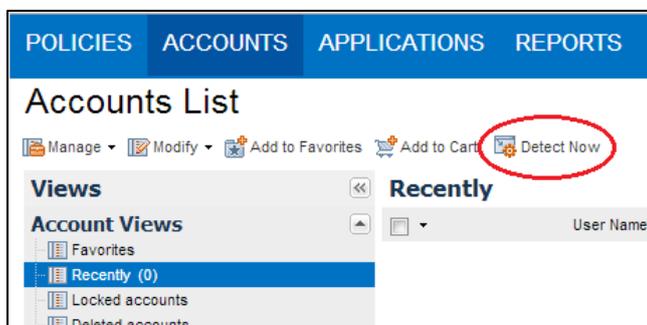
This feature **must be enabled** manually. For more information about enabling this feature, refer to *Configuring Automatic Provisioning*, page 390.

Running Auto-Detection in Simulation Mode

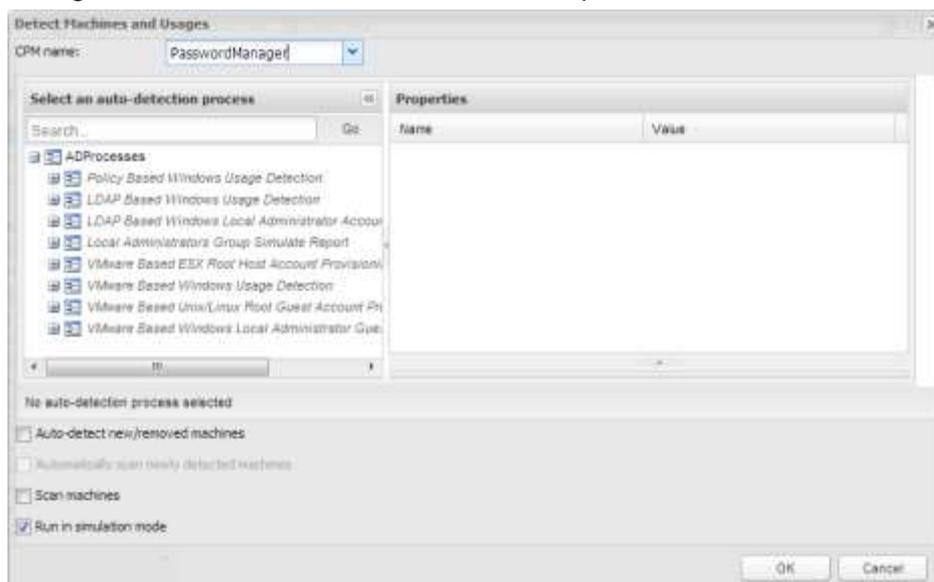
Auto-detection processes can be initiated in the PVWA in 'simulation mode' without provisioning new accounts or service accounts in the Vault or creating notifications. All the machines, accounts, and service accounts that are detected are listed in a report that is generated and saved in the Reports Safe. This mode enables users to make sure that all the auto-detection processes are defined according to their needs.

To Invoke Automatic Provisioning Manually in the PVWA

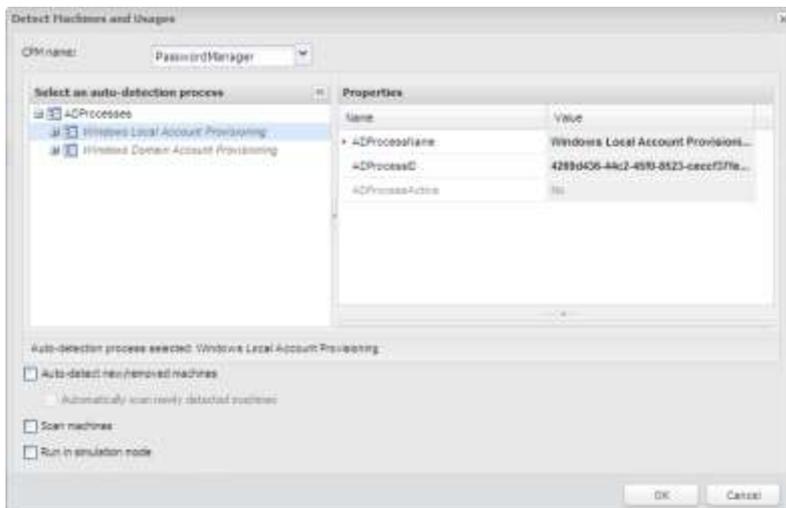
1. Display the Accounts List; members of the Vault Admins group can see the **Detect Now** button. This button enables authorized users to run auto-detection processes manually.



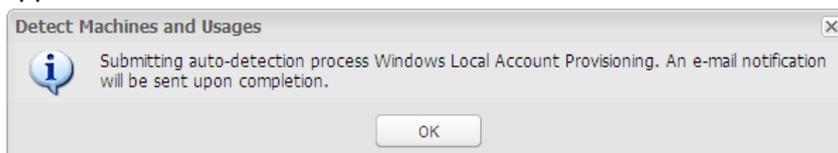
2. Click **Detect Now**; the Detect Machines and Usages window appears. This window enables you to select a process and customize where and how it will run in the manually activated process. These customizations override the configurations defined in the auto-detection process.



3. In the CPM drop-down list, select **PasswordManager**, which is the name of the CPM that manages the auto-detection process. This was defined when this process was configured. If you are authorized to run auto-detection processes for the selected CPM, the list of processes that can be run by this CPM are displayed. If not, a message is displayed indicating that you are not authorized to run auto-detection processes for this CPM.
4. Select the process to run; the name of the selected process appears under the list of processes and the details of the process appear in the Properties list.



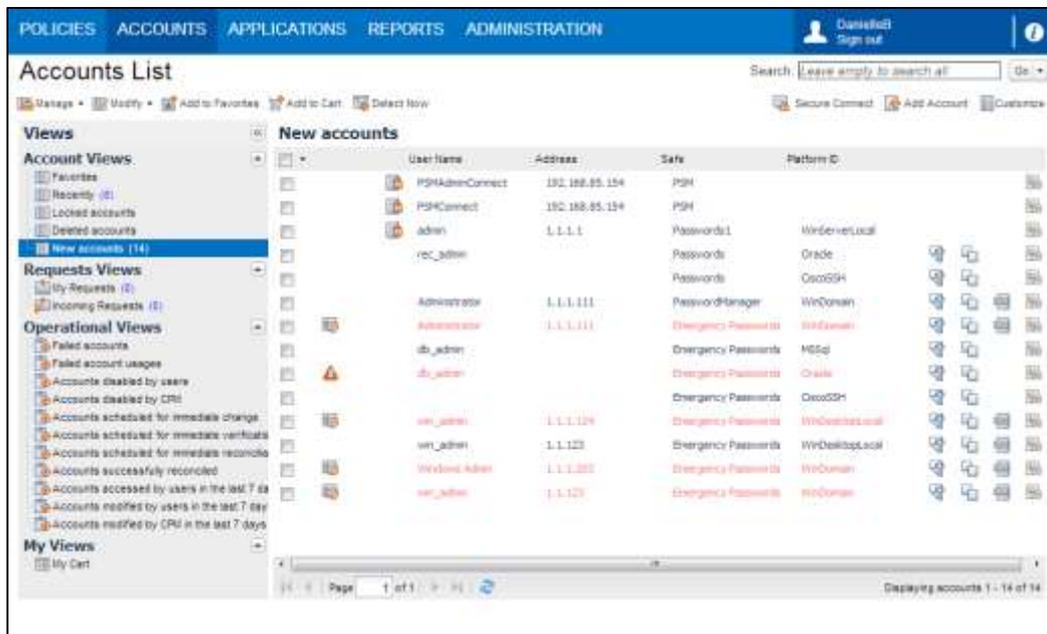
5. Select the following options to indicate which machines the auto-detection process will run on.
 - **Auto-detect new/removed machines** – The process will detect machines in the external directory defined in the process. If the process is not configured to auto-detect machines, this option will be disabled and you will not be able to select it.
 - **Automatically scan newly detected machines** – The process will scan new machines that have just been detected by the CPM. This option is dependent on the previous option, and can only be selected if the previous option is enabled.
 - **Scan machines** – The process will scan machines that have been detected previously by the CPM.
6. To run the auto-detection process in simulation mode, select **Run in simulation mode**. This option is enabled for users who are members of the group that is authorized to access the System Configuration page (by default, Vault Admins) and the group that is authorized to monitor the PVWA (by default, PVWAMonitor).
7. Click **OK** to start running the auto-detection process; the following message appears.



Checking for New Accounts

You can check for new accounts that have been added to any of the Safes where you are an owner.

- In the Accounts page, click **New Accounts**.



The Accounts List displays all the accounts that were added to your Safes during a predefined number of days. This number of days appears in the tooltip that appears when you point to **New Accounts**.

If the PWA is configured to check for new and locked passwords automatically, the New Accounts counter that displays the number of new accounts in the Vault will be updated each time the Accounts page is displayed or refreshed. For more information, refer to *Displaying New and Locked Accounts*, page 497.

Defining Custom Account Properties

During installation, a number of password properties are created automatically in the Vault. These properties enable the passwords to be managed automatically by the Privileged Account Security solution. Users who have the Manage Vault File Categories authorization can specify additional password properties manually that can be applied to all passwords that are stored in the Vault.

Creating an Account Property

1. Log onto the PrivateArk Client with an administrative user.
2. From the **File** menu, select **Server File Categories**; the File Categories window appears.

The File Categories window displays the following information:

Column	Displays
Name	The name of the category
Type	The type of the category. This might be text, numeric, or list.
Valid Values	The available options from which you can choose if the category is a list.
Default	The default setting for the category.
Required	Indicates whether or not the user is required to supply the category before the file/object can be stored.

3. Click **New**; the Add File Category dialog box appears.
4. In the Name edit box, type the new category. For example, if you want to be able to find files according to the IP address where the password will be used, you would type 'IP address'.
5. From the Type drop-down list, specify the type of category. Choose from text, numeric or list.

If the category is a List type, the Valid values section of the dialog box becomes active.

- To add a value, in the Value edit box, type the first value for the list, then click **Add**; the value is added to the list. Add as many values as necessary.
- To set a value as the default value which will appear automatically as the property, select a value, then click **Set as Default**.
- To remove a value from the Values list, select the value, then click **Delete**.

- To make the password properties a requirement for every password stored in the Vault, select **Required Category**.



- Click **OK**; the new File Category appears in the File Categories window.
- To ensure that this password property is used in all Safes that will be created from now on, select **Use File Categories in new Safes**, then click **OK** to set the new password property in the Vault.

Updating Account Property Definitions

After an account property has been defined, it can be changed and, if it is a list of values, more values can be added to it.

To Update Password Property Definitions

- In the PrivateArk Client, display the File Categories on Vault window.
The list of categories displays all the password properties that are currently available.
 - To create a new password property, click **New**; the New File Category window appears. Specify the password property as explained in *Creating an Account Property*, page 161.
 - To edit an account property, click **Edit**; the Edit File Category window appears. In the Name edit box, select the property to edit.
 - If the property is text or numeric, you can change it now.
 - If the property is a value, select the property, then in the Type edit box, specify a value and click **Add**; the new value is added to the list of values for that property.
 - To delete a password property, select the property to delete, then click **Delete**.
- In the File Categories window, click **OK**; the changes to the password property are applied and can be used when specifying properties on new or existing passwords.

Moving Accounts between Safes

Users can move accounts between Safes and reorganize accounts. This procedure deletes the original account from the current Safe and moves the entire account to a target Safe, including all the account properties. This feature moves the following information:

- **Accounts** – Accounts can be moved between Safes, including the password and the account properties.
- **Account groups and group members** – Members of account groups can be moved between Safes.
- **Accounts and their usages** – Accounts can be moved between Safes, together with all associated service accounts.

Before Moving Accounts

Before you move the selected account(s), check the following:

In the source Safe:

1. Check that the user has the following authorizations:
 - List passwords
 - Retrieve passwords
 - Update password properties
 - Delete passwords
 - Access Safe without confirmation – If the user doesn't have this authorization, they must have a confirmed request so that they can access the password.
2. Check the current CPM status for the account:
 - Is the CPM currently performing an account management task? Passwords that are in the process of being managed cannot be moved. Wait until the CPM has completed its tasks, then move the password.
 - Has the account been marked for a CPM management task? As the account is disabled before it is moved, the CPM will not be able to complete its management tasks on this account. You can either wait until the CPM has finished its management tasks on this account and then move it, or move it and it will be managed in the target Safe.
3. If passwords are in exclusive mode, make sure that the password is not locked, as locked passwords cannot be moved. Wait until the password has been released then move the password.
4. Make sure that you are aware of the following:

When an account is moved to a different Safe, the following information is not moved with it:

 - **Audit records** – All records of activity on the accounts that are moved are left in the source Safes.
 - **Versions of the password** – All previous versions of accounts that are saved in the Safe according to the Safe history configurations are left in the source Safe.

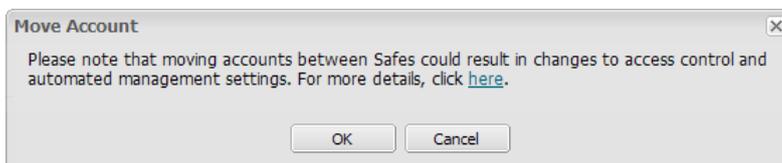
- **Requests** – All requests for access and confirmations that have been received from authorized users.
- **Object Level Access Control configurations** – Specific access control at account level.
- **Links to the account** – Accounts that are associated with the accounts that are moved are not updated with the new Safe details. In addition, accounts that are referenced in platforms (e.g. reconcile accounts) should be updated with the new Safe name.
- **One-time passwords** – Accounts that are configured for one-time use and have been used but not changed before they are moved to a different Safe will not be changed before they are used again.
- **PSM Recordings** – When accounts that are connected to PSM recordings are moved to a different Safe, the connection between the account and the recording is lost. Relevant recordings can be found by performing a search through the MONITORING page.
- **Moving accounts to subfolders** – When an account is moved to the target Safe, folders one level below the 'Root' level will be created automatically. If an account is in a lower level in the source Safe, the user must create the folder tree in the PrivateArk Administrative Client.
- **Moving exclusive account groups** – When members of exclusive account groups are moved to a different Safe, it is recommended to move all the members in the group and not just some of them. If only some of the members of the group are moved, the rest of the members in the group are locked and they must be released manually before they become available to other users. For more information about exclusive accounts, refer to *Accounts Check-out and Check-in*, page 246.

In the target Safe:

1. Check that the user has the following authorizations:
 - List passwords
 - Create passwords
 - Update passwords
 - Update password properties
2. Check that the name of the password being moved is unique. If there is an account with the same name in the target Safe, the account will not be moved. However, if there is a deleted account with the same name, the account will be moved successfully.
3. In the platform that the password is linked to, check that the target Safe is listed in the **AllowedSafes** parameter. This ensures that the same platform can be applied to passwords after they have been moved.

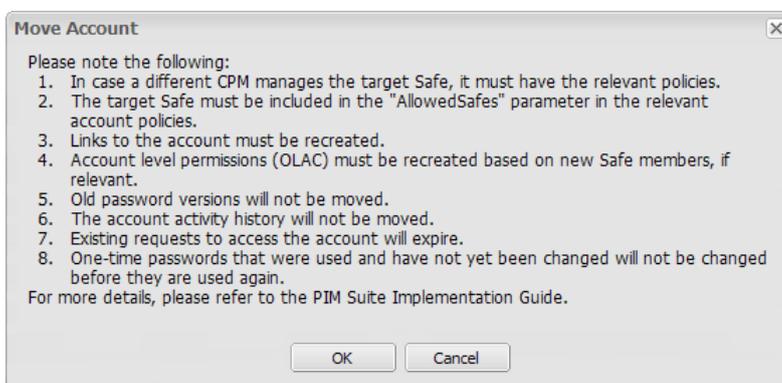
Moving Accounts

- Select the account(s) to move:
 - In the **Accounts List**, select one or more accounts, or,
 - Display the **Account Details** page for the account to move.
- Click **Move**; the following message appears:



- Click **here** to display more information about which information is moved with the account and which information can be lost when accounts are moved.

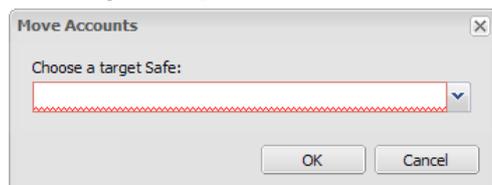
Note: It is essential that you are aware of the implications of moving accounts between Safes **before** you move them.



For more information about points 2, 5, 6, 7, and 8, refer to *Before Moving Accounts* above. Make sure that you have checked all these points before starting to move the account(s).

For more information about points 1, 3 and 4, refer to *After Moving Accounts* below.

- After you have read the information in the above message, click **OK**; the Move Accounts window appears.
- Specify the Safe where the accounts will be moved to, then click **OK**; the accounts are moved to the target Safe.
 - If you specify the name of a Safe that does not exist or which you are not authorized to access, the target Safe edit box will be cleared as shown in the following example:



Select a Safe name from the drop-down Safe list, then click **OK**.

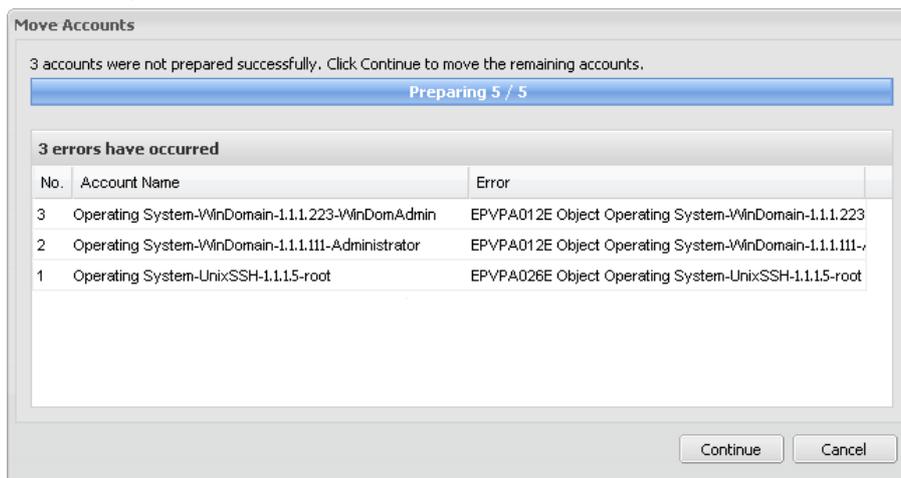
6. A progress bar indicates the status of the process.

- If the accounts are moved successfully, the following message is displayed:



Click **Close**; the Accounts List or Account Details page is displayed, depending on the page in which you selected accounts to move.

- If one or more of the accounts could not be moved successfully, the errors are displayed:



Click **Continue** to continue moving the account(s) that can be moved successfully to the target Safe,

or,

Click **Cancel** to cancel the entire Accounts transfer process and close the Move Accounts window.

After Moving Accounts

1. Accounts that were not moved successfully are disabled for automatic CPM management in the source Safe. Enable them manually so that they will be managed automatically according to your enterprise policies.
2. Check that all the accounts that were moved to a target Safe are associated with the correct platforms.
3. For accounts that were linked to the moved accounts in the previous Safe, recreate all the relevant links.
 - i. In the Accounts page, search for the name of the password that was moved.
 - ii. In the list of search results, display the Account Details page for each password, and in the CPM tab, check if the original password is used as a logon or reconcile account.
 - iii. If so, in the additional passwords section, click **Associate** to re-link either the logon account or the reconcile account; a list of accounts appears.
 - iv. Select the account that was moved, then click **Associate**; the selected account is associated with the original account and is listed in the CPM pane of the Account Details page.

For more information about linked accounts, refer to *Linked Accounts*, page 220.

- In addition, check in the UI & Workflows parameters of the platform settings if any of the moved accounts are used in platforms as a reconcile account, and update the Safe name to indicate its new location.
4. For accounts that were configured for account level permissions, recreate these access permissions, based on the Safe members in the new Safe.
 5. If all the members of a group have been moved to another Safe, the group manager object in the source Safe will be deleted and cannot be undeleted. If necessary, the group in the source Safe must be recreated manually.

If you moved only some of the members of a group that is configured for exclusive mode, the rest of the members in the group in the original Safe are locked and they must be released manually before they become available to other users. This can be done by releasing any of the group members in the source Safe. For more information about exclusive accounts, refer to *Accounts Check-out and Check-in*, page 246.

Accounts Feed

CyberArk's new Account Feed introduces the next generation workflow of discovering and provisioning of privileged accounts. This workflow is aligned with business processes in order to simplify and accelerate the deployment and management of privileged account security. The Accounts Feed process is divided into three main steps:

- **Discover** – Automate privileged account discovery, which is designed to quickly locate critical accounts and credentials.
- **Analyze** – Provide an easy view of all discovered accounts that enable you to analyze, determine which account is no longer needed and can be deleted, and assess the risk of each account.
- **Provision** – The scope of the accounts to manage can be provisioned in the Vault in a simple and intuitive way.

The Privileged Account Security solution scans your machines according to the source that was defined, such as Active Directory or a CSV file, to discover all the privileged accounts in your organization (such as Windows and Unix accounts) and their dependencies (such as Windows Services), giving you a clear and comprehensive picture of existing accounts in your organization. All the detected accounts are displayed in the Pending Accounts page in the PVWA, where you can view them and onboard them, based on various criteria that you can define. The Privileged Account Security solution uses the CyberArk Central Policy Manager Scanner to run account discoveries. A scanner is installed with each CPM in your environment, enabling you to scan all distributed networks in your organization.

You can schedule account discoveries to run automatically, once or at regular intervals, streamlining account management and ensuring that the pending account list contains the most up-to-date details about the privileged accounts in your environment. Then you can view recurring discoveries and see when they last ran, when they'll next run, and how long each one took.

Supported Target Machines

The Accounts Feed can scan the following target machines:

▪ **Windows Accounts**

- Discovery processes detect the following Windows accounts:

- Local and domain accounts
- Windows Services and Scheduled Tasks accounts

Note: During a discovery on a specified source machine, if an account from another domain is detected, and there are trusts between the domains, the scanner will not retrieve all the information for that account as the other domain was not configured as a source machine.

- Servers:

- Windows 2000
- Windows 2003
- Windows 2008
- Windows 2012

Note: To discover Scheduled tasks on Windows 2012, the CyberArk Scanner (CPM) must be installed on Windows 2012.

- Workstations:

- Windows 2000
- Windows XP
- Windows Vista
- Windows 7
- Windows 8

Note: In order to discover Scheduled Tasks on Windows 8, the CyberArk Scanner (CPM) must be installed on Windows 2012.

- Supported protocols:

The following protocols are supported when accessing the Active Directory:

- LDAP
- LDAPs

▪ **Unix Accounts**

- Discovery processes detect the following Unix accounts:

- Local accounts

Note: Domain users that are used to authenticate to Unix machines (using AD-Bridge integration) are currently not discovered.

- Unix platforms:

- RHEL 4-6
- Solaris Intel and Solaris SPARC 9, 10, 11
- AIX 5.3, 6.1, 7.1
- ESXi 5.0, 5.1
- SUSE 10
- Fedora 18,19, 20
- CentOS 6
- Oracle Linux 5

Managing Discovery Processes

The Discovery Management page allows you to create new discovery processes and view their statuses. In addition, you can select an existing discovery process and view its details in the Preview pane. The grid can be refreshed to update discovery processes that are currently running in the system, providing you with a clear and concise view of discovery processes in real time.

The Discovery Setup grid gives you an at-a glance view of configured discoveries. Select a discovery to view more details.

The following table describes the information displayed in the Discovery Setup grid.

Column	Displays
Discovery name	The name of the defined discovery.
Type	The type of discovery process. Possible values are: <ul style="list-style-type: none"> Onetime Recurring
State	The current state of the discovery process. Possible values are: <ul style="list-style-type: none"> Running Pending The next run date and time (for recurring discoveries) For more information, refer to <i>Viewing the Status of the Discovery Process</i> , page 183.
Last run time	The time on the scanner machine when this discovery starts. If this is a recurring discovery, this time indicates the last time this discovery was started.
Last run status	The status of the discovery the last time it was run.

Viewing Discovery Details

You can view the details of each discovery in the Preview pane in the Discovery Management page. These details give you an overall view of the discovery and its status.

The details in the Preview Pane change, depending on whether the selected discovery is Windows or Unix.

To View Discovery Details

- Select or click a discovery row; its details are displayed in the Discovery Preview pane.

The screenshot shows the Discovery Management page with a table of 14 discovery setups. The selected row is 'Windows discovery from Comp...' with a state of 'Recurring' and a last run time of '2/15/2016'. The Discovery Preview pane on the right shows details for this discovery, including the discovery name, CPM scanner, state, type, recurrence, and creation information.

Discovery name	Type	State	Last run time	Last run status
Windows discovery from Comp...	Recurring	2/15/2016	-	-
Windows discovery from GUI...	Onetime	-	2/18/2016	Comp
Windows discovery from GUI...	Onetime	-	2/18/2016	Comp
Windows discovery from GUI...	Onetime	-	2/18/2016	Comp
Windows discovery from SubO...	Onetime	-	2/18/2016	Comp
Unix discovery from file (DreM...	Onetime	-	2/18/2016	Comp
Windows discovery from QA te...	Onetime	-	2/18/2016	Comp
Unix discovery from file (DreM...	Onetime	-	2/18/2016	Comp
Windows discovery from Accou...	Onetime	-	2/18/2016	Comp
Windows discovery from GUI Ty...	Onetime	-	2/18/2016	Comp
Windows discovery from Accou...	Onetime	-	2/18/2016	Comp
Unix discovery from file (DreM...	Onetime	-	2/18/2016	Comp
Windows discovery from Accou...	Onetime	-	2/18/2016	Comp
Windows discovery from GUI Ty...	Onetime	-	2/18/2016	Comp

Discovery Preview

- Discovery name: Windows discovery from Computers@company.com
- Domain: company.com
- Organizational unit: CN=Computers,DC=company.com,DC=com
- CPM scanner: PasswordManager
- State: 2/15/2016 12:00:00 AM
- Type: Recurring
- Recurses on: Monday
- Starting from: 00:00 AM
- Created on: 2/11/2016 2:11:40 PM
- Created by: administrator

The following table describes the information displayed in the Discovery Preview pane.

Column	Displays
Discovery name	The name of the defined discovery process, which indicates the following: <ul style="list-style-type: none"> ▪ Type of discovery – Windows or Unix ▪ Discovery source – Domain, OU, or file Note: These details differ according to the type of discovery.
CPM Scanner	The name of the CPM scanner that will run this discovery.
State	The current state of the discovery process. Possible values are: <ul style="list-style-type: none"> ▪ Running ▪ Pending ▪ The next run date and time (for recurring discoveries) For more information, refer to <i>Viewing the Status of the Discovery Process</i> , page 183.
Type	The type of discovery process. Possible values are: <ul style="list-style-type: none"> ▪ Onetime ▪ Recurring

Column	Displays
Rekurs on	The days on which the discovery process runs.
Starting from	The time when the discovery process runs.
Created on	The date and time when the discovery process was created.
Created by	The name of the user who created this discovery.
Start running time	The time when the discovery started.
Last run time	The last time when this discovery was started, according to the time on the scanner machine.
Last run end time	The last time that this discovery ended.
Last run status	The final status of the discovery the last time it was run.

Creating Discovery Processes

Discovery processes scan predefined machines for new and modified accounts and their dependencies, and then display the discovered accounts so that you can see which accounts should be onboarded into the Vault where they can be managed automatically and securely, according to your enterprise compliancy policies.

Before creating discovery processes, make sure that the user who will perform the discovery has the following permissions:

- For Windows discoveries:
 - In the Active Directory:
 - Read permissions in the OU to scan and all sub-OUs
 - On target machines:
 - Domain Administrator
- or,
- Equivalent Domain User:
 - User with read permissions on the Active Directory
 - User with local administrative rights for Windows

Note:

- In Windows Vista or newer, a user who is not a domain Administrator must be a domain user and must belong to the Administrators group or to a group nested within the Administrators group.
- In older versions of Windows, the domain user can be a member of any privileged group.

For more information, refer to *To Create a Discovery Process for Windows Accounts*, below.

- For Unix discoveries, the user must have one of the following privileges:

Privilege	Enables user to retrieve ...
root or user with uid=0	All account details
sudoers for the "cat /etc/passwd" command	The minimum details required to create a pending account (user name and address)
sudoers for the following commands: <ul style="list-style-type: none"> ▪ cat "/etc/shadow" ▪ cat "/etc/passwd" ▪ cat "/etc/security/passwd" (AIX) ▪ cat "/etc/security/lastlog" (AIX) ▪ cat /etc/group ▪ cat "/etc/sudoers" ▪ lastlog grep -v '*' ▪ hostname -s ▪ ls -d /etc/[A-Za-z]*[_-][r]e[lr]* grep -v 'lsb os\system' ▪ test -f "{0}"; echo \$? 	All account details

For more information, refer to *To Create a Discovery Process for Unix Accounts*, page 177.

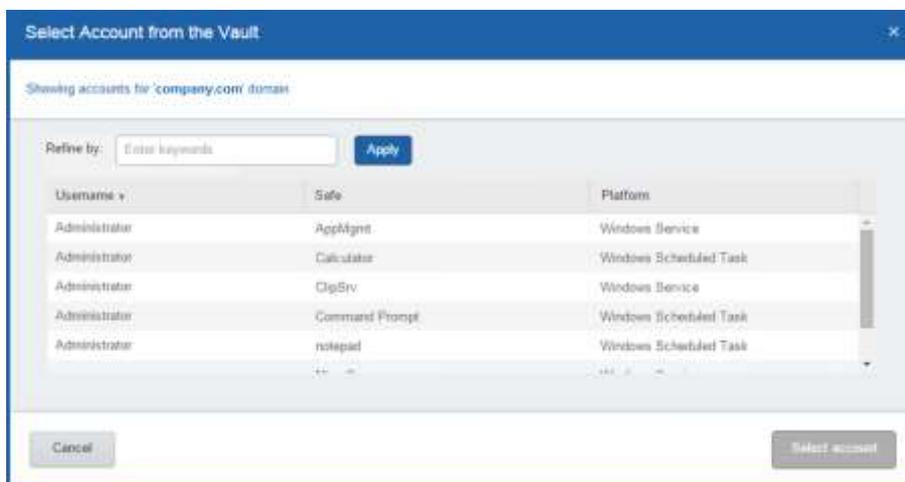
To Create a Discovery Process for Windows Accounts

- Log onto the PWWA with a user who is a member of the Vault Admin group.
- In the **ACCOUNTS** page, under **Accounts Management**, click **Accounts Discovery**; the Pending Accounts page appears.
- In the Pending Accounts page, click **Discovery Management**; the Discovery Management page appears.
- In the Discovery Management page, click **New Windows Discovery**; the New Windows Accounts Discovery window appears.

5. Under **Which account to use for scanning?**, do the following:
- i. Specify the Active Directory domain from which the list of machines to scan will be retrieved.

Notes:

- Make sure that the PVWA machine has access to the Active Directory in order to show the tree-view of the OUs.
 - Make sure that the CPM scanner that will perform the scan has access to the Active Directory.
 - Specify the domain name in fully-qualified domain name (FQDN) format and with up to 170 characters.
- ii. Select the user who will perform the scan. Either select an account from the Vault or manually specify a user and password:
 - **Select from Vault** – Select a Vault account to run the scan. This is recommended for recurrent scans.
 - i. Click **Click to select an account from the Vault**; a list of Vault accounts appears. These are all domain accounts in the specified domain.



- ii. Select the account to use for the discovery process, then click **Select Account**; details of the selected account are displayed.

Note: Make sure the user in this account has the relevant permissions. For more information, refer to *Creating Discovery Processes*, page 172.

- **Specify Account** – Specify a domain account that will run the scan.
 - i. Click **Specify Account**; prompts for the required details appear.

- ii. Specify the following information:
 - **User** – The Active Directory user that will access the domain to retrieve the list of machines and will access each machine to perform the scan.

Notes:

 - Specify the user name in username or domain\username format.
 - This user requires read permissions in the OU and all sub-OUs to scan.
 - **Password** – The domain user's password.
6. Under **What to scan?**, specify the OU that will be scanned for accounts and their dependencies. You can either select it from the tree view or type its distinguished name.
 - **To select the OU from a tree** – Click **Browse**; the PVWA connects to the Active directory using the user credentials specified in the 'Which user to use for scanning?' section and displays the accessible Active Directory tree. Select the OU to scan, then click **OK**; the selected OU appears in the Accounts Discovery window. This OU will be scanned recursively.

Note: You can only select one OU. If no OUs are selected, by default, the selected domain will be scanned.
 - **To type the OU's distinguished name** – In **OU to scan**, type the distinguished name of the OU.

Note: For LDAPs, type the OUs distinguished name manually.

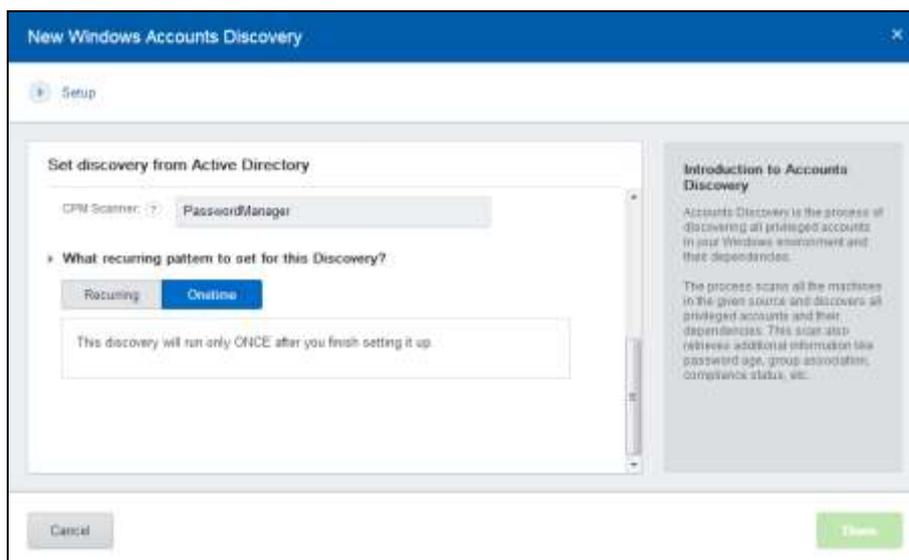
7. Under **Which CPM scanner to use?** select the CPM scanner that will scan for accounts and their dependencies. The CPM scanner will scan only machines that it can physically access.

The screenshot shows the 'New Windows Accounts Discovery' window in a 'Setup' state. It is titled 'Set discovery from Active Directory'. Under 'What to scan?', the 'OU to scan' field contains 'company.com' and a 'Browse' button is visible. Under 'Which CPM Scanner to use?', a dropdown menu shows 'PasswordManager' selected. Under 'What recurring pattern to set for this Discovery?', the 'Recurring' button is selected, and the 'Recur on' section has 'Monday' checked. At the bottom, there are 'Cancel' and 'Done' buttons.

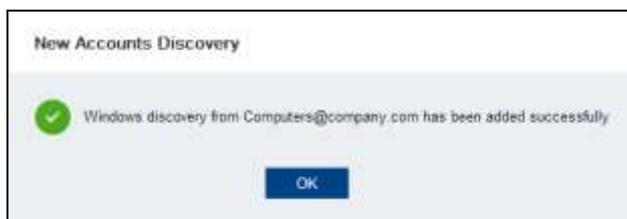
- If multiple CPM scanners are installed, select the relevant scanner from the drop-down list.
 - If only one CPM scanner is installed, only the name of that scanner will be displayed and it will be automatically selected.
8. Under **Should this discovery be one time or recurring?** select one of the following:
- **Recurring** – Enables you to define an automatic recurring discoveries. Specify the following details:
 - **Recur On** – Select the day(s) when the discovery will run.
 - **Starting** – Set the time after which the discovery will run.

This screenshot shows the same 'New Windows Accounts Discovery' window, but with more details in the 'What recurring pattern to set for this Discovery?' section. The 'Recurring' button is selected. The 'Recur on' section has 'Monday' checked. The 'Starting' section has '12:00 PM' selected. The 'Done' button is highlighted in green.

- **One time** – Defines a one time discovery process that will run after you finish setting it up.



9. Click **Done**; the following message appears:



10. Click **OK**; the discovery is added to the list of discoveries in the Discovery Management page.
 - One time discoveries are performed as soon as the scanner finishes current discoveries.
 - Recurrent discoveries are added to the list of pending discoveries and will be performed on the defined day at the specified time.

To Create a Discovery Process for Unix Accounts

1. Log onto the PVWA with a user who is a member of the Vault Admin group.
2. In the **ACCOUNTS** page, under **Accounts Management**, click **Accounts Discovery**; the Pending Accounts page appears.
3. In the Pending Accounts page, click **Discovery Management**; the Discovery Management page appears.

- In the Discovery Management page, click **New Unix Discovery**; the New Unix Accounts Discovery window appears.

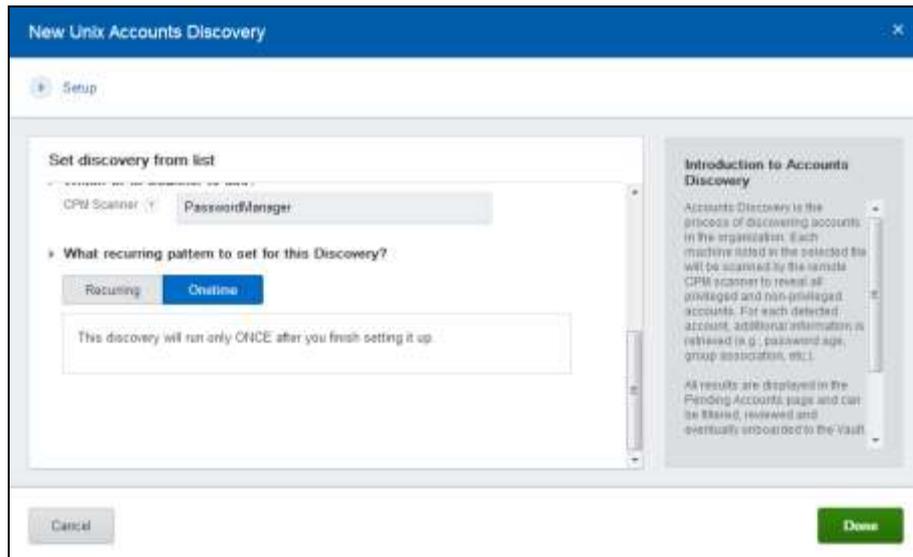
- Under **Which file contains the list of machines?**, click **Browse**, then select the CSV source file that the discovery will use.
- Under **Which user will scan the machines?**, specify the name of a user who will connect to the Unix machines to scan them. The system will search the Vault for this user's password credentials for each machine that is listed in the source list. If the account contains only an SSH key, the default password will be used.
Note: The Master Policy rules that are relevant to this account are not applied when it is used in discoveries.
- Under **What is the user's default password?**, specify the password to use when the system cannot find the specified user in the Vault.

8. Under **Which CPM scanner to use?** select the CPM scanner that will scan for Unix accounts. The CPM scanner will scan only machines that it can physically access.

- If multiple CPM scanners are installed, select the relevant scanner from the drop-down list.
 - If only one CPM scanner is installed, only the name of that scanner will be displayed and it will be automatically selected.
9. Under **What recurring pattern to set for this Discovery?** select one of the following:

- **Recurring** – Enables you to define an automatic recurring discoveries. Specify the following details:
 - **Recur On** – Select the day(s) when the discovery will run.
 - **Starting** – Set the time after which the discovery will run.

- **One time** – Defines a one time discovery process that will run after you finish setting it up.



10. Click **Done**; the following message appears:



11. Click **OK**; the discovery is added to the list of discoveries in the Discovery Management page.

- One time discoveries are performed as soon as the scanner finishes current discoveries.
- Recurring discoveries are added to the list of pending discoveries and will be performed on the defined day at the specified time.

Deleting Discovery Processes

When discovery processes are no longer required and they become redundant, they can be deleted, so that only the necessary scheduled processes are listed.

Note: A discovery process cannot be deleted while it is running.

To Delete Discovery Processes

1. In the Discovery Management Page, select the discovery process to delete.

The screenshot shows the 'Discovery Management' page in the CyberArk console. The page has a navigation bar with 'POLICIES', 'ACCOUNTS', 'APPLICATIONS', 'REPORTS', and 'ADMINISTRATION'. The user is logged in as 'administrator' with a last sign-in of '2/10/2016'. The page title is 'Accounts > Discovery Management' and 'Discovery Management'. There are buttons for 'New Linux Discovery' and 'New Windows Discovery'. A sidebar on the left has 'Discovery Management' selected. The main content area shows a table of 14 discovery setups. The table has columns for 'Discovery name', 'Type', 'State', 'Last run time', and 'Last run status'. One row, 'Windows discovery from OUI', is highlighted in blue. A 'Delete' button is located at the bottom right of the page.

Discovery name	Type	State	Last run time	Last run status
Windows discovery from Camp...	Recurring	On	2/11/2016	-
Windows discovery from OUI...	Onetime	On	2/10/2016	Comp
Windows discovery from OUI...	Onetime	On	2/10/2016	Comp
Windows discovery from OUI...	Onetime	On	2/10/2016	Comp
Windows discovery from SubO...	Onetime	On	2/10/2016	Comp
Unix discovery from file (Defi...	Onetime	On	2/10/2016	Comp
Windows discovery from QA te...	Onetime	On	2/10/2016	Comp
Unix discovery from file (Defi...	Onetime	On	2/10/2016	Comp
Windows discovery from Acces...	Onetime	On	2/10/2016	Comp
Windows discovery from OUI Ty...	Onetime	On	2/10/2016	Comp
Windows discovery from Acces...	Onetime	On	2/10/2016	Comp
Unix discovery from file (Defi...	Onetime	On	2/10/2016	Comp
Windows discovery from Acces...	Onetime	On	2/10/2016	Comp
Windows discovery from OUI Ty...	Onetime	On	2/10/2016	Comp

2. Click **Delete**; the following message appears, prompting you for confirmation:

The screenshot shows a 'Delete Discovery' dialog box. It contains a warning icon and the text: 'Are you sure you want to delete Discovery 'Windows discovery from OUI@company.com'?'. Below this, it says: 'Deleting a discovery means it will no longer be available in the Discovery Management page and its recurrent discoveries will no longer occur.' At the bottom, there are two buttons: 'Cancel' and 'Delete discovery'.

3. Click **Delete discovery**; the system deletes the discovery process in one of the following ways:

- **Recurrent discoveries** – The selected discovery is deleted together with all its details and details of the previous times it ran.
- **One time discoveries** – The selected discovery is deleted.

4. After the discovery has been deleted successfully, the following message appears:

The screenshot shows a 'Delete Discovery' dialog box with a success message. It contains a green checkmark icon and the text: 'Discovery 'Windows discovery from OUI@company.com' has been deleted'. At the bottom, there is an 'OK' button.

Discovering Accounts

The Accounts Feed discovers both local and domain accounts, as well as both privileged and non-privileged. The discovery process classifies each account, so that you know whether it is a local or domain account (in Windows discoveries), and privileged or not. Additional information also helps you understand the type of accounts that have been discovered and helps you to assess the risks associated with each account.

In addition, in Windows discoveries, the discovery finds Windows Services and Windows Scheduled Tasks that use the detected privileged accounts. In Unix discoveries, the discovery does not find dependencies on Unix machines.

Accounts that already exist in the Vault will not be rediscovered. This refers to accounts that were added in the PWWA, onboarded using the Accounts Feed, or provisioned using the AddAccount web service.

Accounts that are displayed in the Pending Accounts list may have changed since they were initially discovered. In order to make sure that the Pending Accounts list reflects the current status, you can perform a new discovery process in which the same accounts are rediscovered and their details are updated.

Notes:

- When you configure the discovery to scan a company domain, sub-domains will not be scanned automatically. For example, when you scan the **mycompany.com** domain, the **sub.mycompany.com** domain will not be scanned automatically.
- When scanning accounts and groups in a trusted domain, it is recommended to perform a separate scan for each existing domain.
- Only one discovery process at a time can be executed by the Scanner. If you define multiple discoveries that are using the same CPM scanner, they will wait in the queue as pending and will be run one after the other.

Viewing the Status of the Discovery Process

Once you have defined a Discovery process, and while it is running, you can check its status in the Discovery Management page. The following statuses are displayed:

- **Pending** – The discovery is still waiting in the queue to be run and has not yet started.
- **Running** – The discovery is currently running and scanning for accounts.
- **Completed Successful** – The discovery was completed successfully and no errors were encountered during the scan.
- **Completed with errors** – The discovery was completed but that errors occurred.
- **Failed** – The discovery failed to run, usually this happens because of a fatal error, such as a failed connection to the Active Directory or a user with insufficient privileges. The discovery process will stop immediately and update its status to Failed.

For information about errors, refer to the CACPMScanner.log file in the PasswordManager\Logs folder. After you have solved the problem, create a new discovery process that will run again. For more information, refer to *CyberArk Central Policy Manager Scanner Logs, 504*.

Pending Accounts

The Pending Accounts page displays all the accounts that were discovered during multiple scans. Accounts that were previously onboarded into the Vault using the Add Account page in the PVWA, using the onboarding option in the Accounts Feed or using the AddAccount web service are not displayed in this list.

The accounts listed in this view are only onboarded to the system and will be managed automatically after you manually initiate the onboarding process.

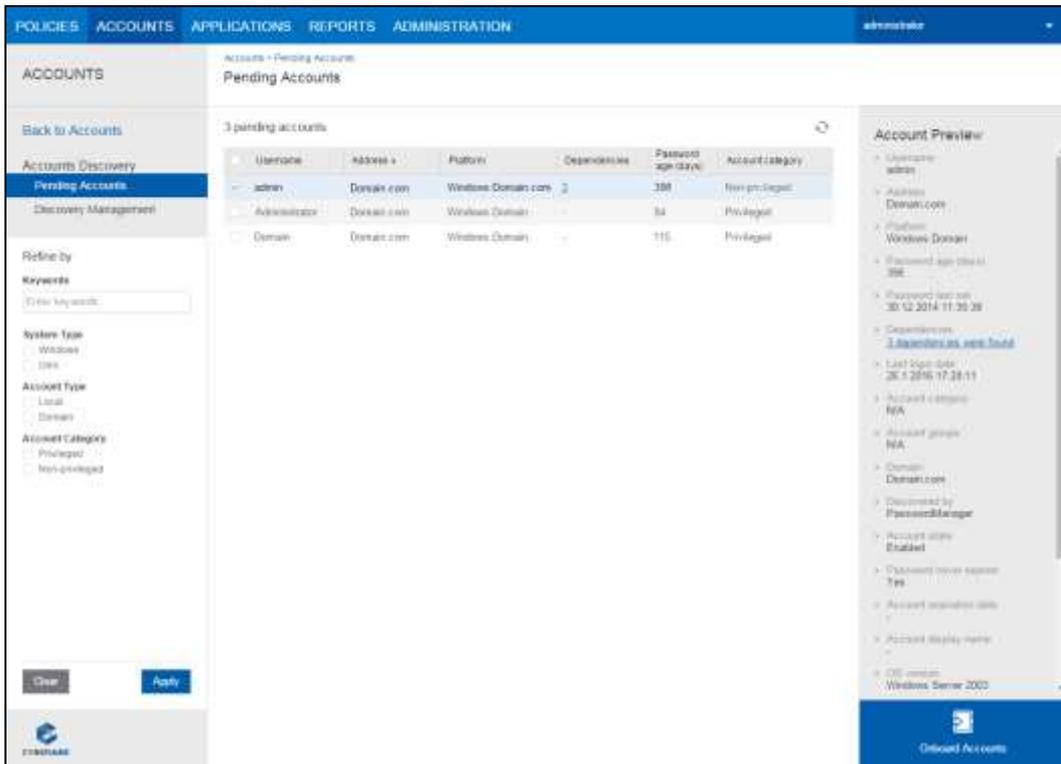
The screenshot displays the 'Pending Accounts' page in the CyberArk interface. The page title is 'Accounts - Pending Accounts' and 'Pending Accounts'. The main content area shows a table with 3 pending accounts:

Username	Address	Platform	Dependencies	Password age (day(s))	Account category
admin	Domain.com	Windows Domain	3	357	Non-privileged
Administrator	Domain.com	Windows Domain	-	84	Privileged
Domain	Domain.com	Windows Domain	-	115	Privileged

The left sidebar includes a 'Refine by' section with filters for Keywords, System Type (Windows, On), Account Type (Local, Domain), and Account Category (Privileged, Non-privileged). The right sidebar contains an 'Introduction to Pending Accounts' section with explanatory text and a 'Go to Discovery Management' link.

To View Account Details

- Select an account to view its details in the Account Preview pane.



The following table describes the information displayed in the Pending Accounts grid and the Account Preview pane. This information changes, depending on whether the selected pending account is a Windows or Unix account.

Column	Displays
Username	The login name of the account about which information was retrieved.
Address	The IP address/DNS of the discovered account.
Platform	The platform that the Discovery has attached to the discovered privileged accounts.
Password age	The current account password age, in days. This will appear as a fraction if the age is less than one day.
Password last set	The date and time the password was last set.
Dependencies	The number of dependencies that use the detected account. Click the specified number of dependencies to view the dependencies' details. For more information, refer to <i>Viewing Account Dependencies</i> , page 188. This is relevant for Windows accounts only. For Unix accounts, this column will always be empty.

Column	Displays
Last login date	<p>Displays the last date and time that the account was used for login.</p> <ul style="list-style-type: none"> ▪ Local accounts – The last date and time the account was used to log into the current computer. This is relevant for Windows and Unix accounts. Note: In Solaris, the last login date is N/A. ▪ Domain accounts – The last date and time the account was used to log into any computer in the domain. This is relevant for Windows accounts only.
Account category	<p>Whether the target account is privileged or non-privileged.</p> <ul style="list-style-type: none"> ▪ Windows accounts – This category only applies to local accounts. For domain accounts, N/A will be displayed. This column displays the following values: <ul style="list-style-type: none"> ▪ Privileged – Indicates that the account is a member of the Administrators or Power Users group. ▪ Non-Privileged – Indicates that the account is a non-privileged local or domain account. ▪ Unix accounts – This category only applies to local accounts. This column displays the following values: <ul style="list-style-type: none"> ▪ Privileged – Indicates the following: <ul style="list-style-type: none"> ▪ The account is a member of the GID=0 group ▪ The account is set to UID=0 ▪ Local account privileges have been escalated using the sudoers file, unless a sudo-replacement solution is used <p>Note: In AIX, this might indicate that the account has an "admin" attribute in the /etc/security/user file.</p> ▪ Non-Privileged – Indicates that the account is a non-privileged local account.
Domain	<p>The name of the domain where the account was discovered. This is relevant for Windows accounts only. For Unix accounts, this column will always be empty.</p>
UID or GID	<p>The unique ID of the account's user or group. This is relevant for Unix accounts only.</p>
Account groups	<p>The name of the local groups of which the account is a member. If the account does not belong to any local group, N/A is displayed. This column will be updated to include a list of groups of which the account is a member after each new discovery. This category applies to only local accounts for domain accounts N/A will be used.</p>

Column	Displays
Account type	The type of account. <ul style="list-style-type: none"> ▪ Local – Indicates a local account. This is relevant for Windows and Unix accounts. ▪ Domain – Indicates a Windows domain account.
Discovered by	The name of the CPM scanner that discovered the account.
Account state	The current state of the discovered account. This field contains one of the following four values, based on priority (from high to low): <ul style="list-style-type: none"> ▪ Disabled ▪ Enabled
Password never expires	Whether the password was configured never to expire. This indicates that the user will not be required to change their password based on the domain password policy.
Account expiration date	The date and time on which the account is configured to expire.
Account display name	The account display name as it appears in the account properties.
OS version	The operating system version. <ul style="list-style-type: none"> ▪ Windows accounts – The OS version that is defined in the computer's account in the Active Directory. ▪ Unix accounts – The OS version, including the type of Unix platform and its version.
Machine type	Whether the computer is a server or a workstation.
Account description	The account's description field as it appears in the account properties.
Organizational unit	The Organizational unit (OU) as defined in the Active Directory. This is relevant for Windows accounts only. For Unix accounts, this column will always be empty.
Discovery date	The date on which the account was discovered.

Updating Pending Accounts

Accounts that are displayed in the Pending Accounts list may have been rediscovered in repeated discovery processes and might have changed since they were initially discovered. When existing pending accounts are rediscovered before they are onboarded, the Pending Accounts list is updated to reflect changes, whether performed manually by the user or automatically by the system. These changes include:

- Password age
- Dependencies
- Last login date
- Account category
- Account expiration date
- Password never expires
- Account state (enabled/disabled)
- Account group

If a new dependency was discovered, the pending account dependencies will be updated with this new dependency. If the account was already onboarded through the Accounts Feed, newly detected dependencies will be automatically onboarded as well, in order to maintain a complete picture of the environment and simplify the onboarding process. For more information about onboarding pending accounts and their dependencies, refer to *Onboarding Accounts*, page 191.

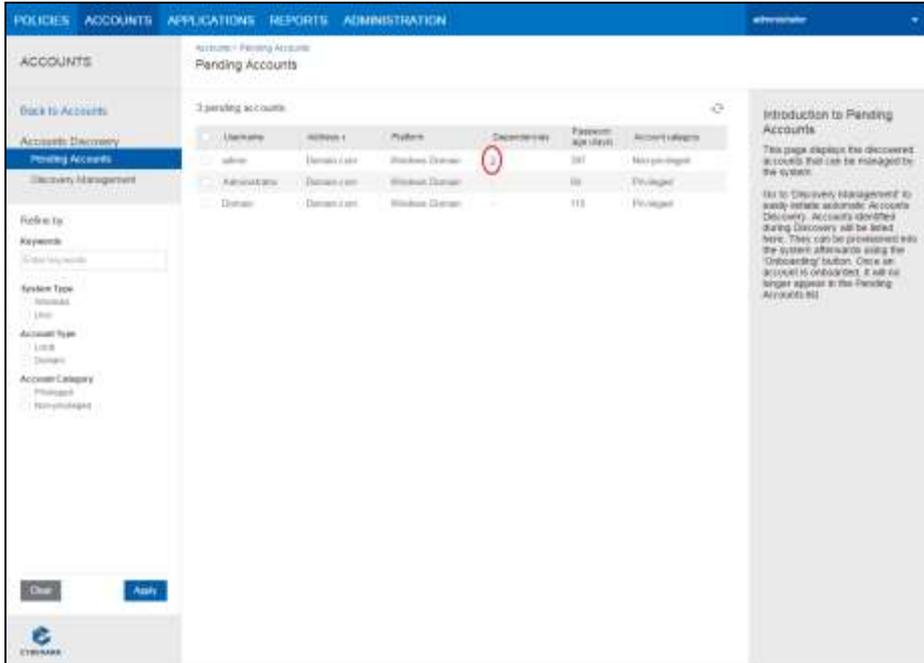
This updated information gives a clear and concise view of accounts in your environment in real time, and helps you decide what to onboard.

Viewing Account Dependencies

In the Pending Accounts page you can view the number of dependencies that use the detected accounts, as well as the details of each dependency, including the address where the dependency is configured and the type of dependency.

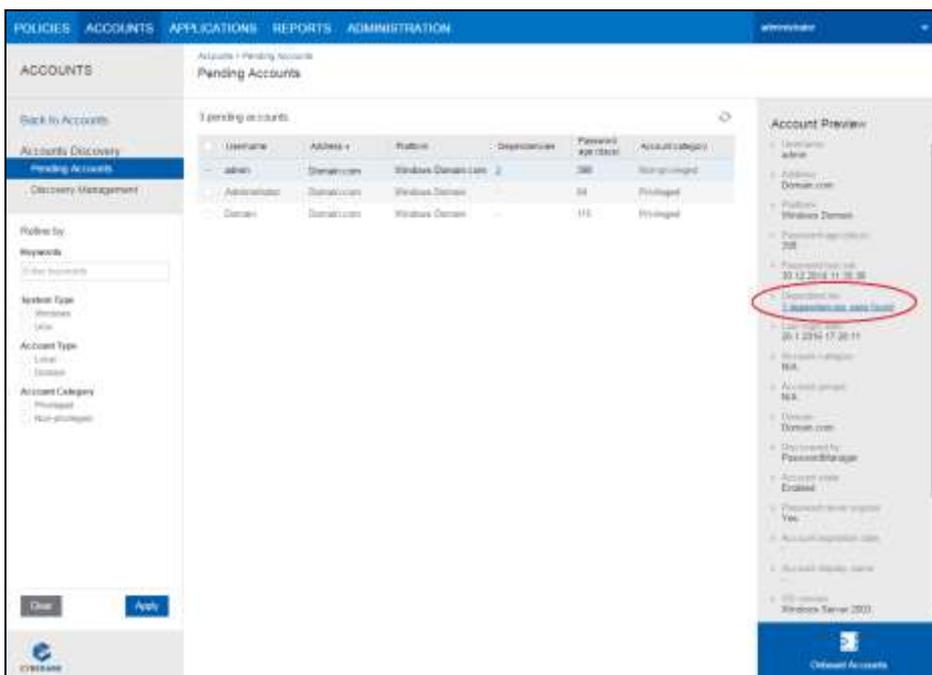
To View Dependencies

- In the Pending Accounts page, in an account row, click the displayed number of dependencies.

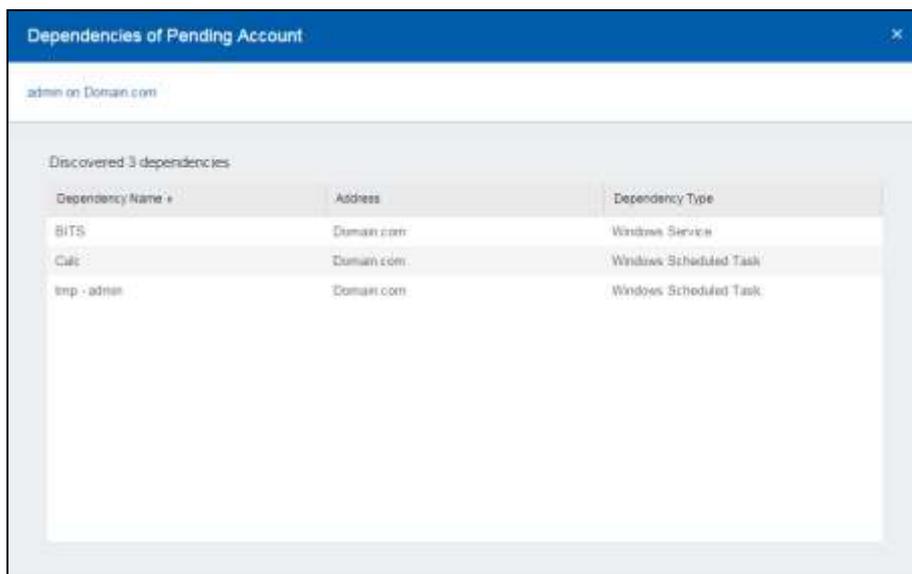


Or,

- In the Pending Accounts page, select an account; its details are displayed in the Account Preview pane.
- In the Account Preview pane, click the **Dependencies** link.



The Dependencies of Pending Account window appears and displays details about all the discovered dependencies for that account.



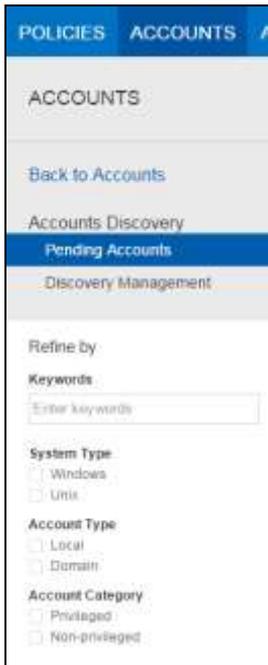
The following table below explains the columns in the Dependencies of Pending Accounts window.

Column	Specifies
Dependency Name	The name of the discovered dependency.
Address	The IP address/DNS of the discovered dependency.
Dependency Type	The type of dependency. For example, Windows Service or Windows Scheduled Task.

Search and Filtering Options

In order to reduce the scope of accounts displayed in the Pending Accounts grid, you can refine the list by using either the Keywords field or the other filter options. You can select filtering options to view results according to the selected filter, and then clear these filters to list all pending accounts.

- **Apply** – Clicking **Apply** displays a list of accounts in the grid according to the filter selected.
- **Clear** – Clicking **Clear** clears the list of filtered accounts and displays all the discovered accounts.



You can refine the filtering options by any of the following options:

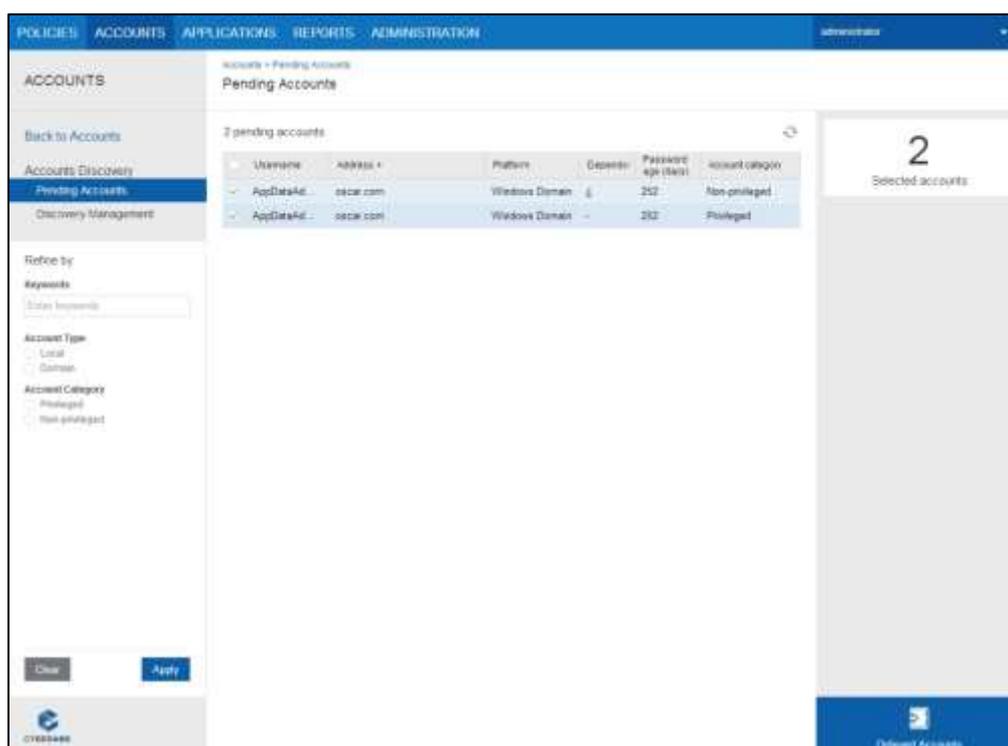
- **Keywords** – In the **Keywords** field, specify an account keyword to search for. Specify multiple keywords separated by a space. For example, **administrator Windows domain.com**.
- **System Type** – In the **System Type** area, select the relevant filter:
 - **Windows** – Select this option to view Windows accounts.
 - **Unix** – Select this option to view Unix accounts.
- **Account Type** – In the **Account Type** area, select the relevant filter:
 - **Local** – Select this option to view Windows local accounts.
 - **Domain** – Select this option to view Windows domain accounts.
- **Account Category** – In the **Account Category** area, select the relevant filter:
 - **Privileged** – Select this option to view privileged accounts.
 - **Non-privileged** – Select this option to view non-privileged accounts.

Onboarding Accounts

You can select a specific account or multiple accounts to onboard to the Vault, regardless of the number of accounts that were discovered during a scan. If an account contains dependencies, the dependencies are automatically onboarded with the account. A newly discovered dependency could potentially be non-legitimate or malicious. Therefore it is recommended to review and approve each newly discovered dependency to prevent such dependencies from being onboarded automatically by the system. When a discovery finds new dependencies associated to a domain account that was previously onboarded or already exists in the system, by default, the dependencies will automatically be onboarded as disabled for automatic CPM management. Local accounts will be onboarded as enabled for automatic CPM management. For more information about enabling dependencies, refer to *Resuming Automatic Management*, page 214.

To Onboard Accounts

1. In the Pending Accounts page, select the account(s) to onboard. If multiple accounts are selected, the number of accounts to be onboarded will be displayed in the Preview pane. If only one account is selected, then the Preview pane will display additional account details.



2. At the bottom of the Account Preview pane, click **Onboard Accounts**; the Onboard Accounts window appears.

3. Specify the Safe where the account will be stored when it is onboarded to the Vault:

- From the **Store in Safe** drop-down list, select a Safe.

Notes:

- Only Safes where the user is a member with the Add accounts permission are displayed in this list.
- Internal Safes are not displayed in this list.

Or,

- Create a new Safe:
 - i. From the Store in Safe drop-down list, select **Create New Safe**; the Create New Safe window appears.

- ii. In the Create New Safe window, enter a name for the new Safe and click **Create**; the new Safe is created and will be automatically selected in the Store in Safe list in the Onboard Accounts window.

Notes:

- Users require the Add Safes permission at Vault level to create a new Safe.
 - Safes that are created in the PVWA are based on properties specified in a Safe Template. Safe properties and access control can be configured afterwards. For further information, refer to *Updating Safe Properties in the PVWA*, page 66.
4. From the **Assign Platform** drop-down list, select the platform that will be associated with the onboarded account .

Notes:

- Only active platforms are displayed in this list.
 - Only platforms that can be associated with accounts in this Safe are displayed in this list.
5. In the Password section, select one of the following options:
- **Automatically reconcile password** – The CPM will reconcile the credentials for all selected accounts after they are onboarded to the Vault.
- This option is only enabled for platforms that are configured for account reconciliation. For more information about configuring platforms for credential reconciliation, refer to *Modifying Target Account Platforms*, page 109.

The screenshot shows a dialog box titled "Onboard Accounts" with a close button in the top right corner. Below the title bar, it says "Onboarding 2 selected accounts". The main content area is divided into two panels. The left panel, titled "Setup accounts onboarding", contains the text "The accounts will be onboarded with the related dependencies" and three configuration fields: "Store in Safe" with a dropdown menu showing "Select Safe or create new Safe", "Assign platform" with a dropdown menu showing "Select platform", and "Password" with a radio button selected for "Automatically reconcile password". A red circle highlights the "Automatically reconcile password" option. The right panel, titled "Introduction to Accounts Onboarding", contains a paragraph of text: "During this process, you define what to do with the accounts that were detected during Discovery. The service accounts will be automatically onboarded." At the bottom of the dialog, there are two buttons: "Cancel" on the left and "Onboard" on the right.

If the selected platform is not configured for reconciliation, you cannot select this option and a relevant message appears, as shown in the following example:

The screenshot shows the 'Onboard Accounts' dialog box. The title bar says 'Onboard Accounts'. Below the title bar, it says 'Onboarding 2 selected accounts'. The main content area is titled 'Setup account(s) onboarding'. It contains the following fields and options:

- 'Store in Safe': A dropdown menu.
- 'Assign Platform': A dropdown menu set to 'Windows Desktop Local Accounts'.
- 'Password': A section with two radio buttons:
 - 'Automatically reconcile password': This option is disabled and circled in red. Below it, a message reads: '* Reconcile is currently not set for this platform'.
 - 'Set a default password': This option is selected.
- 'Default Password': An input field.
- 'Confirm Password': An input field.

On the right side, there is a sidebar titled 'Introduction to Accounts Onboarding' with the following text: 'During this process, you define what to do with the accounts that were detected during Discovery. The service accounts will be automatically onboarded.'

At the bottom, there are two buttons: 'Cancel' (disabled) and 'Onboard' (active).

- **Set a default password** – You can define the password for all the selected accounts that will be set when they are onboarded to the Vault.

Specify the password that will be set in the selected accounts, and then confirm it.

Note: This sets passwords stored in accounts in the Vault, but does not reset actual passwords on target systems. For more information about synchronizing passwords in the Vault with passwords on target systems, refer to *Reconciling Passwords*, page 207.

Onboard Accounts

Onboarding 2 selected accounts

Setup account(s) onboarding

The account(s) will be onboarded with the related dependencies

Store in Safe:

Assign Platform: **Windows Desktop Local Accounts**

Password:

Automatically generate password

* Password is currently not set for this platform

Set a default password

Default Password:

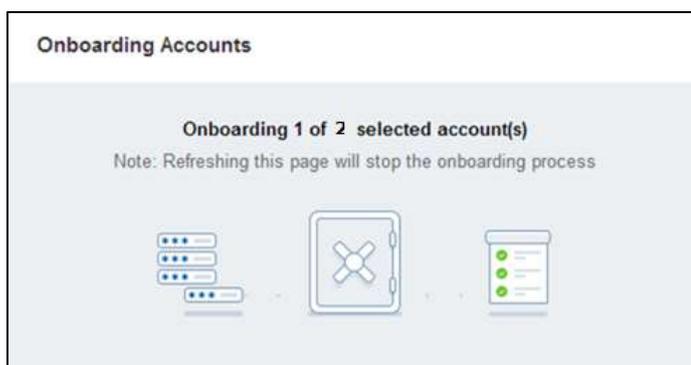
Confirm Password:

Introduction to Accounts Onboarding

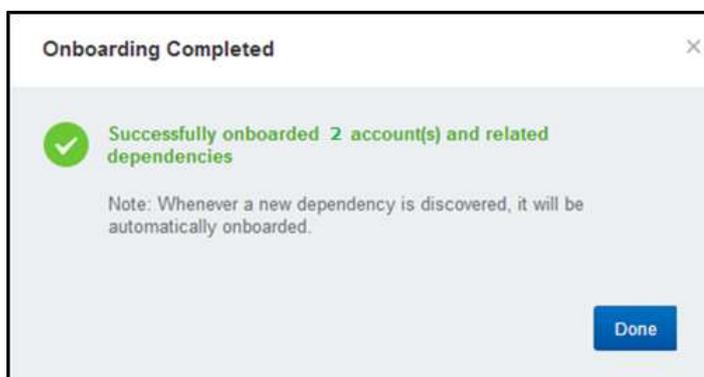
During this process, you define what to do with the accounts that were detected during Discovery. The service accounts will be automatically onboarded.

Cancel Onboard

6. Click **Onboard**; the onboarding process begins. The following window indicates its progress.

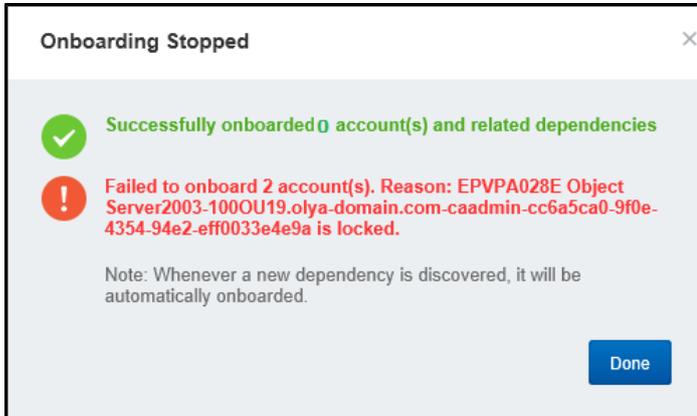


7. When the onboard process has finished successfully and the selected accounts and their dependencies have been onboarded to the Vault where they are stored securely and can be managed automatically, the Onboarding Completed message appears.



8. Click **Done** to acknowledge the successful onboard process.

If the system cannot onboard an account, the following message appears and the entire onboarding process stops. This does not affect the accounts that were successfully onboarded before this error occurred.



Configuring the Onboarding Process

To prevent dependencies that could be potentially non-legitimate or malicious from being automatically onboarded by the system, you can configure the workflow in such a way that any newly detected dependencies associated to domain accounts will need to be approved, including the account.

- The **OnboardNewDependencyAsDisabled** parameter, in the Accounts Feed section in the Web Access Options, defines whether to enable or disable this workflow. You can enable the dependencies and the account manually after the new dependencies have been onboarded. This behavior applies for domain accounts only. Dependencies associated to local accounts will be onboarded as enabled for CPM management. Specify one of the following:
 - **Yes** - When new dependencies are detected associated to an account that was already onboarded, these dependencies and their account will be automatically onboarded as disabled for automatic CPM management. This is the default value. For further information refer to *Disabling Automatic Account Management*, page 212.
 - **No** – Newly detected dependencies will always be onboarded as enabled for automatic CPM Management.

Automatic and Manual Account Management

Changing Passwords

Authorized users can change passwords that are stored in the Safe through the Password Vault Web Access. These passwords can be changed manually or replaced by a new password that is randomly generated by the Password Vault.

The CPM generates unique and highly secure passwords using the password policy and the random password generation mechanism. Therefore, passwords that are managed by the CPM do not need to be specified manually.

Passwords are changed automatically by the CPM in the following scenarios:

- **Expiration period** – Passwords that have an expiration period assigned to them are changed at the end of the specified period. This is configured in the Master Policy with the **Require password change every X days** rule.
- **One-time and exclusive passwords** – Passwords that are defined as one-time passwords or that are stored in Safes that are configured for Exclusive Account mode are changed after every use. These are configured in the Master Policy with the **Enforce one-time password access** and **Enforce check-in/check-out exclusive access** rules. When a one-time or exclusive account that is a member of a group has been used and the exclusive account has been checked-in to the Safe again, the password values for the entire group will be changed. These passwords are changed after accounts are checked-in manually or automatically after a minimum validity period defined in the Master Policy or based on the request timeframe described below.
- **Request timeframe** – Passwords that are accessed after a request for access during a timeframe has been confirmed will be changed automatically when the timeframe expires, unless manually checked-in before then.
- **Manual initiation** – after the user clicks ‘change’ or ‘reconcile’ and initiates an immediate change or reconcile CPM operation.

If you decide to specify a password manually, ensure that it is secure by using a combination of letters and numbers. If a predefined platform is enforced, the password complexity requirements are displayed in the Change Password window so that you know which types of characters to include or exclude. In addition, if the CPM prevents you from reusing a certain number of predefined passwords, that is displayed too.

Changing Passwords that are Managed Automatically by the CPM

Users who have the following Safe member authorizations can initiate a password change process by the CPM on multiple passwords:

- Initiate CPM password management operations

In addition, users with the following authorization can specify the new password that will be used:

- Specify next password value

To Select a Single Account

1. Display the Account Details page of the account that contains the password to change then, on the toolbar, click **Change**; the Change Password window appears.

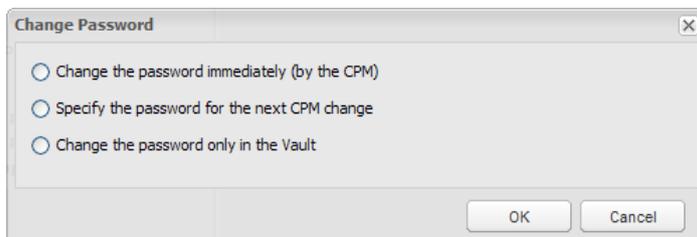
or,

1. In the Accounts page, select the account that contains the password to change.
2. On the toolbar, click **Manage** then, from the drop-down list, select **Change**; the Change Password window appears.

To Select Multiple Accounts

1. In the Accounts page, select the accounts that contain the password to change, or,
Display the Account Details page of the account that contains the password to change.
2. On the toolbar, click **Change**; the Change Password window appears.

Authorized users can change passwords in any of the following ways, according to the options in the Change Password window.



Select the relevant Change Password option:

- **Change the password immediately (by the CPM)** – Initiate an immediate password change in which the CPM will change the password to a new random password.
- **Specify the password for the next CPM change** – Specify the password that the CPM will use the next time it changes the password,
- **Change the password only in the Vault** – This option is disabled if multiple accounts are selected.
- **Accounts groups** – If any of the selected passwords in a member of an accounts group, additional options will be displayed.

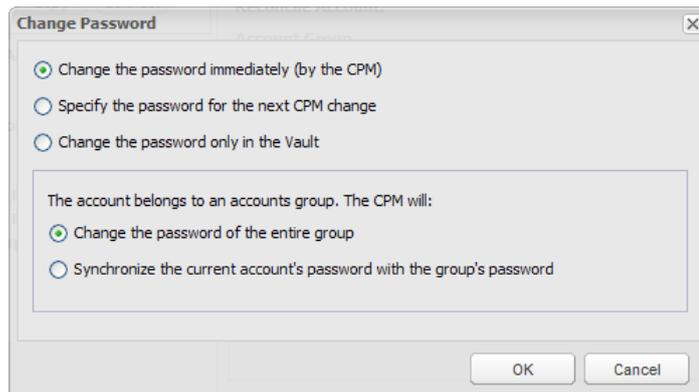
These processes are described below in more detail.

To Change the Password Immediately with the CPM

Authorized users can initiate an immediate password change in which the CPM will change the password to a new random password. To perform this task, users require the following Safe member authorizations:

- Initiate CPM password management operations
1. In the Change Password window, select **Change the password immediately (by the CPM)**.

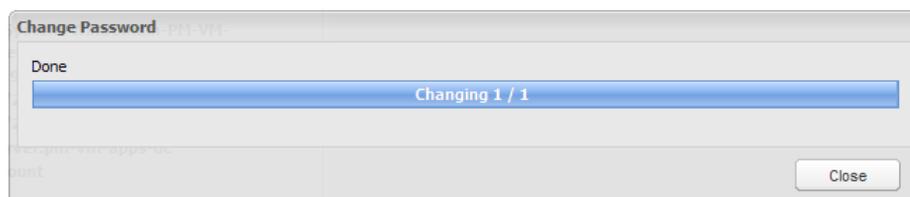
If the account belongs to an accounts group, additional options that are relevant to groups are displayed.



- To change the password in all the accounts that belong to the same group, select **Change the password of the entire group**.
 - To change the password in this account and synchronize it with the password specified in all the other members of the accounts group, select **Synchronize the current account's password with the group's password**.
2. Click **OK**; the CPM changes the selected password to a new random password that is generated automatically by the CPM according to the predefined password policy. Its progress is displayed in a progress bar.

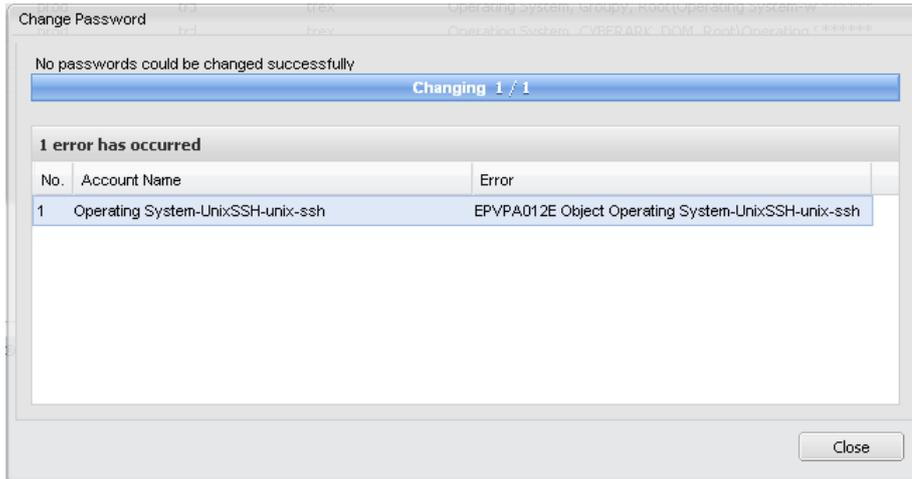
Note: During the procedure, you can click **Cancel** to cancel the password change.

3. When the password has been changed successfully, click **Close**.



The Change Password window is closed and the Account Details page is displayed again.

If the password could not be changed, an error message is displayed below the progress bar, explaining why the password(s) could not be changed.



Using the information in the error message, make the necessary changes so that you will be able to initiate this procedure again successfully.

To Specify the Password for the CPM to Use

Authorized users can initiate a password change process in an account that is managed by the CPM and specify the new password that will be used. The password can be changed in the Vault and reconciled on the remote machine by the CPM during the next CPM process. To perform this task, users require the following Safe member authorizations:

- Initiate CPM password management operations
- Specify next password value

1. In the Change Password window, select **Specify the password for the next CPM change**.



If a predefined password policy is enforced for the account being changed, the password complexity requirements of that policy are displayed.

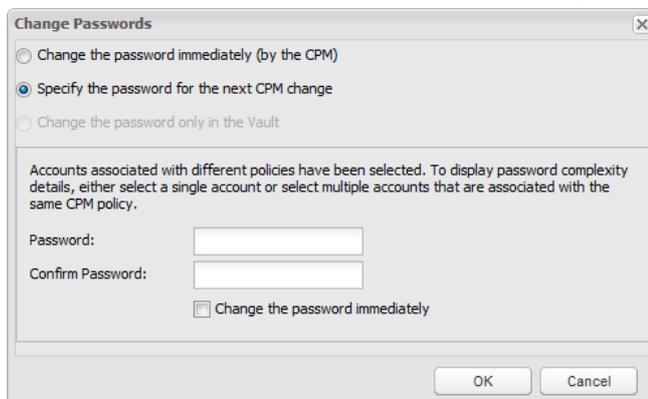
2. In the Password edit box, specify the password for the CPM to use.
3. In the Confirm Password edit box, type the password again to confirm it.
4. By default, **Change the password immediately** is selected. This will initiate an immediate password change by the CPM after you click **OK**.

To prevent an immediate password change, clear this checkbox.

5. Click **OK**; if the specified password contains leading and/or trailing white space character(s), a message appears indicating that they will automatically be removed. For more information about configuring this feature, refer to *Removing White Spaces from Passwords*, page 497.
6. Click **OK**; any white spaces are removed from the specified password and the specified password is either saved for the next time the CPM changes this password, or is used to change the password immediately. The progress of the CPM is displayed in the progress bar.



7. At the end of the password change process, click **Close**; the Change Password window is closed and the Account Details page is displayed again.
 - If the selected accounts are in Safes that are managed by a CPM that is configured to enforce password policy rules and the accounts are associated with different password policies, the following window appears.



- i. Click **Cancel** to return to the Accounts page, then either select a single account or select multiple accounts that are associated with the same platform.
- ii. Repeat this procedure to specify the password that the CPM will use when it changes the password.

To Change the Password in the Vault

Users who have the following Safe member authorization can change passwords that are managed manually:

- Update password value
1. In the Change Password window, select **Change the password only in the Vault**. If a predefined password policy is enforced for the account being changed, the password complexity requirements of that policy are displayed.

The screenshot shows the 'Change Password' dialog box with the following content:

- Three radio buttons at the top:
 - Change the password immediately (by the CPM)
 - Specify the password for the next CPM change
 - Change the password only in the Vault
- A text box containing password rules:

The new password must comply with the following rules:

Must contain:

 - At least 8 characters
 - At least 1 digits
 - At least 2 lower case characters
 - At least 2 upper case characters
 - At least 1 special characters

Must not contain:

 - Special characters: &
 - Identical passwords to the last 7 passwords
- Two text input fields labeled 'Password:' and 'Confirm Password:'.
- A dropdown menu labeled 'Generate Password'.
- 'OK' and 'Cancel' buttons at the bottom.

In addition, if the CPM is configured to prevent users from reusing a specific number of previous passwords, that is displayed too.

If the account belongs to an accounts group, additional options that are relevant to groups are displayed.

The screenshot shows the 'Change Password' dialog box with the following content:

- Three radio buttons at the top:
 - Change the password immediately (by the CPM)
 - Specify the password for the next CPM change
 - Change the password only in the Vault
- A text box containing password rules:

The new password must comply with the following rules:

Must contain:

 - At least 8 characters
 - At least 1 digits
 - At least 2 lower case characters
 - At least 2 upper case characters
 - At least 1 special characters

Must not contain:

 - Special characters: !?
 - Identical passwords to the last 7 passwords
- Two text input fields labeled 'Password:' and 'Confirm Password:'.
- A text box containing group-related options:

The account belongs to the accounts group Domain Managers. The CPM will:

 - Change the password of the entire group
 - Change the password of this account only
- A dropdown menu labeled 'Generate Password'.
- 'OK' and 'Cancel' buttons at the bottom.

2. In the Password edit box, specify the password for the CPM to use.
3. In the Confirm Password edit box, type the password again to confirm it.
4. To set an empty password, do not specify a password then select **Set an empty password**.
5. If the account belongs to an account group, select the relevant accounts group management option:
 - To change the password in all the accounts that belong to the same group, select **Change the password of the entire group**.
 - To change the password in this account only, select **Change the password of this account only**. This option will not change the passwords in any of the other members of this group.
6. To generate a password automatically, click **Generate Password**; the Change Password window expands to display the Generate Password options.

If the CPM is configured to enforce a password policy rule for the account being changed, you cannot change the password complexity rules.

Change Password

Change the password immediately (by the CPM)

Specify the password for the next CPM change

Change the password only in the Vault

The new password must comply with the following rules:

Must contain:

- At least 8 characters
- At least 1 digits
- At least 2 lower case characters
- At least 2 upper case characters
- At least 1 special characters

Must not contain:

- Special characters: !?
- Identical passwords to the last 7 passwords

Password:

Confirm Password:

The account belongs to the accounts group Domain Managers. The CPM will:

Change the password of the entire group

Change the password of this account only

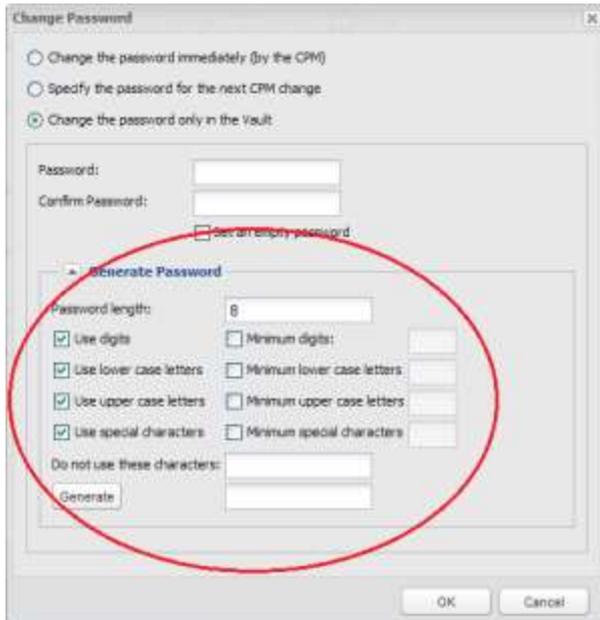
Generate Password

Auto-generate according to policy rules

Generate

OK Cancel

If the CPM is **not** configured to enforce a password policy rule, users can specify password complexity rules themselves, the following window appears:



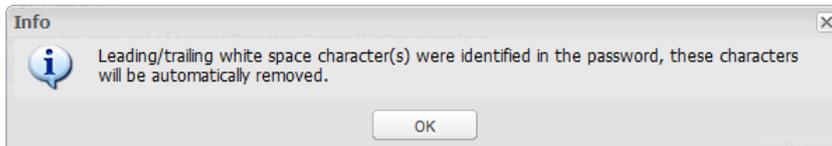
The default password policy settings are displayed, but you can specify the password criteria that will be applied to the new password.

7. Click **Generate**; a random password is generated using the specified password criteria. If the user has the 'Retrieve account' authorization, the new password is displayed.

Note: Digits are never placed as the first or last character of the password, regardless of the password policy or specifications.

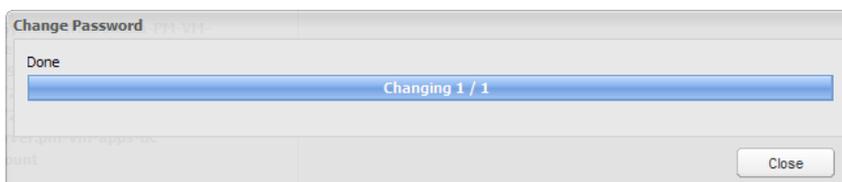
8. Click **OK**; the CPM changes the selected password to a new specified password. Its progress is displayed in a progress bar.

If you manually specified a password that contains leading and/or trailing white spaces, a message appears indicating that they will automatically be removed.



Click **OK**; any white space characters are removed from the specified password and the new account is added. For more information about configuring this feature, refer to *Removing White Spaces from Passwords*, page 497.

9. When the password has been changed successfully, click **Close**.



The Change Password window is closed and the Account Details page is displayed again.

If the password could not be changed, an error message is displayed below the progress bar, explaining why the password(s) could not be changed. Use the information in the error message to make the necessary changes so that you will be able to initiate this procedure again successfully.

Verifying Passwords

All passwords must be handled through the PVWA interface to ensure that the passwords on remote devices must be synchronized with the corresponding passwords in the Password Vault. However, if a password on the remote device is changed manually and not through the PVWA, it is no longer synchronized with its corresponding password in the Vault, and it becomes unavailable.

Whenever this happens, it is essential for the relevant personnel to be alerted as soon as possible so that they can identify the unsynchronized password and regain control over the remote device.

The CPM can verify password content on remote devices to ensure that they are synchronized with corresponding passwords in the Password Vault, and are valid and up-to-date. This process can either be managed automatically by the CPM or manually by an authorized user. If the password on the remote machine is not synchronized with the password in the Vault, the CPM alerts the user and can start a reconciliation process to synchronize the passwords. For more information about reconciling passwords, refer to *Reconciling Passwords*, page 207.

The CPM verifies all types of passwords, including group passwords and linked passwords. Passwords that are created automatically in the Vault as a result of auto-detection are verified immediately. The CPM does not lock passwords when it verifies them, whether in regular or exclusive mode.

Automatic password verification is determined by the following:

- The Master Policy defines how frequently passwords will be verified.
- Platform settings applied to the account determine how verification is initiated and the hours during which the verification process will take place.

Password verification is entirely independent of the password change process, which is configured separately.

For more information about password verification on other platforms, contact CyberArk support.

To Verify a Password Automatically

Users who belong to the Vault Admins group can configure password verification processes in the platform in the platform settings page. The Vault Admin group must be an owner of the CPM Safe with the following authorizations:

- Retrieve passwords
 - Update password value
1. Click **ADMINISTRATION** to display the **System Configuration** page, then click Platform Management to display a list of supported target account platforms.
 2. Select the platform to configure, then click **Edit**; the platform settings page for the selected platform appears.
 3. In the **Password Verification** parameters, specify the parameters that determine the automatic verification process for passwords linked to the platform.

The frequency of password verification processes is specified in the Master Policy.

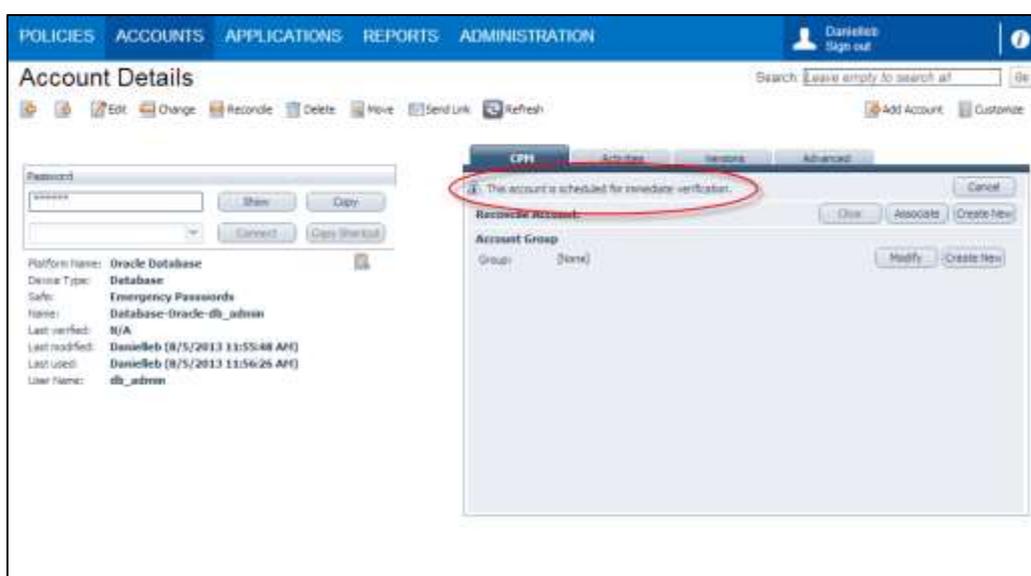
For more information about these parameters, refer to *Verifying Passwords*, page 379, in *Configuring Accounts for Automatic Management*.

To Verify a Password Manually

Although password verification processes can be scheduled to take place automatically at regular intervals, a verification process can also be initiated manually in the PVWA. Users who have the following Safe member authorizations can initiate a password verification process by the CPM:

- Initiate CPM password management operations
 1. In the Accounts list, click the account to verify and display the Account page.
 2. In the toolbar, click **Verify**; a confirmation box appears prompting you to confirm the password verification process.
 3. Click **OK**; the account is marked for verification and the CPM will verify it during the next password management cycle.

The CPM tab displays a message indicating that the account will be verified.



You can cancel the verification process any time before it occurs.

Reconciling Passwords

Passwords in the Vault must be synchronized with corresponding passwords on remote devices to ensure that they are constantly available. Therefore, the CPM runs a verification process to check that passwords are synchronized. If the verification process discovers passwords that are not synchronized with their corresponding password in the Vault, the CPM can reset both passwords and reconcile them. This ensures that the passwords are resynchronized automatically, without any manual intervention.

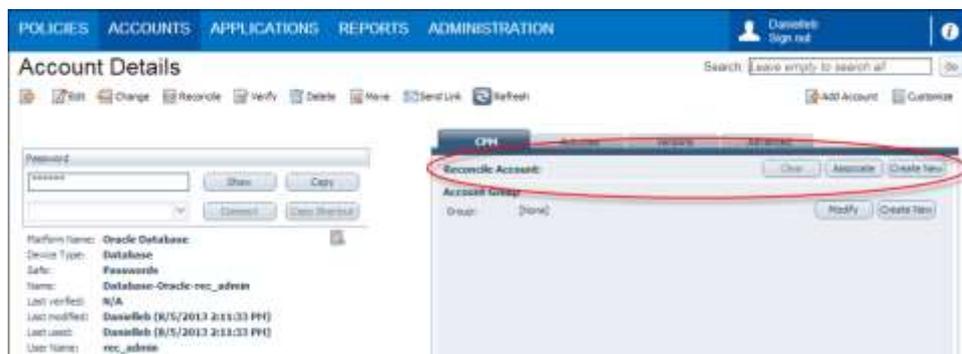
The platform contains rules that determine whether automatic reconciliation will take place when a password is detected as unsynchronized, or whether it is launched only through a manual operation by an end user/system admin. A reconciliation account password that will be used to reset the unsynchronized password can be defined either in the platform or at account level. This account can be stored in a separate Safe, where it is only accessible to the CPM for reconciliation purposes.

During password verification, the CPM plug-ins return a list of predefined errors to the CPM. Each platform specifies the specific errors that will launch a reconciliation process for passwords linked to that platform. This enables each enterprise to specify its own prompts for reconciling passwords and gives maximum flexibility to individual needs.

During password reconciliation, the unsynchronized password is replaced in the Vault and in the remote device with a new password that is generated according to the relevant platform. As soon as reconciliation is finished successfully, all standard verifications and changes can be carried out as usual. Users can see details of the last reconciliation process in the Operational Views in the Accounts List.

To Define a Reconciliation Account Password

- **At platform level** – All accounts attached to a specific platform will use the reconciliation account password specified in the platform. For more information, refer to *Reconciling Passwords*, page 379, in *Configuring Accounts for Automatic Management*.
- **At account level** – A reconciliation account password can be defined at account level and will override the account specified in the platform.
 1. Display the Account Details page for the account to link to a reconciliation account.



2. In the CPM pane, either link the current account to an existing account or create a new one.
 - **To link to an existing reconciliation account password:**
 - i. Click **Associate**; the Accounts list appears.
 - ii. Select an account to use as the reconciliation account password, then click **Associate**.
 - iii. The selected account is linked to the current account and its name appears in the CPM pane of the account's Account Details page.
 - **To create a new reconciliation account password:**
 - i. Click **Create New**; the Add Reconcile Account page appears.
 - ii. Define the new reconcile account password, then click **Link**; the new password is created and its name appears in the CPM pane of the password's Password Details page.

The screenshot shows the 'Add Reconcile Account' page in a web interface. The page title is 'Privileged Account Windows Desktop Local Accounts-win_admin-1.1.123: Add Reconcile Account'. The interface includes a navigation bar with 'POLICIES', 'ACCOUNTS', 'APPLICATIONS', 'REPORTS', and 'ADMINISTRATION'. The main form has the following fields and options:

- Store in Safe:** A dropdown menu with 'Select' as the current value.
- Device Type:** A dropdown menu with 'None' as the current value.
- Password Control:** A section with 'Password' and 'Confirm Password' input fields.
- Name:** A section with two radio buttons: 'Auto-generated (name pattern: {DeviceType}-{PolicyID}-{Address}-{Username}-{OS})' (selected) and 'Custom { }'. Below the radio buttons is a text input field.
- Disable automatic management for the account:** A checkbox that is currently unchecked, with a 'Reason' text input field below it.
- Buttons:** 'Link' (circled in red) and 'Cancel' buttons at the bottom left.

To Reconcile a Password Automatically

Users who belong to the Vault Admins group can configure password verification processes in the platform settings page. The Vault Admin group must be an owner of the CPM Safe with the following authorizations:

- Retrieve accounts (files)
 - Update password (file) value
1. Click **ADMINISTRATION** to display the **System Configuration** page, then click **Platform Management** to display a list of supported target account platforms.
 2. Select the platform to configure, then click **Edit**; the platform settings page for the selected platform appears.
 3. In the **Password Reconciliation** parameters, specify the parameters that determine the automatic reconciliation process for passwords linked to the platform.

For more information about these parameters, refer to *Reconciling Passwords*, page 379, in *Configuring Accounts for Automatic Management*.

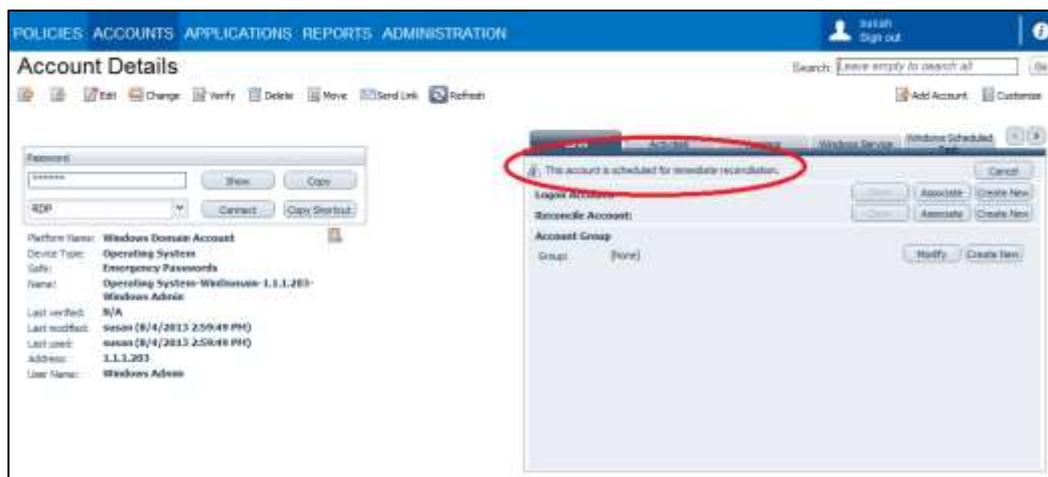
To Reconcile a Password Manually

Although password reconciliation processes can be scheduled to take place automatically at regular intervals, a reconciliation process can also be initiated manually in the PVWA by users who have the following Safe member authorizations:

- Initiate CPM password management operations

Users who belong to the Vault Admins group can configure password reconciliation processes in the platform settings page. The Vault Admin group must be an owner of the CPM Safe with the following authorizations:

- Retrieve accounts
 - Update password value
1. In the Accounts list, click the account to reconcile and display the Account Details page.
 2. In the toolbar, click **Reconcile**; a confirmation box appears prompting you to confirm the password reconciliation process.
 3. Click **OK**; the password is marked for reconciliation and the CPM will reconcile it during the next password management cycle. The CPM tab displays a message indicating that the password will be reconciled.



You can cancel the reconciliation process any time before it occurs.

Editing Account Properties

Authorized users can edit properties of existing accounts in the PVWA. Different Safe member authorizations enable users to perform different tasks in the Safe on accounts.

Safe members with the following authorization can update account properties:

- Update password properties

Safe members with the following authorization can rename accounts:

- Rename accounts

Safe members with the following authorization can move accounts to a different folder:

- Move accounts/folders

To Edit Account Properties

1. In the Accounts list, select the account to edit, then click **Edit**; the Edit Account window appears.

The screenshot shows the 'Edit Account' window for a Windows Desktop Local Account. The window title is 'Edit Account: Windows Desktop Local Accounts-win_admin-1.1.123'. The interface is divided into several sections:

- Store in Safe:** Device Type: Operating System (Change to: Operating System dropdown); Platform Name: Windows Desktop Local Accounts (Change to: Windows Desktop Local Accounts dropdown).
- Emergency Passwords:** Address: 1.1.123 (Change to: text input); User Name: win_admin (Change to: text input).
- Optional Properties:** Logon To: (Change to: text input); Location: (Change to: text input); Owner Name: (Change to: text input).
- Disable automatic management for this account:** Reason: (text input).
- Show advanced section:** (link).
- Buttons:** Save, Cancel.

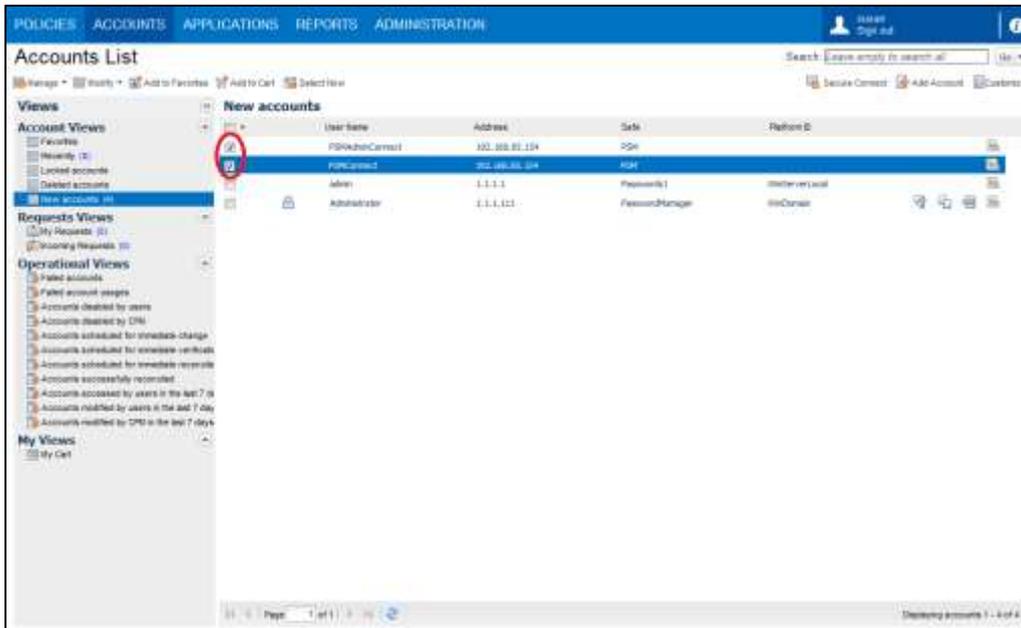
2. Change the account properties as required.
3. To change the name of the account or the folder where the account is stored in the Safe, click **Show advanced section** then specify the new account properties.
4. Click **Save**; the account properties are changed in all the selected accounts.

Editing Multiple Accounts

You can edit the properties of several accounts simultaneously to new common properties. You can select any type of accounts, regardless of the devices where they are used or their current properties. For example, you can change all the accounts on one IP address, or assign several accounts to a group at the same time.

To Edit Multiple Accounts

1. In the Accounts list, select the accounts to edit.



2. From the **Modify** drop-down menu, select **Edit**; the Edit Accounts window appears.
3. Click **Show** to display the current values of the selected accounts.
4. Change the relevant properties, then click **Save**; the account properties are changed in all the selected accounts.

Disabling Automatic Account Management

You can disable automatic account management and prevent the CPM from changing the password value when you don't want it changed for any reason, for example, when a device is temporarily unavailable.

To Disable Automatic Management

1. Display the Account Details window for the account to disable.
2. Click **Edit**; the Edit Account window appears.
3. Select **Disable automatic management for this account** and specify a reason. The reason is optional.

The screenshot shows the 'Edit Account' window for 'Windows Desktop Local Accounts-win_admin-1.1.1.124'. The window has a blue header with navigation tabs: POLICIES, ACCOUNTS, APPLICATIONS, REPORTS, ADMINISTRATION. The user 'DanielleB' is logged in. The main content area is divided into sections for 'Emergency Passwords', 'Optional Properties', and 'Optional Properties'. The 'Optional Properties' section has a checkbox labeled 'Disable automatic management for this account' which is checked and circled in red. Below this checkbox is a text field containing the reason: 'This IP address is currently unavailable'. At the bottom left, there are 'Save' and 'Cancel' buttons.

4. Click **Save**; this account is disabled and the Account Details window is displayed.
5. On the CPM tab, a message appears indicating that automatic management for this account has been disabled.

The screenshot shows the 'Account Details' window for 'Windows Desktop Local Accounts-win_admin-1.1.1.124'. The window has a blue header with navigation tabs: POLICIES, ACCOUNTS, APPLICATIONS, REPORTS, ADMINISTRATION. The user 'DanielleB' is logged in. The main content area is divided into sections for 'Password', 'RDP', and 'Account Details'. The 'Account Details' section has a 'CPM' tab selected. A message in the CPM tab reads 'Automatic management for this account was disabled by the user.' and is circled in red. Below the message are sections for 'Logon Account', 'Reconcile Account', and 'Account Group'. At the bottom left, there are 'Edit', 'Change', 'Delete', 'Move', 'Send Link', and 'Refresh' buttons.

6. Click **More details** to display the reason why the account was disabled.

Automatic Error Handling

You can also configure the CPM and control the way the CPM manages accounts when they cannot be changed. In addition to managing retry attempts, these configurations also enable the CPM to identify error types.

The CPM will disable automatic management for specific accounts in the following situations:

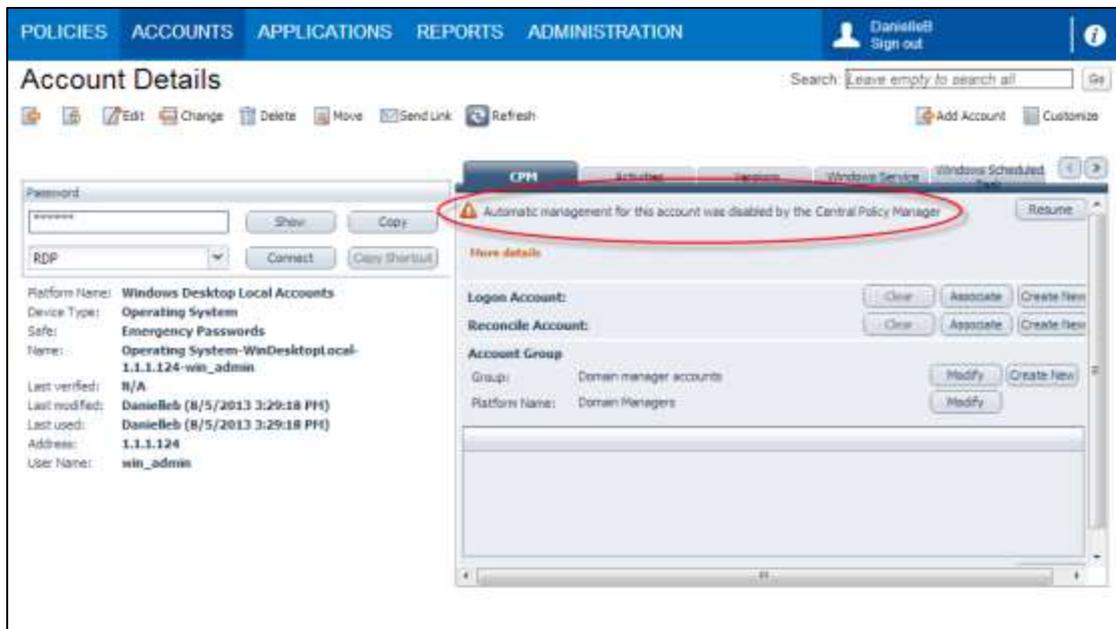
- If the CPM cannot change the password and has reached the maximum number of retries that is specified in the platform.
- If the CPM cannot change the password due to missing or incorrect information. In this situation, the CPM will not succeed in changing the password, regardless of the number of retries. On each platform, different situations cause the CPM to disable automatic account management. These situations are listed with a code number in the platform that is applied to the account.

In both the above situations, the CPM will specify the reason why the password cannot be changed. This reason appears in the Account Details window where the user can view it and intervene to resolve the problem.

Viewing CPM Retries

When a password cannot be changed, but the maximum number of retries has not yet been reached, a message appears in the CPM tab of the Account Details page indicating that the CPM cannot change the password.

At this point, automatic management has not been disabled. However, when the CPM reaches the maximum number of retries, the password will be disabled and an additional message will appear indicating that automatic management has been disabled for this password.



- Click **More details** to display the reason why the password was disabled.

Resuming Automatic Management

Automatic management for passwords can be re-enabled manually in the Password Details window. If automatic management was disabled manually because of an administrative reason, it can be restored whenever the CPM is required to manage the password again. However, if the automatic management was disabled due to an error, it is essential to resolve the problem before re-enabling automatic management so that the CPM can manage the password properly.

To Find Passwords with Disabled Automatic Management

In the ACCOUNTS page, select one of the following operational views to display a list of disabled accounts:

- Accounts disabled by users
- Accounts disabled by CPM

To Resume Automatic Account Management

You can resume automatic account management in one of the following ways:

- In the list of disabled accounts, select the accounts to re-enable and click **Resume**; the PVWA re-enables automatic management for these accounts.

Or,

1. Display the Account Details window for the account whose automatic management has been disabled; on the CPM tab, a message appears indicating that automatic management is disabled for that account.
2. On the CPM tab, click **Resume**; automatic account management is re-enabled for the account and its password will be changed the next time the CPM changes passwords.

Account Groups

The CPM can manage groups of accounts, so that all the passwords contained in the members of a group are changed together. This password can either be generated randomly by the CPM or specified by the user. After the password change process, all the members of the account group will have the same password.

The group is assigned to a platform that determines when the password will be changed and the restrictions that the password will have. Each group member will be changed by the plug-in specified in the platform assigned to the account.

The following example demonstrates a typical situation in which an account group enables successful password administration that combines security with convenience.

A particular database that contains extremely important information is replicated for use in the occurrence of a disaster recovery. This database is accessed with administrative passwords that are changed weekly or monthly, according to the organization's password policy. As the applications that use the database are automatically transferred to work with one of the replications in a failover situation, all the user passwords must be the same in all the replications. These requirements put a burden on the administrator to change the password weekly or monthly, as well as on the users who must remember new passwords regularly.

Using an account group, the administrator can create an account for each database replication and link them to a group manager. The administrator can manually specify an easy-to-remember password for the group manager that will automatically replace the password in each group member.

In this way, the CPM ensures that the organization's password policy of changing the password weekly or monthly is upheld, whilst enabling users to remember easy passwords.

An account group may contain an unlimited number of members who can be related to different platforms with different plug-ins. You can add multiple members to a group with different platforms.

In account groups that contain one-time or exclusive passwords, all the members of the group will be changed automatically after the one-time password has been used or after an exclusive password has been checked in. For more information about one-time or exclusive accounts groups, refer to *Managing One-time and Exclusive Account Groups*, page 219.

Creating Platforms for Groups

Account Groups require two types of platforms:

- **Group Manager platform** – Defines how the accounts in the group will be managed.
- **Group Member platform** – Group members can be individually associated with any platform that defines their use.

When you create an account group, first create the group manager platform, then create the group members and link them to the group. Users who are members of the Vault Admins group can manage account group platforms.

To Create a Group Manager Platform

1. Click **ADMINISTRATION** to display the **System Configuration** page, then click Platform Management to display a list of supported target account platforms.
2. Select **Sample Password Group Platform**, then click **Duplicate**; the Duplicate Platform window appears.
3. Type the name and a description of the new group platform, then click **Save & Close** to create the new platform.
4. Select the new group platform, and then click **Edit**; the configuration page for the selected platform appears.
5. Specify the group manager account management properties. These parameters determine how the CPM will manage members of the account group. Most of these parameters cannot be defined for individual group members and must be defined in the group manager.

The required properties are described in the following table. For a complete list of parameters that can be set in platforms for account groups, refer to *Appendix D: Managing Platforms for Groups*, page 980.

Note: For more information about configuration auto-detection for account groups, refer to *Managing Auto-Detected Accounts with Groups*, page 417.

Property	Indicates ...
Automatic Password Management General Properties:	
SearchForUsages	Whether or not CPM will search for copies of the account after it successfully changed and synchronized them. Specify Yes .
General Properties:	
Status	Indicates whether a platform is active or inactive. For more information, refer to the Privileged Account Security Implementation Guide.

6. Click **Apply** to save the new configurations and apply them immediately, or,

Click **Save** to save the new configurations and apply them after the period of time specified in the **RefreshPeriod** parameter.

To Create a Group Member Platform

Group member accounts can be associated with any platform other than the group manager platform.

1. Define the platform that will be applied to group members. For more information about adding and customizing platforms, refer to *Managing Target/Service Account Platforms*, page 108.
2. Add a new account as described in *Adding Accounts*, page 122, and associate it with the platform that defines its use.

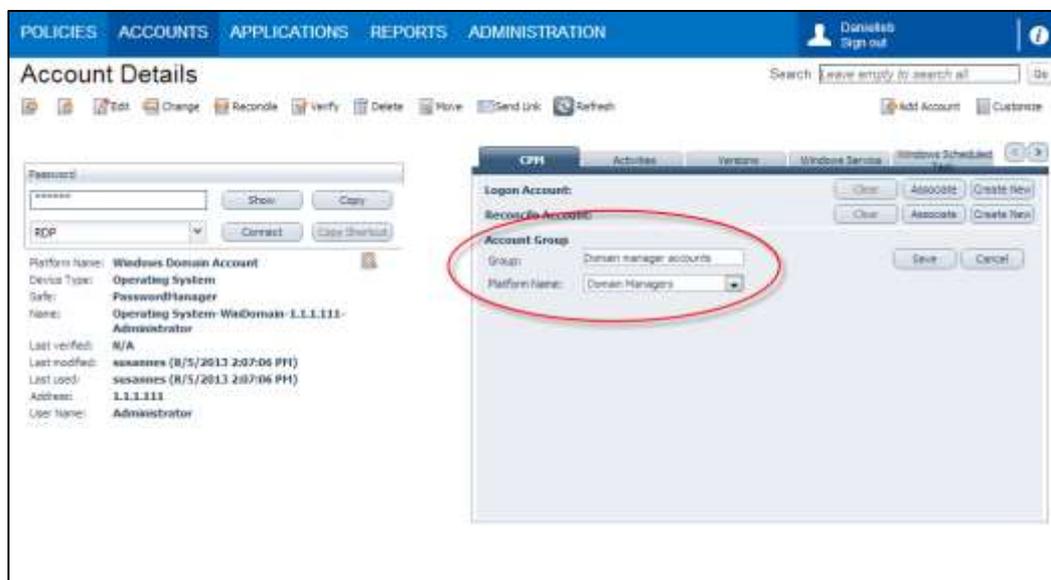
Note: All members of account groups must be stored in the same Safe.

Defining Account Groups

After the group manager platform and individual platforms have been created, you can define account groups.

To Define a Group

1. Display the Account Details window for an account that will belong to a group.
2. In the CPM tab, in the Account Group section, click **Create New**; the Account Group fields appear.
3. In the Group field, specify the name of the group
4. From the **Platform Name** drop-down list, select the group platform that you created for this group. This list is filtered and only active platforms for this type of group are displayed.

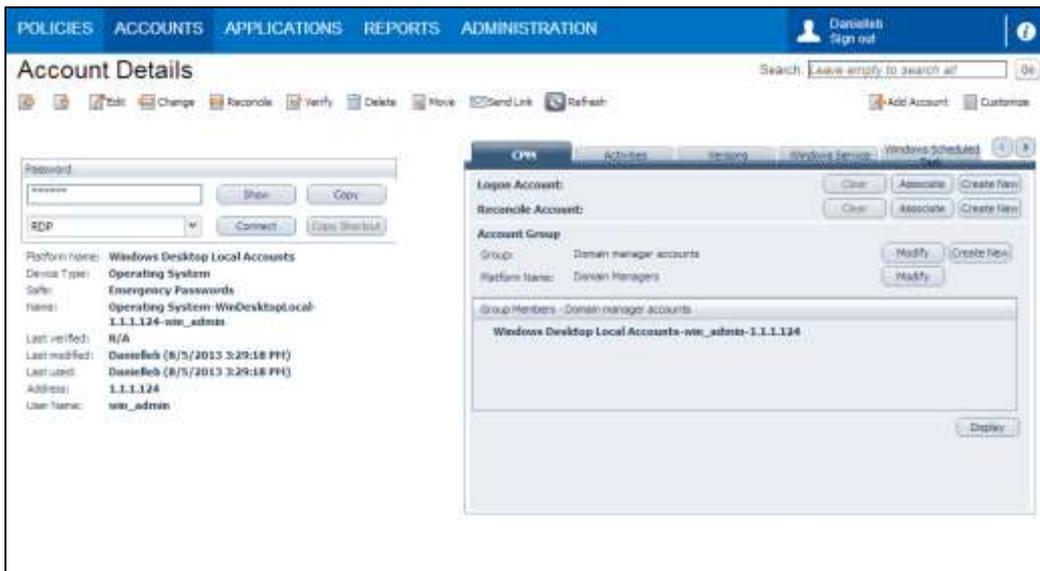


5. Click **Save**; the Group is created and the Group Member is added to it.
6. Click **Display** to view all the accounts that belong to the Account Group.

Note that the current account does not appear in the Group Members list. For more information, refer to *Viewing CPM Details*, page 497, in *Configuring the PVWA*.

To Add Members to a Group

1. Add a new account and click **Save**; the Account Details page appears,
or,
In the Accounts List, click an existing account; the Account Details page appears.
2. In the CPM pane, click **Modify**; the Group field appears.
3. From the Group drop-down list, select the name of the account group that this password will become a member of, then click **Save**; the account is attached to the specified account group.
4. Click **Display** to view all the accounts that belong to the Account Group.



Note that the current account does not appear in the Group Members list. For more information, refer to *Viewing CPM Details*, page 497, in *Configuring the PVWA*.

For information about modifying group passwords automatically or manually, refer to *Changing Passwords*, page 197.

Managing One-time and Exclusive Account Groups

Groups that are made up of exclusive accounts or that contain one-time accounts must be configured to ensure that group management processes maintain the status of these accounts.

For more information about creating one-time and exclusive accounts, refer to *Accounts Check-out and Check-in*, page 246.

Managing Exclusive Accounts Groups

All the accounts in exclusive accounts groups are locked when any account in the group is retrieved. Each account can only be retrieved after it has been changed and released.

After this type of group is released, all members of the group will be changed before they are released. From the time when the password change process begins until the last password in the account group has been changed, the group is not fully available. This means that group members that have not yet been changed, cannot be retrieved.

The Master Policy defines one-time and exclusive accounts. In addition, the following platform settings ensure that these accounts are maintained as exclusive accounts in groups:

- To ensure that all the accounts in the group will only be released after their passwords have been changed, in both the group manager platform and the group member platform, set the **UnlockIfFails** parameter to **No**.
Note: When this parameter is set to **Yes**, if the password change process fails, the account will be unlocked and can be accessed, but it will not have been changed.
- To ensure that accounts will be released immediately after they are changed, in the group member platform, set the **ResetOverridesMinValidity** parameter and the **ResetOverridesTimeFrame** parameter to **Yes**.
- In the group manager platform, make sure that the value of the **ImmediateInterval** parameter is higher than the value of this parameter in the group member platform.
- To specify the number of minutes between the last time that an account was retrieved and when the entire group is replaced, in the group member platform, specify the **MinValidityPeriod** parameter.

Managing One-Time Accounts in Groups

When a one-time account in a group is retrieved, the group isn't locked and all other members are still available for other users. However, after the time specified in the **MinValidityPeriod** elapses, all the passwords in the entire group of accounts are changed.

- To make sure that the passwords for all members of a group are changed after any member account is retrieved, in the group member platform, set the **OneTimePassword** parameter to **Yes**.
- To specify the number of minutes between the last time that an account was retrieved and when the entire group is replaced, in the group member platform, specify the **MinValidityPeriod** parameter.

Linked Accounts

Some platforms require additional passwords in order to manage other passwords. For example, a user who cannot log on remotely to a Unix machine to change a password requires an additional password in order to log onto the target Unix machine using a different user, and then take over the user's ID so that the password can be changed.

Information about additional accounts is transferred automatically to the third party plug-in that manages the passwords.

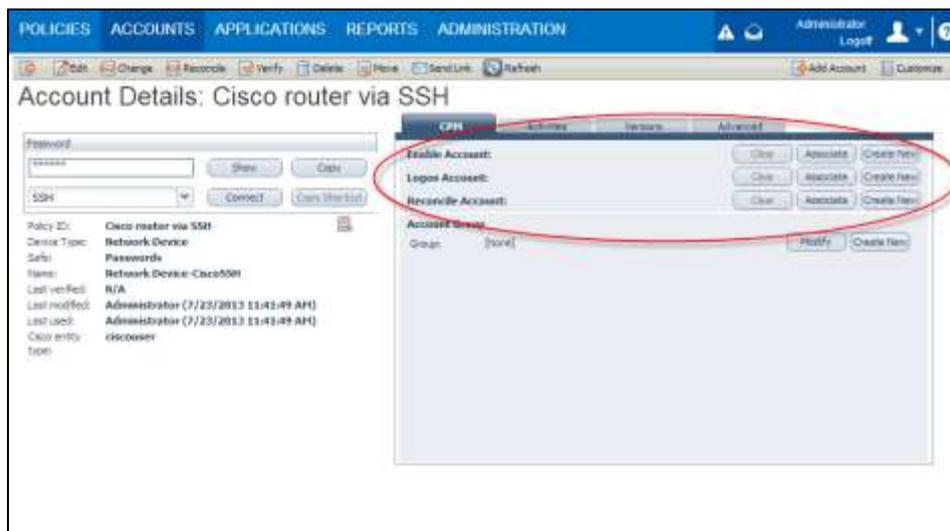
You can associate the following linked accounts:

- **Enable account** – An account that contains the password that will enable the CPM to switch to 'enable' mode and perform tasks on a remote machine.
- **Logon account** – An account that contains the password that is required to log onto a remote machine to perform a task using the regular account. Logon accounts can be configured either at platform level or at account level.
Note: In PSM and PSM SSH Proxy connections, the logon account can only be defined at account level.
- **Reconcile account** – An account that contains the password used in reconciliation processes. For more information about reconciling passwords, refer to *Reconciling Passwords*, page 207.

To Link an Additional Account to an Existing Account

Safe members require the following authorizations for this task:

- Retrieve accounts
 - Update password properties
1. Add a new account and click **Save**; the Account Details window appears, or,
In the Accounts List, click an existing account; the Account Details window appears.
 2. In the CPM pane, in the accounts section, you can associate either a logon account or a reconciliation account.
 - If a default logon account has been configured for the platform that manages this account, that account is listed. You can associate another logon account or leave the default account as it is.
 - If a default logon account has **not** been configured:
 - i. Click **Associate** next to the type of account to link; the Linking Account window appears. This window lists the frequently used accounts. If the account that you require does not appear in this list, do a search for the required account.



- ii. Select the required account, then click **Associate**; the selected account is linked to the original account and the details of the linked account are listed in the Logon credentials section.

To Create a New Account and Link it Immediately

Safe members require the following authorizations for this task:

- Retrieve accounts
 - Add accounts
1. In the Accounts List, click an existing account; the Account Details window appears.
 2. In the CPM pane, in the additional accounts section, click **Create New**; the Add Account Credentials window appears.

For more information about adding new accounts, refer to *Creating Accounts*, page 122.

3. Specify the account properties for the new linked account, then click **Link**; the new account is created and linked immediately to the original account.

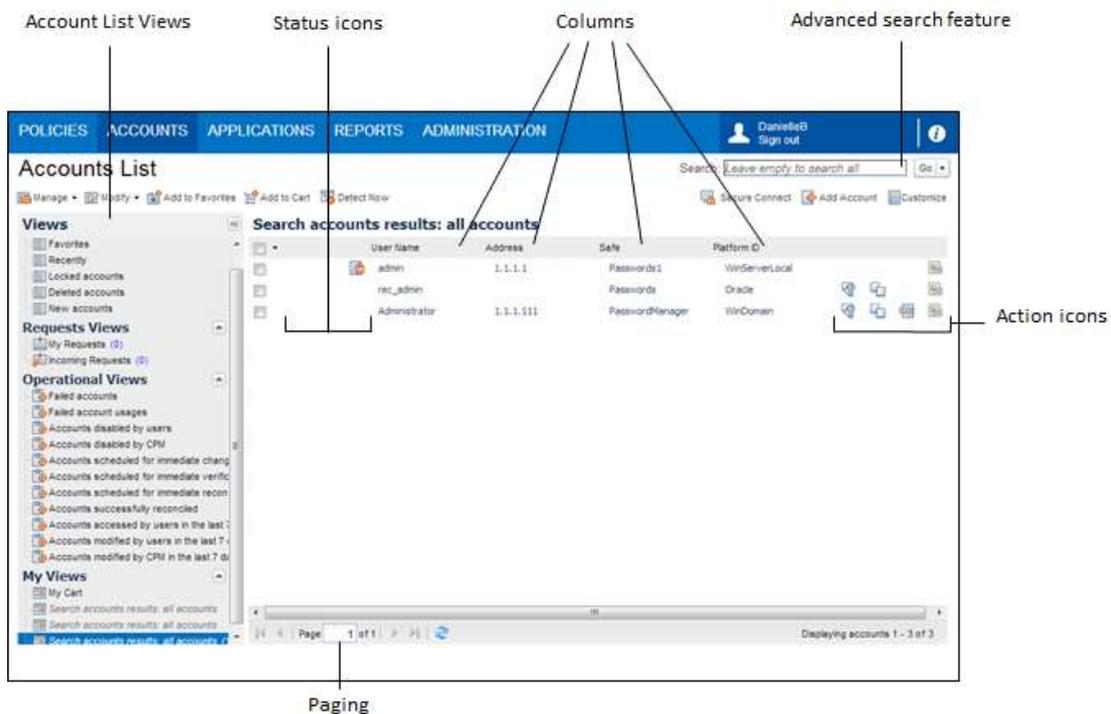
The details of the linked account are listed in the additional accounts section.

Account Access Workflows

Accessing Accounts and Service Accounts

The Accounts page displays your accounts in a set of views that you can display, sort, and access quickly and easily. These multiple views enable you to display accounts according to predefined criteria, based on account and operation status. You can also define customized views according to your own requirements and save them, so that you can display search results in one quick step. You can perform a variety of management tasks in each list of accounts, depending on your own permissions for accessing these accounts.

These different views, available at your fingertips, and the ability to manipulate entire lists, combined with the multiple actions that you can initiate on the same page increase usability and streamline account management, making it intuitive and efficient.



The main features and functionality of the Account page are described below.

Account Views

The selected view determines the accounts that are displayed in the Accounts List. The first time you display each view, the number of accounts in that view is displayed with the view title. This is updated each time you display that view.

The Accounts page is divided into the following views:

- **Account View** – This view enables you to list accounts according to their status. This view offers the following statuses:
 - **Favorites** – You can add accounts to the Favorites list so that you can view accounts that you use frequently at a single click. This list is personal to each user, and accounts are added and removed from it manually.
 - **Recently** – A list of the accounts you recently used in the PVWA. The number of accounts can be customized according to your needs, depending on the number of recent accounts you need to access regularly.
 - This list includes the following account activities:
 - Newly added accounts
 - Manual password changes
 - Show password
 - Copy password
 - Connect to a remote machine using a regular connection
 - Connect to a remote machine through a PSM connection

Note: The first time you display the Recently view after upgrading the PVWA, the accounts you used in the previous version are added to the list of accounts used in the current version, giving you a complete list of recently used accounts.
- **Locked accounts** – A list of accounts that are locked by your user.
- **Deleted accounts** – A list of accounts that were deleted. Only authorized users can see this view and undelete accounts. This replaces the Archive link in previous versions.
- **New accounts** – A list of new accounts that have been added to the Privileged Account Security solution.
- **Request view** – This view enables you to view accounts according to requests and confirmation. It includes the following lists:
 - **My requests** – A list of requests for access to Safes or accounts, created by your user, and their status.
 - **Incoming Requests** – A list of requests waiting for confirmation. Only users who are authorized to confirm requests can display this view.

- **Operational View** – Users who are members of the PVWAMonitor group can display different operational views, which includes lists of accounts and service accounts at different stages of operations and with various statuses. In addition, you can initiate mass operations. This view offers the following operations:
 - **Failed accounts** – A list of accounts that could not be managed successfully by the CPM, resulting in an error.
 - **Failed account usages** – A list of service accounts that could not be managed successfully by the CPM, resulting in an error.
 - **Disabled accounts** – Lists of accounts that have been disabled manually by users or automatically by the CPM, and are not currently managed automatically by the CPM.
 - **Successfully reconciled accounts** – A list of accounts that were successfully reconciled by the CPM.

The following views display accounts that are managed automatically by the CPM. It includes accounts that are marked for an activity manually by a user but are changed automatically by the CPM.

- **Scheduled accounts** – Lists of accounts that are scheduled for immediate change, verification, or reconciliation by the CPM. These lists also include accounts that were scheduled for an immediate task but are no longer managed by the CPM because the Safe where they are stored is no longer managed by the CPM.
- **Accessed accounts** – A list of accounts that were accessed by users during the previous seven days.
- **Modified accounts** – Lists of accounts that were modified manually by users or automatically by the CPM during the previous seven days.
- **My Views** – This view displays personalized lists defined by the user, and includes a customized cart and search results. Users can create and modify these views, and save them for future reference.
 - **My Cart** – The user's cart enables you to perform mass operations across multiple account views. You can add different accounts to your cart, according to the operation to perform, and regardless of specific searches and the order in which search results are displayed. This list is personal to each user and is cleared when the user logs off from the PVWA.
 - **Customized views** – A set of customized views that users can save to display in one quick step whenever necessary. These views include search results, service accounts of a specific account, and dashboard links. You can save these personalized views and even mark one so that it is displayed as the default view the next time you log on and display the Accounts page.

Managing Accounts

A variety of drop-down lists and buttons enable you to perform multiple actions on the displayed accounts, according to your permissions in the Vault.

- **Toolbar** – The accounts list toolbar displays the actions that can be performed on the accounts displayed in each list. The drop-down lists and buttons differ according to the list that is displayed as well as according to your permissions in the Vault, so you can only view the actions that you are authorized to perform on the displayed accounts.
- **Status icons** – The status icons enable you to see the status of each account at a glance. The accounts lists display the following account statuses:

Icon	Status	Description
	Disabled	Automatic management for this account has been disabled by either the user or the CPM.
	Error	The CPM failed to perform an automatic management task.
	Locked	The account is locked. Move your mouse pointer over this icon to display the name of the user locking the account.
	Dual control	A request for permission to access this account must be created and confirmed before users can access it.
	Pending request	Your request for authorization to retrieve this account has not yet been confirmed.
	Confirmed request	Your request for authorization to retrieve this account has been confirmed.

- **Action icons** – The action icons enable you to perform actions on accounts in one simple click. The Accounts List displays only the icons that initiate activities that you are authorized to perform. The accounts lists display the following action items:

Icon	Title	Description
	Show password	Displays the password for a predefined number of seconds in a pop-up window. You can copy the displayed password directly from this window.
	Copy password	Copies the password for use without displaying it.
	Connect to	Enables you to activate a transparent connection to a remote machine. If the account can connect to remote machines using more than one connection type, a list of the configured connections is displayed. Alternatively, click the 'Connect to' icon to use the default connection.
	Action menu	Displays a list of additional actions that you can perform from the accounts list. <ul style="list-style-type: none"> ▪ Add to/remove from Favorites – This option is displayed if the account appears in the Favorites list. ▪ Add to/remove from Cart ▪ Display Account usages ▪ Display failed account usages

Displaying Accounts and Service Accounts

The accounts lists are displayed in a grid that you can organize according to your requirements and personal preferences, using the following features:

- **Multiple Paging** – After a search, all the accounts that meet the specified criteria are displayed in multiple pages. This is relevant for Account views, Operational views, and My Views.

The Accounts List facilitates full sorting, meaning that when you sort the displayed accounts according to column, all the accounts are organized in the new order across all the pages in the list.

- **Column displays** – You can reorganize and resize the columns in your accounts list, as well as sort the accounts according to some of the displayed columns. Any changes that you make in the column display are saved, and are applied next time you display an accounts list.
- **Hidden columns** – By default, the accounts list hides several columns that include information about the displayed accounts, which leaves more room in the list for other details. The user can display these columns manually, or the accounts list can be configured to display them automatically. For more information, refer to *Displaying Hidden Columns*, page 338.
- **Group by** - You can group accounts in the Accounts List in groups according to the displayed properties. This enables you to easily identify accounts that have the same properties. For more information, refer to *Grouping Recordings by Properties*, page 338.
- **Searching for accounts** – In addition to finding accounts in any of the available account views, you can search for them using either a regular search or an advanced search. The advanced search feature enables you to search in specific Safes, according to keywords. For more information, refer to *Searching for Accounts and Service Accounts*, page 228.
- **Searching for Account Usages** - As well as displaying accounts, you can also display service accounts and monitor their status. Although service accounts are not displayed as part of general search results, you can display an entire list of service accounts and see an overall picture of their status and master accounts. For more information, refer to *Searching for Accounts and Service Accounts*, page 228.
- **Displaying search results** – Each time you perform a search, the results are displayed in My Views where you can select them as often as you wish to display the search results without repeating the search. For more information, refer to *Managing Customized Views*, page 235.

Adding Accounts to your Cart

You can add accounts to your personal Cart in order to perform mass operations such as password change, verification, reconcile, resume, release, etc. across multiple account views.

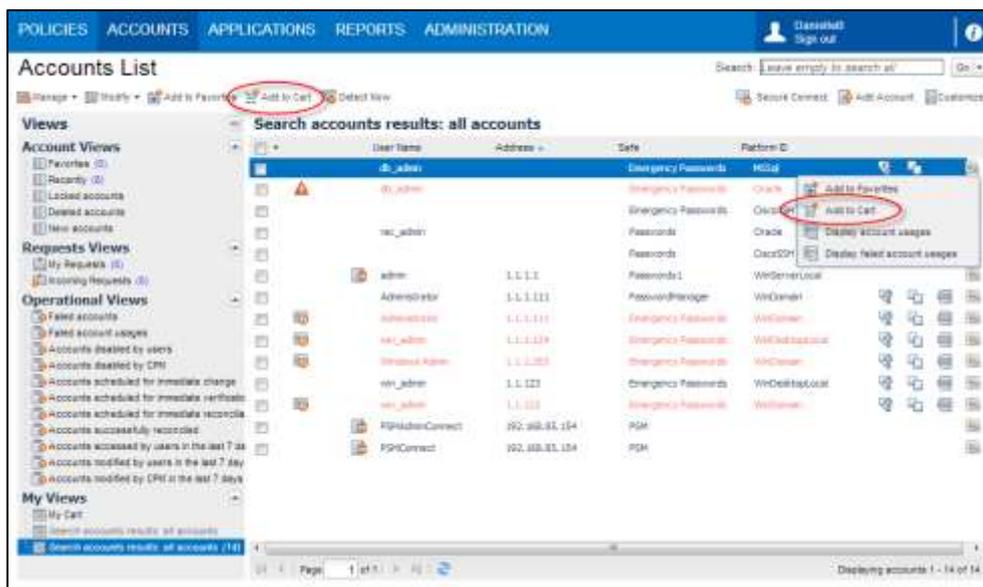
You can add accounts to your Cart with the following buttons:

- **Add to Cart** – Enables you to add selected accounts to your Cart according to the operation to perform, regardless of specific searches and the order in which the search results are displayed.
- **Add All to Cart** – Enables you to add all accounts in all pages of the search results to your Cart in one click.

The contents of each user's Cart is personal and is cleared when the user logs off from the PVWA.

To Add Selected Accounts to your Cart

1. In the Accounts List, display the account to add to your Cart.
2. Add the account to your cart:
 - Select the account, then on the toolbar, click **Add to Cart**.
 or
 - In the line of this account, click the **Action menu** icon, then from the pop-up action menu, select **Add to Cart**; the selected account is added to your Cart.

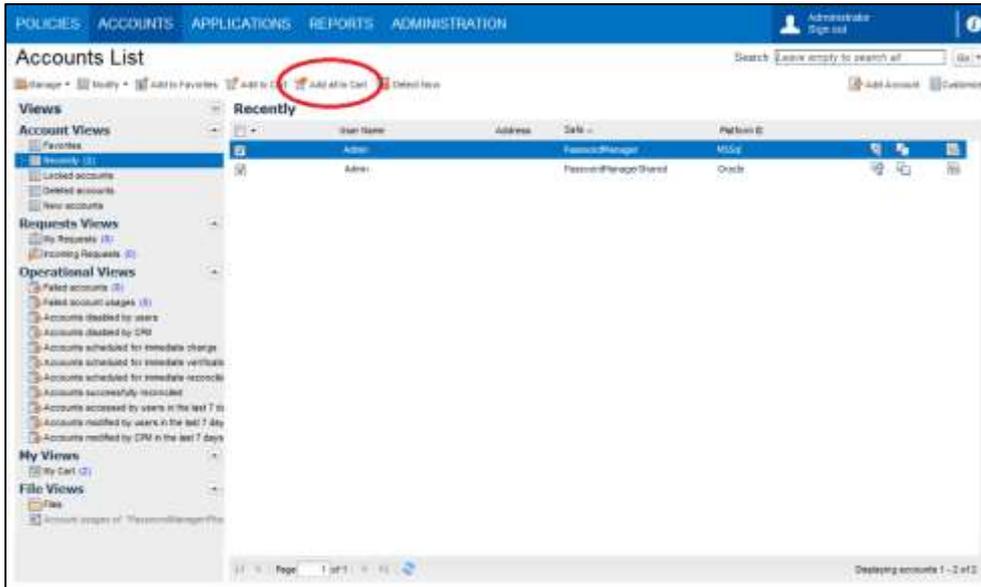


3. Click **My Cart** to display the contents of the Cart.

For more information, about selecting accounts to perform bulk operations, refer to *Selecting Accounts*, page 236.

To Add All Accounts to your Cart

1. In the Accounts page, on the toolbar, click **Add All to Cart**.



All accounts in all pages of the search results are added to your Cart.

2. Click **My Cart** to display the contents of the Cart.

Finding Accounts and Service Accounts

Users who have the Retrieve accounts and List accounts authorizations in the Safe where accounts are stored can view the passwords in accounts. Once they have found the account they are looking for, the authorization determines the tasks that they can perform, as follows:

- **Retrieve accounts** – Users can view the password.
- **Retrieve accounts and Use accounts** – Users can use the password to connect to a remote device.

For more information about Safe authorizations and tasks that can be performed, refer to *Adding and Managing Safe Members*, page 68.

Searching for Accounts and Service Accounts

Accounts that are retrieved or stored recently appear 'Recently' accounts lists. If the account you are looking for does not appear in this list, you will have to search for it.

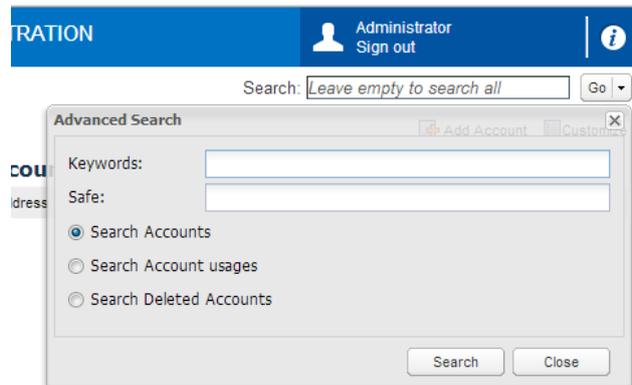
To Search for an Account

- In the Accounts page, specify the Search criteria:
 - To specify a regular search:
 - i. In the Search field, specify a keyword to search for. You can specify up to four keywords.
Specify focused search criteria to optimize the search, resulting in quick and accurate results. For example, IP address, user name, platform name, Safe name.

Note: You can specify a Safe name that includes spaces. This Safe name does not need to be specified within quotation marks.

You can carry out a search for all the accounts in the Vault that you have access to by leaving the Search field empty. However, this might take a while as the process searches the entire Vault.

- ii. Click **Go**; the search is carried out in all the Safes in the Vault that you are authorized to access.
- To specify an advanced search:
 - i. Click the drop-down arrow in the **Go** button; the advanced search pop-up window appears.



- ii. In the Keywords field, specify a keyword to search for. You can specify up to four keywords. If you leave this empty, a general search will be performed.
- iii. In the Safe field, specify the name of a Safe to search. If you don't specify a Safe, the search will be carried out in all the Safes in the Vault that you are authorized to access.
- iv. Select the type of account to search for.
- v. Click **Search**; the advanced search is carried out.

The PWA displays all the accounts that meet the specified criteria in the Accounts Results list. After a search that finds service accounts, the service accounts themselves are displayed in the search results, but not the master account.

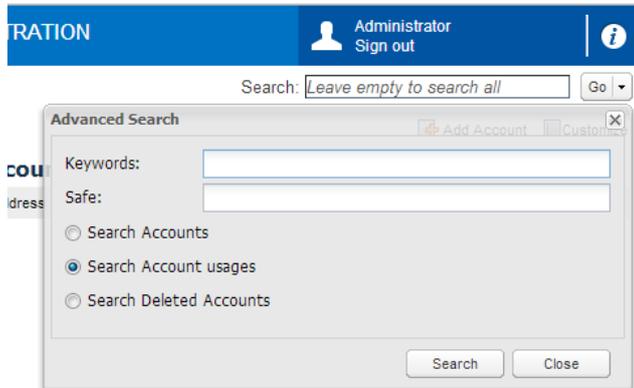
At the bottom of the list of accounts, you can see the number of accounts that met the search criteria, and the number of pages in the list.

- Click a column heading to reorganize the displayed accounts according to that column.
- Browse through the pages in the list to view additional accounts.

To Search for Service Accounts

You can search for service accounts in the Advanced Search window.

1. In the Accounts list, click the drop-down arrow in the **Go** button; the advanced search pop-up window appears.



2. In the Keywords field, specify a keyword to search for. You can specify up to four keywords. If you leave this empty, a general search will be performed.
3. In the Safe field, specify the name of a Safe to search. If you don't specify a Safe, the search will be carried out in all the Safes in the Vault that you are authorized to access.
4. Select **Search Account usages**, then click **Search**; an advanced search for service accounts is carried out, and a list of accounts that meet the specified criteria is displayed.



- Service accounts are displayed according to their master accounts. Click **Display master account details** to view more information about the master account



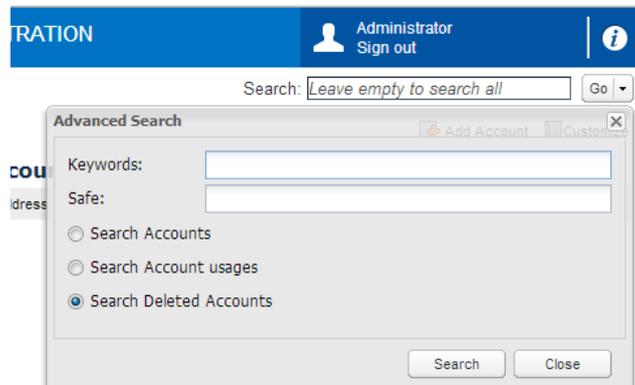
- Service accounts whose automatic management has been disabled are displayed in red and are marked with the disabled icon.

For information about managing failed service accounts, refer to *Managing Failed Service Accounts*, page 238.

To Search for Deleted Accounts

You can search for deleted accounts in the Advanced Search window.

1. In the Accounts list, click the drop-down arrow in the **Go** button; the advanced search pop-up window appears.



2. In the Keywords field, specify a keyword to search for. You can specify up to four keywords. If you leave this empty, a general search will be performed.
3. In the Safe field, specify the name of a Safe to search. If you don't specify a Safe, the search will be carried out in all the Safes in the Vault that you are authorized to access.
4. Select **Search Deleted Accounts**, then click **Search**; an advanced search for deleted accounts is carried out, and a list of accounts that meets the specified criteria is displayed.

Managing the Accounts List

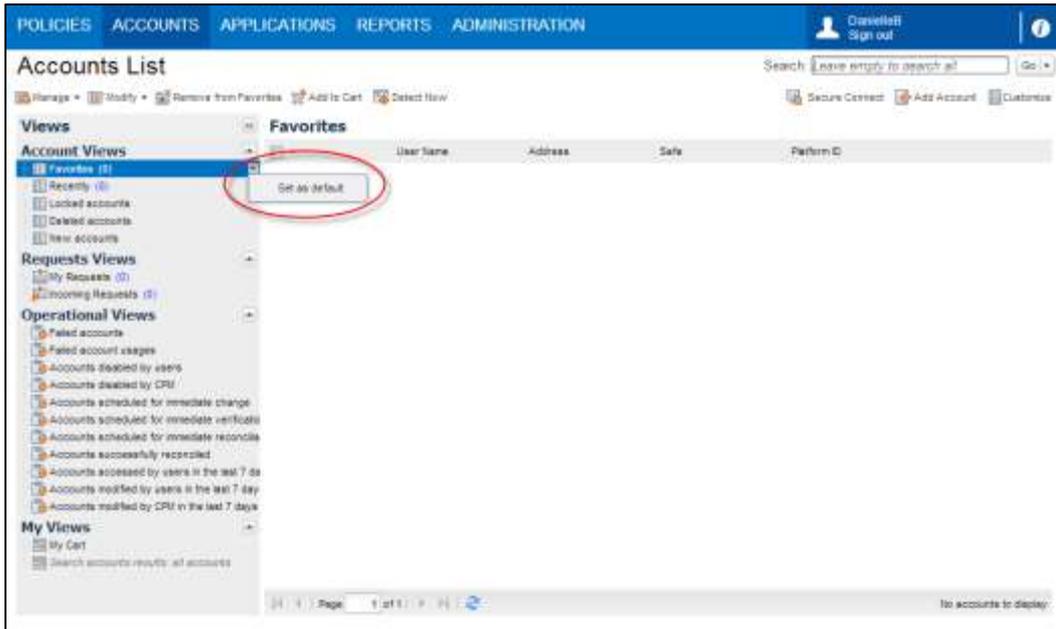
The Accounts List displays a set of predefined views. You can customize the Accounts List so that it displays your personal preferences, and you can add accounts to predefined lists for easy access.

Setting the Default View in the Accounts List

When you display the Accounts List, by default, the list of Recently accessed accounts is shown. However, you can change the default list and display the list that is most useful for you. You can either set a predefined view or one of the views in your Cart.

To Set the Default View

1. In the Accounts page, point to the view to display as default, then click the drop-down arrow in the selection.



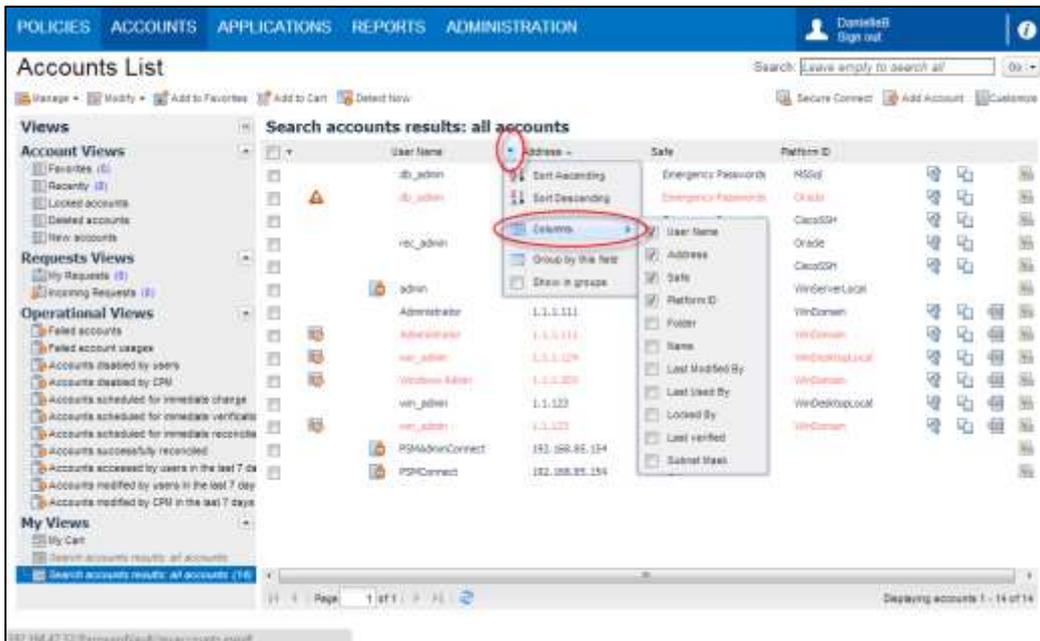
2. Select **Set as default**; the selected view will be displayed each time you display the Accounts List, until you select another view or change the default view.

Displaying Hidden Columns

Information about the accounts in the Accounts List is displayed in columns. However, by default, not all the available columns are displayed. You can customize your own Accounts List and display the columns that are more useful for your needs.

To Hide and Display Columns in the Accounts List

1. In the Accounts List, click the drop-down button in one of the column titles.



2. From the drop-down menu, select **Columns**, then select or clear the name of the column to display.

Grouping Accounts by Properties

You can organize the accounts in the Accounts List in groups according to the displayed properties. This enables you to easily identify accounts that have the same properties.

To Group Accounts according to Properties

1. In the Accounts List, click the drop-down button in the title of the column that will determine the property by which accounts will be sorted.

The screenshot shows the 'Accounts List' interface with a table of accounts. The 'Address' column header has a dropdown menu open, and the 'Group by the field' option is selected and highlighted with a red circle. The table contains the following data:

User Name	Address	Safe	Platform
ib_admin		Emergency Passwords	PGCD
ib_admin		Emergency Passwords	GoSafe
rec_admin		Emergency Passwords	CloudSH
		Passwords	Oride
admin		Passwords	CloudSH
		Passwords	WinServerLocal
Administrator	1.1.1.111	PasswordManager	WinDomain
Administrator	1.1.1.111	Emergency Passwords	WinDomain
win_admin	1.1.1.124	Emergency Passwords	WinWebAccess
Windows Admin	1.1.1.201	Emergency Passwords	WinExtran
win_admin	1.1.123	Emergency Passwords	WinDesktopLocal
win_admin	1.1.123	Emergency Passwords	WinDomain
POPAdminConnect	192.168.85.134	POP	
POPConnect	192.168.85.134	POP	

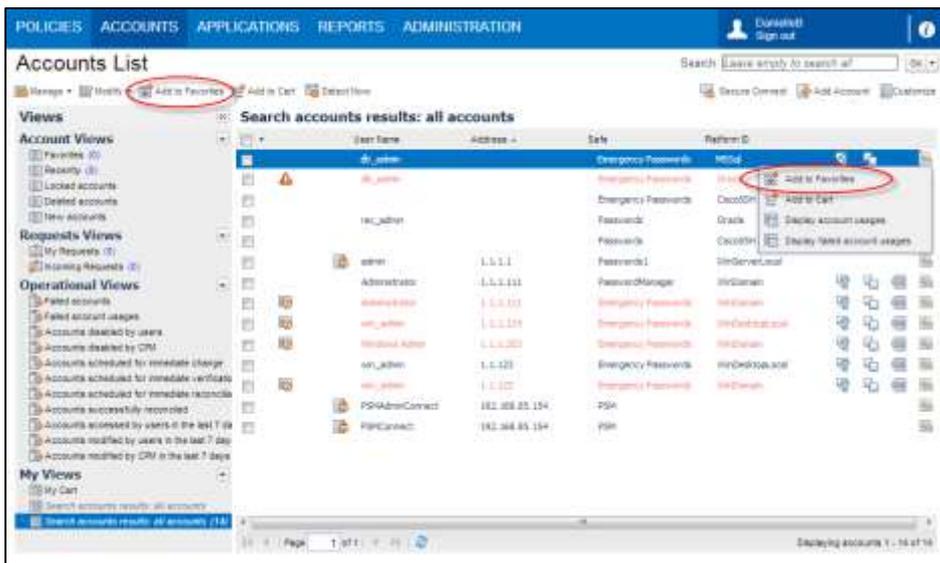
2. From the drop-down menu, select **Group by this field**; the PWA reorganizes the displayed accounts according to the selected property (column title).

Adding Accounts to the Favorites List

You can add accounts to your personal Favorites list. This list is for your account only and is displayed each time you display the Account List.

To Add Accounts to the Favorites List

1. In the Accounts List, display the account to add to the Favorites list.
2. Add the account to the Favorites list:
 - Select the account, then on the toolbar, click **Add to Favorites**.
 or
 - In the line of this account, click the **Action menu** icon, then from the pop-up action menu, select **Add to Favorites**.



The selected account is added to your Favorites list.

3. Click **Favorites** to display the contents of this list.

Customizing your own Views

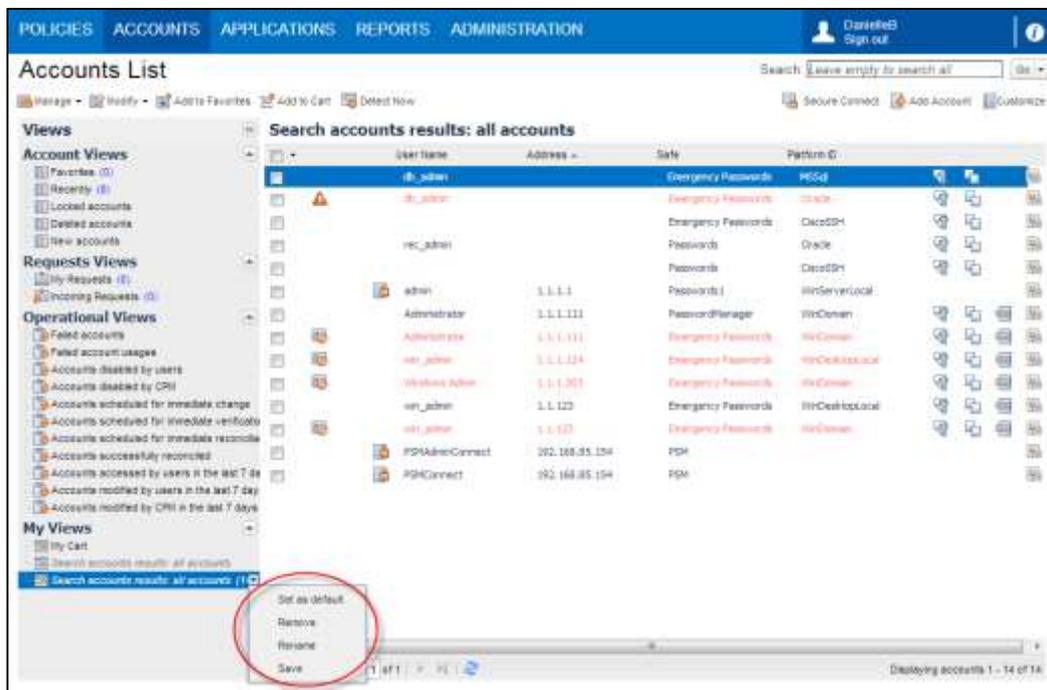
The My Views area in the Account Views displays your own account views. These views enable you to create customized account lists and save them for reference.

Managing Customized Views

Each time you perform a search, the results are listed in My Views where you can display them as often as you wish without repeating the search. Search results are listed temporarily while the user who performed the search is still logged on, after which they are removed from your customized views list. You can save search results so that they are listed again the next time you log on, and even set them as the default view when you display the Accounts page. You can also rename them or remove them from the list.

To Manage Customized Views

1. In **My Views**, point to the view to display as default, then click the drop-down arrow in the selection.



A pop-up menu enables you to do the following activities:

- **Set as default** – Sets the selected view as the default view that will be displayed when the Accounts List is displayed.
- **Remove** – Deletes the selected view from the list of customized views.
- **Rename** – Enables you to specify a name for the selected view.
- **Save** – Saves the selected view using the default name.

2. Select the relevant option.

Selecting Accounts

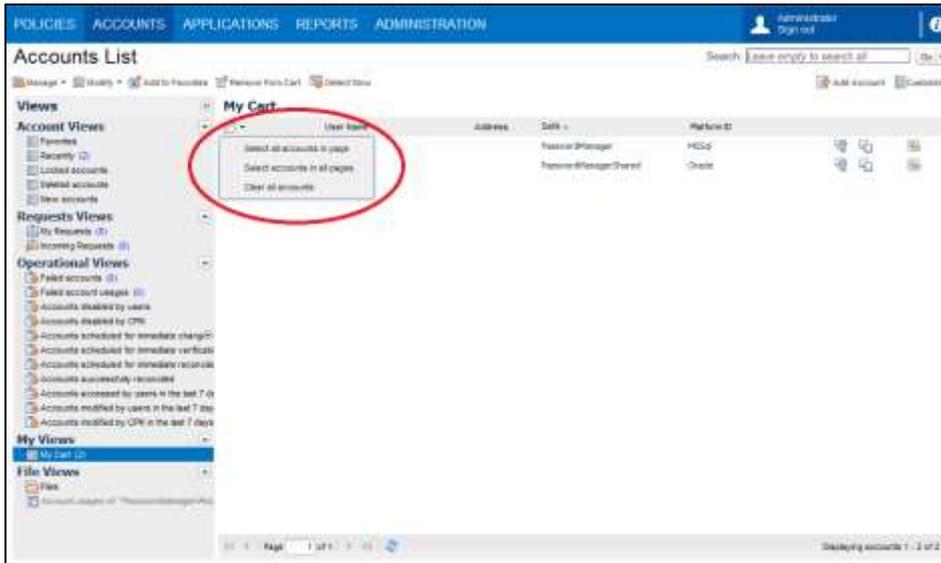
By selecting multiple accounts, you can initiate mass operations. You can select individual accounts or multiple accounts across pages in the Accounts List.

To Select Individual Accounts

- Identify the specific accounts to select, and then select them one at a time.

To Select Multiple Accounts

- In the column title for account selection, click the drop-down arrow; a drop-down menu appears.



- Select the relevant option, as follows:
 - Select all accounts in page** – The PVWA will select all the accounts displayed on the current page of the Accounts List.
 - Select accounts in all pages** –The PVWA will select all the accounts displayed in all the pages of the current Accounts List. This option is only available in the lists displayed by My Cart. In addition, you cannot edit accounts that are selected with this option or add them to the Favorites list.

To Clear Selected Accounts

- In the column title for account selection, click the drop-down arrow; a drop-down menu appears.
- Select **Clear all accounts**; the PVWA clears all the selections in the current Accounts List.

Managing Accounts and Service Accounts

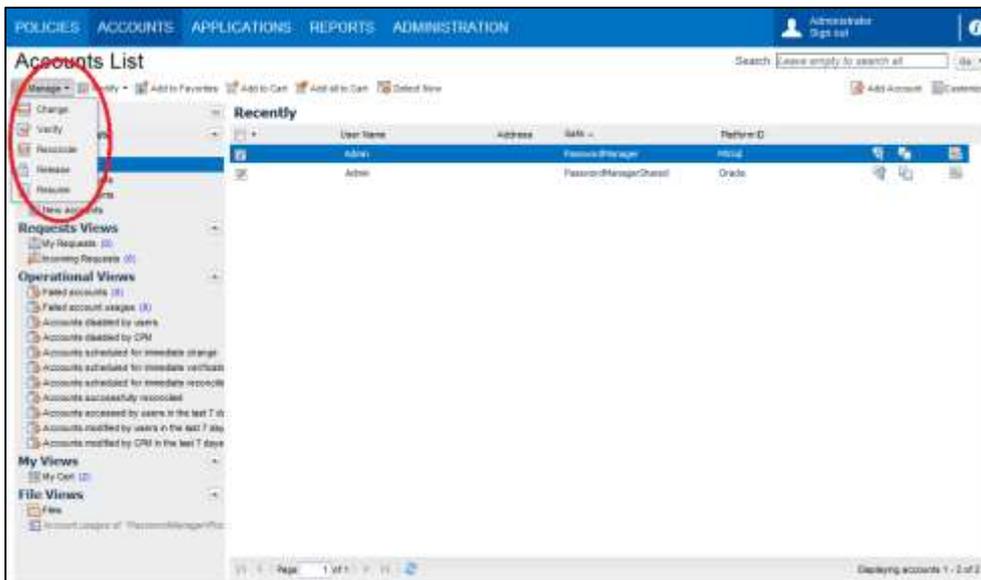
Managing Accounts

You can manage selected accounts in the Accounts List and perform the following activities using the Manage drop-down list on the toolbar:

- **Change passwords** – You can change passwords manually or initiate an automatic change to a password that is generated randomly by the CPM. For more information, refer to *Changing Passwords*, page 197.
- **Verify passwords** – You can initiate manual password verification processes to ensure that passwords on remote devices are synchronized with corresponding passwords in the Password Vault. For more information, refer to *Verifying Passwords*, page 205.
- **Reconcile passwords** – You can initiate automatic reconciliation processes to synchronize passwords on remote machines with corresponding passwords in the Vault. For more information, refer to *Reconciling Passwords*, page 207.
- **Release accounts** – After retrieving an exclusive account, you can release it through the Password Vault Web Access. For more information, refer to *Releasing Exclusive Accounts*, page 247.
- **Resume automatic management** – You can resume automatic management for accounts that were disabled manually or automatically by the CPM. For more information, refer to *Resuming Automatic Management*, page 214.

To Manage Accounts

1. Select the accounts to manage. For more information about selecting individual and multiple accounts, refer to *Selecting Accounts*, page 236.
2. On the toolbar, click **Manage**; the Accounts Management drop-down menu appears.



3. Select the management activity to perform on the selected accounts. If you select an activity that requires more information, the relevant windows are displayed. For example, if you select **Change**, the Change Password window is displayed.

Managing Failed Service Accounts

To Display Failed Service Accounts

The Account Operational Views enables you to display a list of failed service accounts in one click, without performing a search.

- In the list of Operational Views, click **Failed account usages**; a list of failed service accounts in all the Safes that you are authorized to access is displayed immediately.

To Resume Automatic Management for Failed Service Accounts

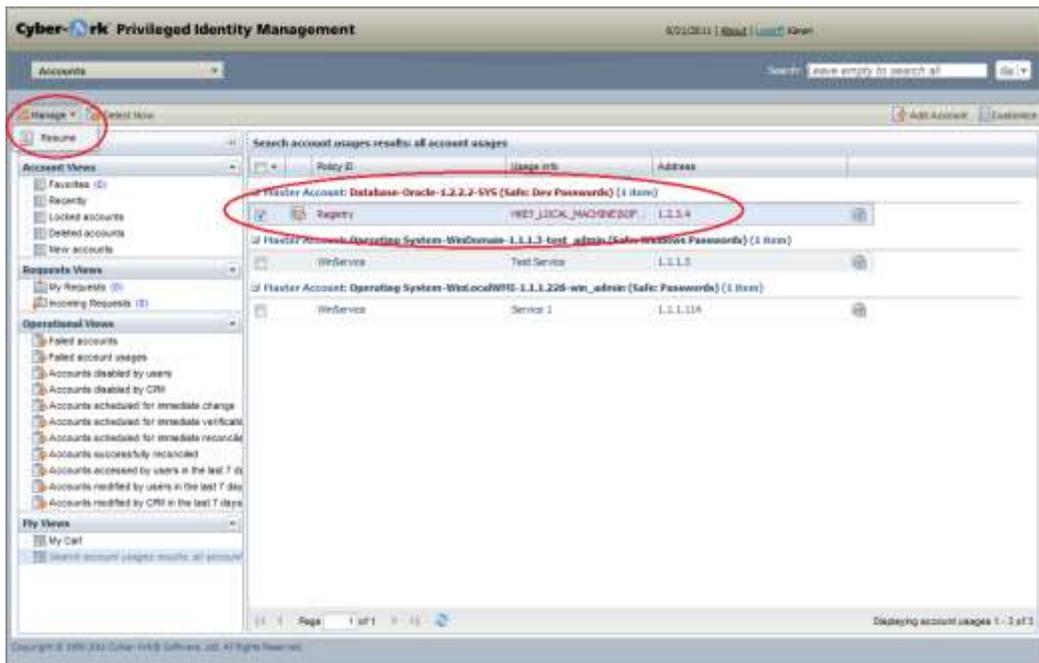
You can manually resume automatic management for failed service accounts that are listed in the **Failed account usages** list.



1. In the Accounts List, display the list of failed service accounts. You can do this in either of the following ways:
 - Using an advanced search, search for service accounts using keywords and Safe names, if possible. All disabled service accounts are displayed in red and are marked with the disabled icon.

or

 - In the Operational Views, display **Failed account usages**.
2. Select the disabled service account(s) that you will resume automatic management for.



3. On the toolbar, from the drop-down Manage menu, select **Resume**; the PWWA resumes automatic management for the selected service account(s) and displays them in the same way as all the other automatically managed service accounts.

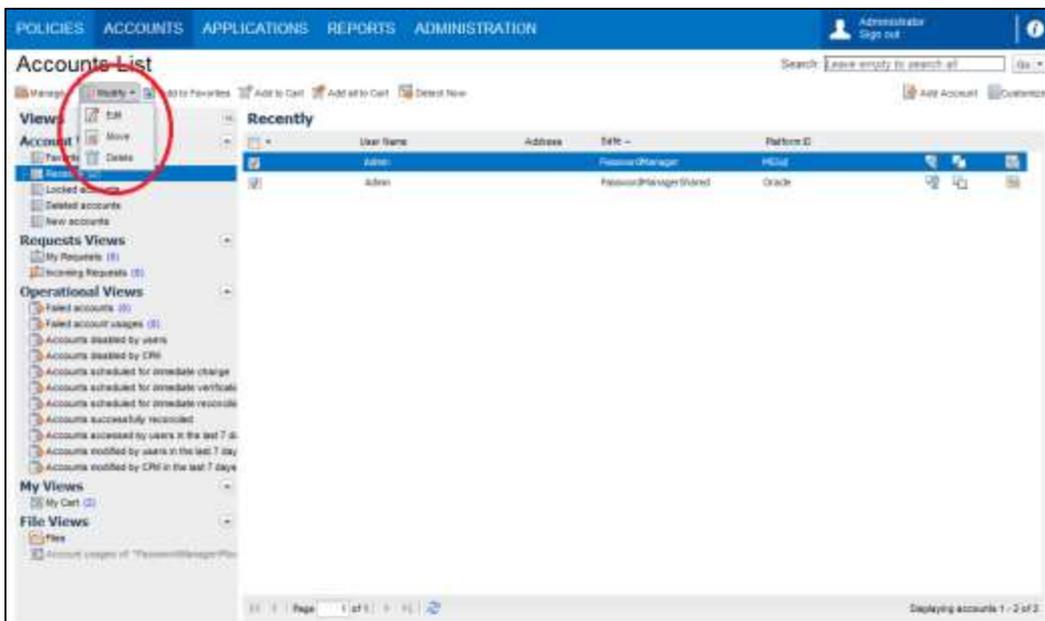
Modifying Accounts

You can modify selected accounts in the Accounts List and perform the following activities using the Modify drop-down list on the toolbar:

- **Edit accounts** – You can edit properties of existing accounts in the PVWA. For more information, refer to *Editing Account Properties*, page 210.
- **Move accounts** – You can move accounts between Safes and reorganize accounts. For more information, refer to *Moving Accounts between Safes*, page 163.
- **Delete accounts** – You can delete selected accounts. Make sure you will not need these accounts again.

To Manage Accounts

1. Select the accounts to modify. For more information about selecting individual and multiple accounts, refer to *Selecting Accounts*, page 236.
2. On the toolbar, click **Modify**; the Accounts Modify drop-down menu appears.



3. Select the modify activity to perform on the selected accounts. If you select an activity that requires more information, the relevant windows are displayed. For example, if you select **Move**, the Move Accounts window is displayed. For more details about these activities, refer to the relevant information in this guide.

Viewing Passwords

When you identify the account that contains the password you require, you can view the password, if you have the appropriate permissions. The password is displayed for a predetermined number of seconds, and then it is replaced by asterisks.



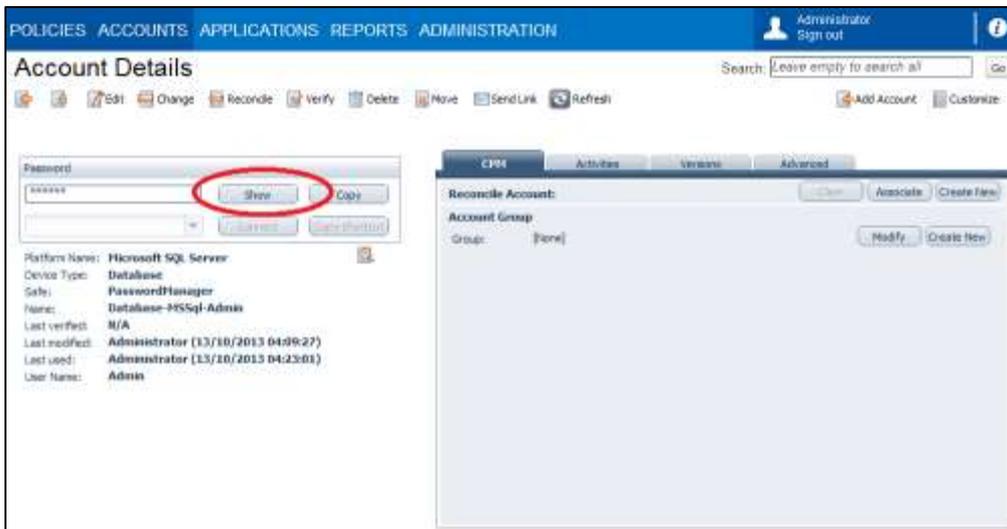
- In the Accounts list, click the **Show password** icon in the line of the account to view; the password in the account line is displayed for a predetermined number of seconds.



If this password is configured for one-time use, exclusive use, or use during a predefined timeframe, the relevant information is displayed in this window.

Or,

- In the Accounts list, click the account to view; the Account Details page appears. In the Password pane, the password appears as a series of asterisks.
- Click **Show**; the asterisks are replaced by the password for a predetermined number of seconds.

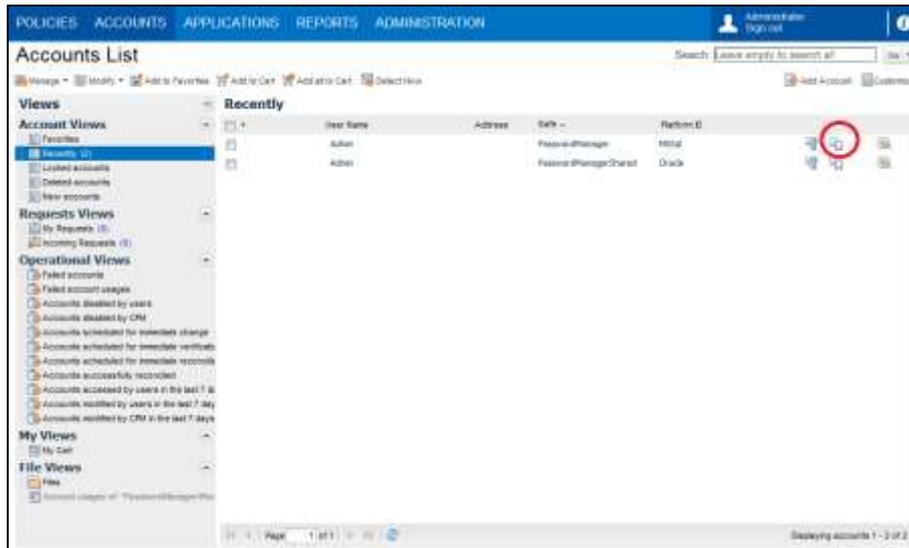


Copying Passwords

Authorized users can copy passwords with or without displaying them in the following pages:

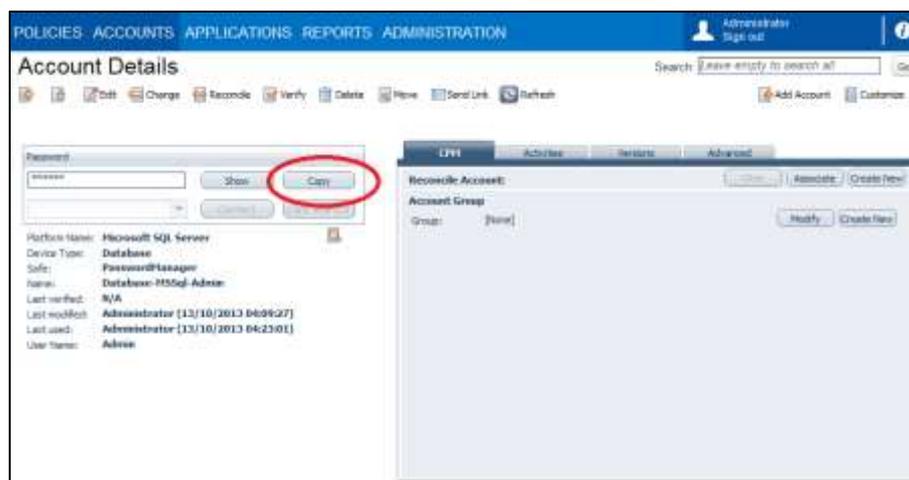
Note: On Chrome, the first time you copy a password, you are prompted to install a Chrome extension. For more information, refer to *Copying Passwords in Chrome*, page 242.

- Accounts List
 - In the Accounts List, in the record of the account whose password you wish to copy, click the **Copy password** icon.



A message appears confirming that the password has been copied to the clipboard.

- Account Details page
 - In the Account Details window, click **Copy**.



A message appears confirming that the password has been copied to the clipboard.

In addition, users who are authorized to view passwords can also copy them in the following window:

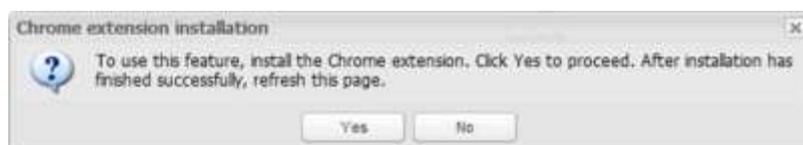
- Show Password window
 - In the Show Password window, click **Copy**.



A message appears confirming that the password has been copied to the clipboard.

Copying Passwords in Chrome

On Chrome, the first time you copy a password, the following message appears:



- Click **Yes** to download and install this extension,
- or,
- Click **No** to deny installation and close this message box.

After the Chrome extension is installed, press **F5** to refresh the PWA page. The next time you copy a password, the extension will work automatically.

Accessing Accounts

A new Account Retrieval form combines all aspects of the account retrieval workflow and enables users to specify information that is required for them to access account in one step. This window combines the following information:

- **A reason for accessing an account** – If the Master Policy requires users to specify the reason for accessing the account, a reason edit box is displayed in this window. Users can either specify a reason in their own words, or can select a reason from a list of predefined reasons. For more information, refer to *Specifying a Reason for Accessing Passwords*, page 244.
- **Ticketing information** – If the platform associated with this account is integrating with a ticketing system, a section for the ticketing system is displayed. Users can specify the relevant ticketing system and ID. For more information, refer to *Integration with Ticketing Systems*, page 244.
- **Dual control requests** – If the Safe where the account is stored requires users to create requests before they can access accounts, request information is displayed in this window. Users can create a request that must be confirmed by authorized users, including a timeframe and whether the request is for single or multiple access. Exclusive and one-time passwords can be changed after the timeframe specified in the request has expired. For more information, refer to *Dual Control*, page 251.
- **Connection details for Privileged SSO** – If the platform associated with this account specifies connection details for a transparent connection to a remote device, the connection details are displayed in this window. This section can be customized to prompt users for additional information before the PVWA logs them on transparently to the remote device. For more information, refer to *Privileged Single Sign-On*, page 267.

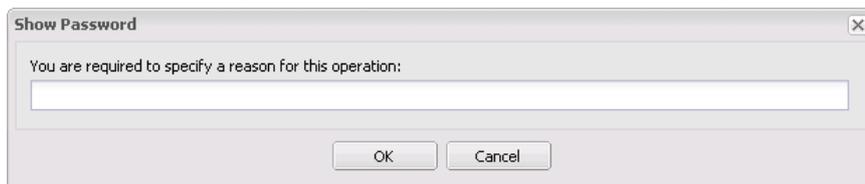
When a user tries to access a password that requires any of the above information, the Account Retrieval page displays all the relevant sections that enable the user to provide the required access information, according to PVWA configuration.

Specifying a Reason for Accessing Passwords

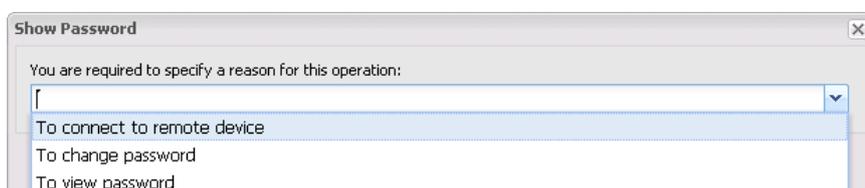
A rule in the Master Policy determines that users can only retrieve passwords after they specify a reason that explains why they want to retrieve them.

To Specify a Reason for Accessing a Password

- In the Account Details page, click **Show**, **Copy**, or **Connect** to access the account; the Show Password window appears and displays the reason edit box.
 - If users can specify a reason in their own words, specify the reason now.



- If users are required to select a reason from a predefined list, specify the reason now.



- Click **OK**; the Privileged Account Security solution will now retrieve the password, and the reason you specified or selected will be stored in the audit log.

Integration with Ticketing Systems

The Privileged Account Security solution integrates with enterprise ticketing systems to ensure that users are authorized to access passwords, and to create an audit of password activity in the Vault.

The Privileged Account Security solution supports ticketing systems in the following ways:

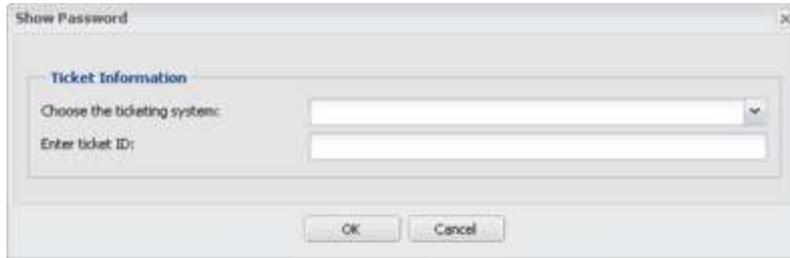
- After a ticket has been opened in the enterprise ticketing system, users are required to specify the name of a ticketing system and the number of a specific ticket that will give them access to the password. After the user specifies the ticketing information, a validation process is launched which, if successful, will permit the user to retrieve the password. If the ticket is not validated, the user will not be permitted to retrieve the password.
- A ticket can be created in the ticketing system when a password is retrieved.

A CyberArk module, which is a type of 'user-exit', can be developed and customized to implement this functionality according to the customer's precise needs. For more information, refer to *Developing an External Module for Ticketing Integration*, page 519.

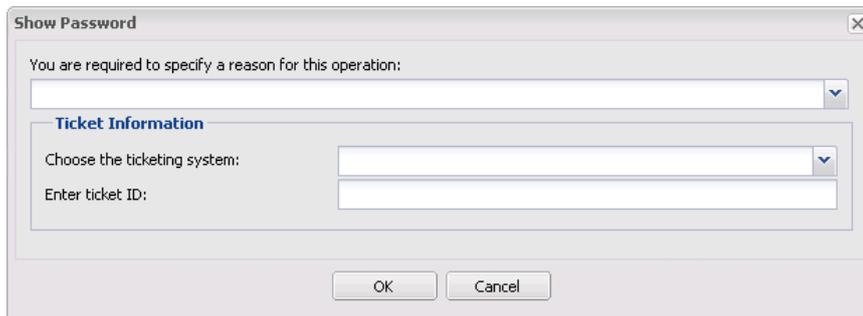
For more information about configuring the PWA to integrate with Ticketing Systems, refer to *Integrating with Ticketing Systems*, page 516.

To Retrieve a Password

1. In the Account Details page, click **Show**, **Copy**, or **Connect** to access the account; the Show Password window appears.



If the ticketing system integration requires a reason, the reason edit box will also be displayed.



2. If a reason is required, specify a reason for retrieving the password or select a reason from the predefined list.
3. Select the ticketing system.
4. Specify the ticket ID, then click **OK**;

Note: The ticket ID must not contain any of the following characters: ()=,.

The integrated ticketing system launches a validation process to authorize the password retrieval, or

To create a ticket, leave the ticket number empty, then click **OK**.

If the validation process fails, the PWWA will display a message indicating that password retrieval has been denied.

Accounts Check-out and Check-in

Auditing and control requirements demand full identification and monitoring of users who access privileged accounts during any given period. In addition, to guarantee accountability, each user who accesses a privileged account must be the only one to do so.

The Master Policy enables organizations to permit users to check out a 'one-time' password and lock it so that no other users can retrieve it at the same time. After the user has used the password, he checks the password back into the Vault. This ensures exclusive usage of the privileged account, enabling full control and tracking for the password.

If the organizational policy determines that a password can only be used once, the Master Policy can also be configured to change the password's value before unlocking it and making it available to other users. If a CPM is installed, this can be done automatically.

Viewing Checked-out Accounts

If an account is checked out, and therefore locked, a 'Locked' icon appears in the Account list on the line of the locked account.



To View Accounts Checked-out by your User

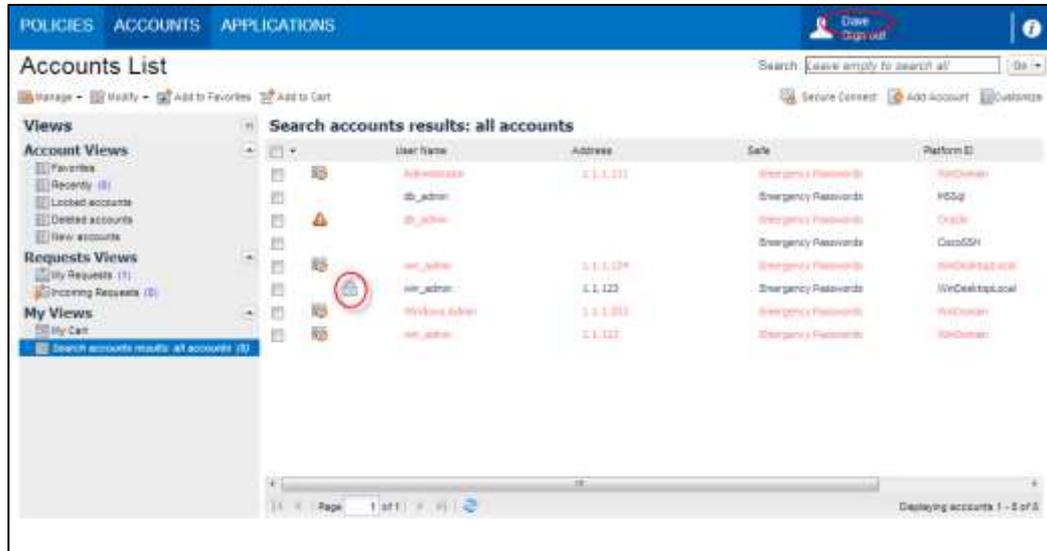
- In the Accounts page, in Views, click **Locked accounts**; the PWA displays the Locked accounts view, which lists the accounts that are locked by your user.

User Name	Address	Role	Pattern ID
var_admin	1.1.1.123	Emergency Passwords	WinDesktopLocal

To View Accounts Checked-out by other Users

You can check for accounts that have been checked-out by other users in the Safes where you are an owner.

- In the Accounts list, display any list of accounts; all the locked accounts are marked with the **Locked account** icon.



Users who have the 'View Safe Members' authorization can see the name of the user who has locked the account when they place the mouse over the locked icon.

Releasing Exclusive Accounts

After retrieving an exclusive account, you can release it through the Password Vault Web Access. If you do not release the account manually, one of the following processes happens, depending on the way the account is managed:

- Account is managed automatically by the CPM** – The CPM will release it automatically after the period of time specified in the platform.
- Account is managed manually, not by the CPM** – The account must be released manually. A notification is sent to a user who is authorized to release the password and change it.

Authorized users can release accounts in either of the following pages:

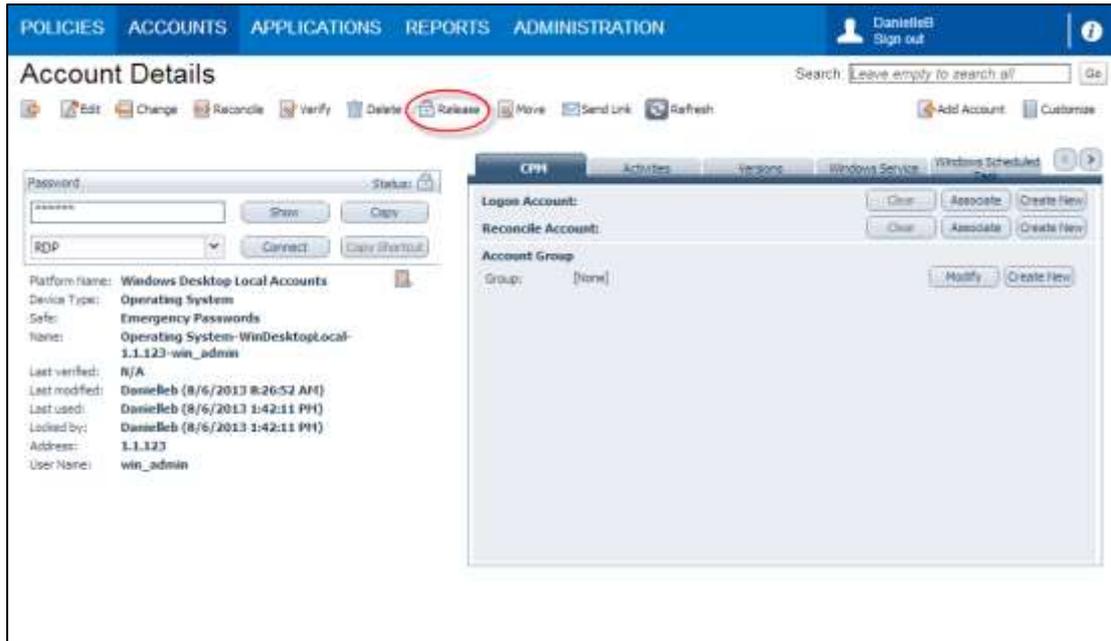
- Locked Accounts
- Edit Account

In addition, administrators can release locked accounts in the following page:

- Account Details

To Release an Exclusive Account in the Account Details Page

- Display the Account Details page of the account to release, then click **Release** to return the account to the Safe.
 - If the account is managed automatically by the CPM, it is released and the password is changed immediately.
 - If the account is managed manually, a notification is sent to a user who is authorized to change the password. The account is released automatically after it has been changed.

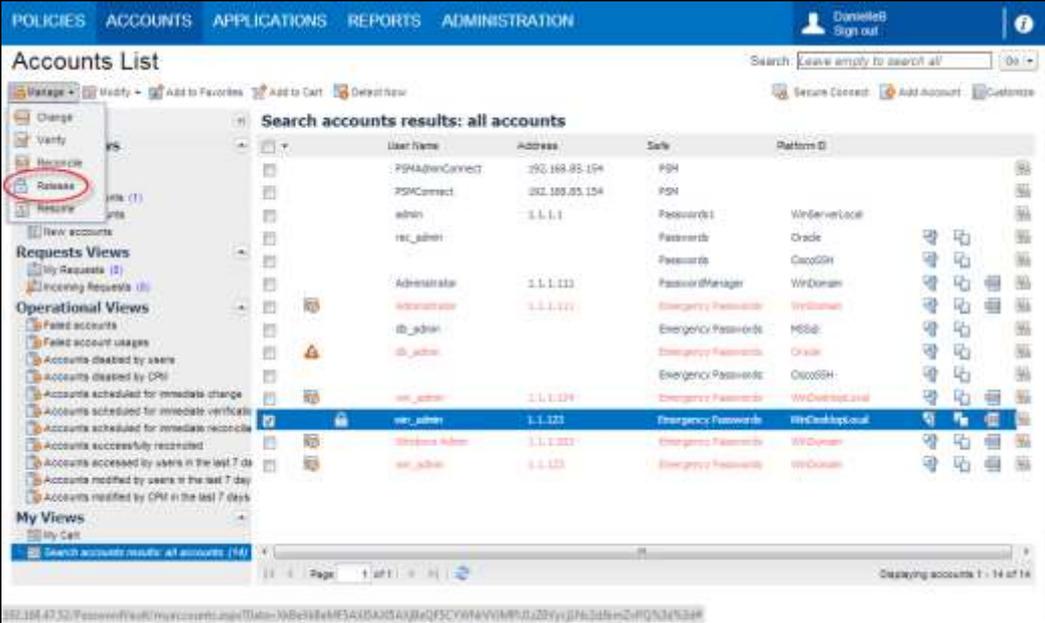


If a user requires an account urgently when it is locked by another user, a user with the 'Unlock Accounts' authorization can unlock it **in the Edit Account page** so that it can be used.

Note: Only give Safe members the 'Unlock accounts' authorization if essential. This action could result in more than one user retrieving the same password, with no accountability over who performed operations using this account during this period of time.

To Release an Exclusive Account in the Locked Accounts List

1. In the Locked Accounts list, select the account to release.
2. From the **Manage** drop-down menu, select **Release**.
 - If the account is managed automatically by the CPM, it is released and the password is changed, so that it can be used by other users.
 - If the account is managed manually, a notification is sent to a user who is authorized to change the password. The account is released automatically after it has been changed.



The screenshot displays the 'Accounts List' interface. The top navigation bar includes 'POLICIES', 'ACCOUNTS', 'APPLICATIONS', 'REPORTS', and 'ADMINISTRATION'. The user is logged in as 'Dimitris' and can sign out. A search bar is present with the text 'Search: Leave empty to search all'. Below the navigation, there are options for 'Storage', 'Verify', 'Add to Favorites', 'Add to Cart', and 'Delete Now'. A 'Manage' dropdown menu is open, showing options: 'Change', 'Verify', 'Release' (highlighted with a red circle), and 'Reserve'. The main content area shows 'Search accounts results: all accounts' with a table of accounts. The table has columns for 'User Name', 'Address', 'Safe', and 'Platform ID'. The account 'ver_admin' is selected and highlighted in blue. The bottom of the page shows a pagination bar with 'Page: 1 of 1' and 'Displaying accounts 1 - 14 of 14'.

User Name	Address	Safe	Platform ID
PSHAdminConnect	250.168.85.154	PSH	
PSHConnect	252.168.85.154	PSH	
admin	1.1.1.1	Password	WinServerLocal
ver_admin		Password	Crack
		Password	CrackSSH
Administrative	1.1.1.111	PasswordMerge	WinDomain
administrator	1.1.1.111	Emergency Password	WinDomain
ID_admin		Emergency Password	MSB
ID_admin		Emergency Password	Crack
		Emergency Password	CrackSSH
ver_admin	1.1.1.134	Emergency Password	WinServerLocal
ver_admin	1.1.1.133	Emergency Password	WinClientLocal
Windows Admin	1.1.1.202	Emergency Password	WinDomain
ver_admin	1.1.1.123	Emergency Password	WinDomain

To Release an Exclusive Account in the Edit Account Page

This procedure is only for administrators who have the **Unlock accounts** permission, to enable users to access an account urgently when it is locked by another user.

Note: Only give Safe members the 'Unlock accounts' authorization if it is essential. This action could result in more than one user retrieving the same password, with no accountability over who performed operations using this account during this period of time.

1. In the Accounts list, select the account to release, then click **Edit**; the Edit Account window appears.
2. Click **Show advanced section**; the advanced options appear.

The screenshot shows the 'Edit Account' interface for a Windows Desktop Local Account. The account name is 'Windows Desktop Local Accounts-win_admin-1.1.123'. The 'Locked by' field is populated with 'DanielleB (8/6/2013 1:42:11 PM)' and a red circle highlights a 'Release' button next to it. The 'Save' and 'Cancel' buttons are disabled. The 'Show advanced section' button is also visible.

These details indicate that the account is locked, the name of the user, and the date and time when the account was locked.

The locked account cannot be changed until it has been released, so while it is locked, the Save buttons are disabled. As soon as the account is released, the Save button is enabled, and the password and account properties can be changed.

3. Click **Release**.
 - If the account is managed automatically by the CPM, it is released and the password is changed, so that it can be used by other users.
- Note:** This bypasses the standard release workflow and should only be used in emergencies.
- If the account is managed manually, a notification is sent to a user who is authorized to change the password. The account is released automatically after it has been changed.
4. If the account is a member of an account group, click **Release Group**; the account is unlocked. To release other accounts in the account group, release them in the same way.

This release will **not** trigger a password change.

Dual Control

The Master Policy enables organizations to ensure that passwords can only be retrieved after permission or 'confirmation' has been granted from an authorized Safe Owner(s).

Authorized Safe Owners can either grant or deny requests. This feature adds an additional measure of protection, in that it enables you to see who wants to access the information in the Safe when, and for what purpose.

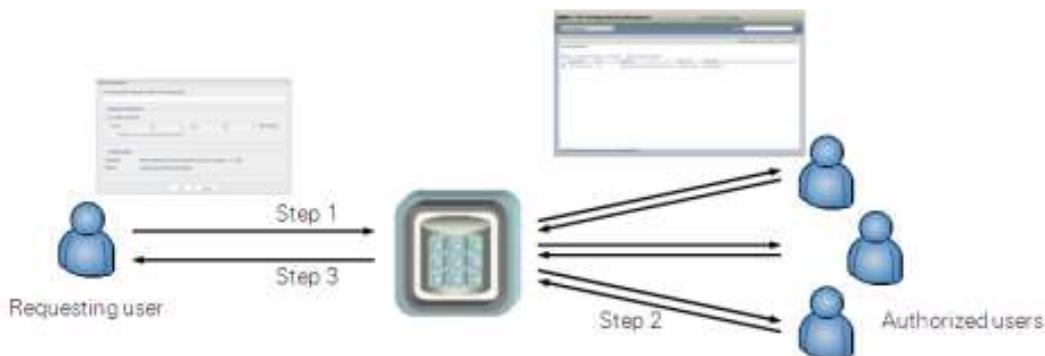
Note: The **first** group member who confirms or rejects a request does so on behalf of the entire group. If more than one confirmation is required, each group is equivalent to a single authorized user and will count as a single confirmation/rejection.

As soon as users receive confirmation for a request from an authorized user, they can access the password or file that the request was created for.

The manual security workflow comprises the following three steps:

1. **Creating a request** – A user who wishes to access an account in an environment where the Master Policy enforces Dual Control must first create a request. In the request, the user specifies the reason for accessing the account, whether they will access it once or multiple times, and the time period during which they will access it. A notification about the request is sent to users who are authorized to confirm this request.
2. **Responding to the request** – Through the notification, authorized users can access the request and view its details. Based on these details, authorized users can either confirm or deny the request. The number of authorized users who are required to confirm requests is defined in the Master Policy.
3. **Activating confirmation** – Each time an authorized user responds to the request, the user who created it receives a notification. When the total number of required confirmations is received for the request, this user receives a final notification. The user can now activate the confirmation and access the account according to the request specifications.

The following diagram shows the above steps:



Users can access requests as long as they are valid. As soon as a request becomes invalid, it cannot be accessed by either the user who created it or by users who are authorized to confirm it.

Requests become invalid because of one of the following reasons:

- The access period that the user specified in the request has passed.
- The user created a request for single access, which has already been used.
- The Safe's request retention period for the request has passed.
- The Safe, file, or password specified in the request has been deleted.
- There are not enough supervisors to authorize this request, the number of supervisors has changed, or the settings for confirmation have been changed.
- The Vault version has been updated.

Dual Control Options

The Privileged Account Security solution offers you several options for dual control:

- **Basic Policy Rule:**

- **Require dual control password access approval** – A request must be confirmed by one or more authorized users before privileged accounts can be accessed. A specific number of authorized users required to confirm requests can be determined in Advanced Settings by the **Number of confirmers required to authorize requests** setting. Dual control mode is enabled when the advanced multi-level and managerial approval modes are inactive.

For more information about configuring dual control, refer to *Dual Control*, page 513.

- **Advanced Settings:**

- **Require multi-level password access approval** – A request must be confirmed by two levels of authorized users before privileged accounts can be accessed. Authorized Safe owners (either groups or users) are assigned a confirmation level, and authorize requests according to that order. This means that the first level of authorized users must confirm requests before they are transferred to the second level of authorized users. Permission to access the requested privileged account is only given after both levels of authorized users have confirmed the request. If a request is denied at the first level, it is not passed on to the second level, and if it is denied at the second level, the confirmations from the first level become irrelevant.

When a number of required confirmers is set by the **Number of confirmers required to authorize requests** advanced setting, this number of confirmers is required **at each level**. If **All** confirmers are required to confirm requests, all confirmers from both levels must confirm requests before accounts can be accessed. For example, if the **Number of confirmers required to authorize requests** setting is set to three confirmers, a total of six confirmers are required to review and approve requests – three confirmers from level one and three confirmers from level two.

For more information about configuring dual control, refer to *Defining Confirmers*, page 262.

- **Only direct manager can approve password access requests** – A request must be confirmed by the direct managers of the user who created the request. This streamlines the confirmation process as, typically, privileged accounts are stored in Safes where multiple authorized users can confirm requests. This workflow integrates with Active Directory to automatically identify the requestor's direct manager.

This advanced setting cannot be enabled together with multi-level confirmation, or with multiple required confirmers (more than one), as requests will never be confirmed and will not be usable.

For more information about configuring the Vault for direct manager confirmation, refer to *Configuring Confirmation*, page 514.

Accessing Privileged Accounts

This section describes how to create requests for access to privileged accounts, track them, and utilize them to access accounts after you have received confirmation from authorized users.

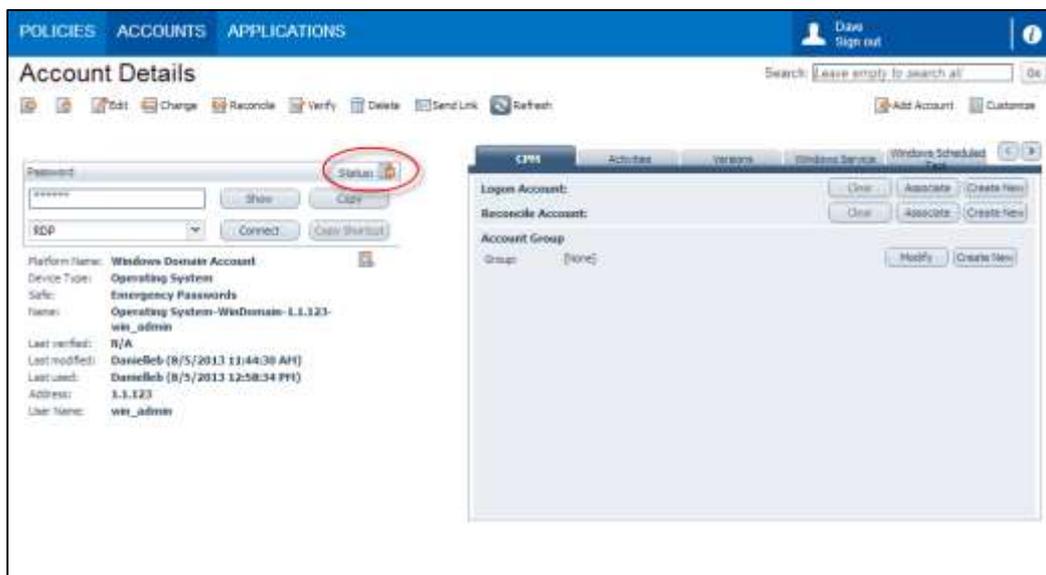
Requesting Access to Accounts

Before a user can retrieve an account in an environment where the Master Policy enforces access confirmation, a request must be sent to authorized users to be confirmed. You can create multiple requests in a single action to streamline the access workflow. If access to multiple accounts and confirmation is required, you can select the required accounts and submit requests for all of them in a single click. For each account, a separate request will be sent for confirmation. Once access to an account is confirmed you can use this account and don't need to wait for confirmation for the other accounts.

To Request Confirmation

1. Create a single or multiple requests:

In either the Accounts List or the Accounts Details page of the password to retrieve, a status icon indicates that a request must be created and authorized before users can retrieve this account.



2. Create a request:

- To create a request for access to a single account:
 - In the Accounts List or the Account Details page, click **Show**, **Copy** or **Connect**; a request is created and the Request Access window appears.

- To create requests for access to multiple accounts:
 - In the Accounts List, select the required accounts and click **Request Access**; the Request Access window appears.

This window prompts the user for all the access information that they are required to provide before they can access the account and view or use the password.

3. In the **Reason** area, type the reason for the request.
4. If a ticket is required to access this account, in the **Ticket Information** area, select the ticketing system and specify the ticket ID.
5. If you require access during a period of time, in the **Request Timeframe** area, select **Access is required from** and specify the dates.
6. If you will need to access the Safe or file/account several times, select **Multiple access is required during this period**.
7. If this request requires multi-level confirmation, in the Confirmation area, the request **Status** indicates the number of authorized users who must confirm the request at each level. Click the information to view a list of the authorized users.

Show Password
 You are required to specify a reason for this operation:
 Request Timeframe
 Access is required:
 From: 29/06/2014 08:00 To: 01/07/2014 17:00 GMT+03:00
 Multiple access is required during this period
 Confirmation
 Operation: Retrieve password Windows Desktop Local Accounts-service1-10.10.0.6
 Status: [1 user\(s\) from the 1st level and 1 user\(s\) from the 2nd level must confirm the request](#)
 OK Cancel

If this request requires Direct Manager confirmation, the Status details indicates the number of authorized users who are required to confirm the request. Click **Status details** to display the name of the group that is required to confirm the request and a list of group members.

- If this request is for confirmation to log onto a remote machine transparently, and you can use either a domain or NIS account, you can select the machine to connect to and enforce. This means that when you connect using a confirmed request, you are automatically logged in to this machine.

Show Password
 You are required to specify a reason for this operation:
 Ticket Information
 Choose the ticketing system: BMC Remedy
 Enter ticket ID:
 Request Timeframe
 Access is required:
 From: 9/20/2012 8:00 AM To: 9/22/2012 5:00 PM GMT+03:00
 Multiple access is required during this period
 Confirmation
 Operation: Retrieve password WinDomain-WinDomAdmin-1.1.1.203
 Status: [1 user\(s\) must confirm the request](#)
 Remote Connection Details
 Remote Machine:
 OK Cancel

- If this request is for confirmation to enable you to connect to a remote database through the PSM, and the system is configured to enable specific users to connect as a different user, the Connect As drop-down list is displayed.

10. From the drop-down list, select the user to use to log onto the remote database.
11. To view more details about the users who will confirm this request, click the linked status; a list of authorized users for this request is displayed. You can view more information about specific users by expanding their user name.
12. Click **OK**; the request is created and sent to users who can authorize it.

or,

Click **Cancel** to close the password retrieval form without sending the request.

After you have created the request, if the ENE is configured to send notifications for new requests a notification is sent to all the authorized users who are required to confirm the request.

- If **Require multi-level password access approval** was enabled, a notification is sent to the first level of authorized users who are required to confirm it. After the required number of authorized users have confirmed the request, a notification is sent to the second level of authorized users who are required to confirm it.
- If **Only direct manager can approve password access requests** was enabled, a notification is sent to your direct managers who are required to confirm it.

For more information about configuring the ENE to send notifications, refer to *Configuring the ENE*, page 805.

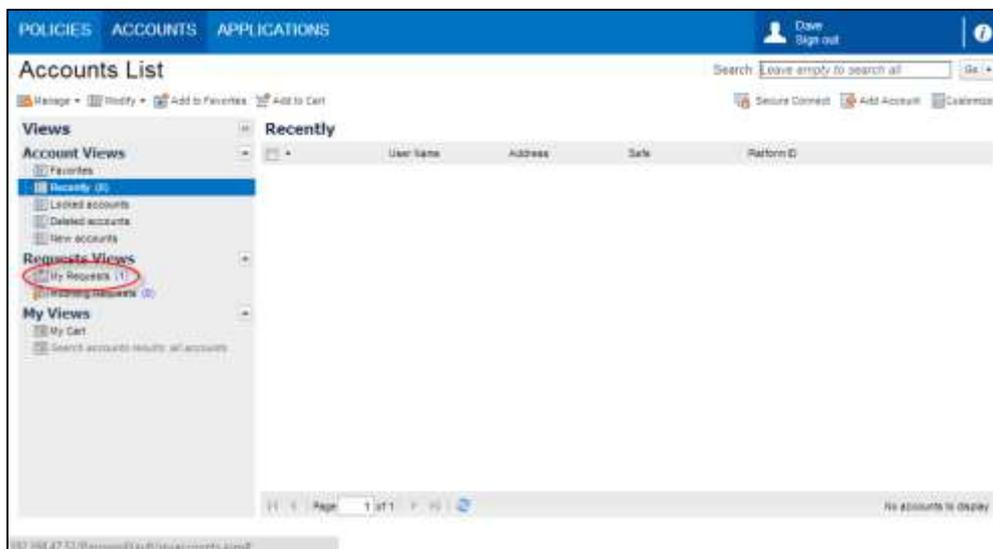
If a user tries to access the same account or file again before receiving confirmation, the Request Details page appears. A second request is not sent as the previous request is still unanswered.

Viewing your Requests

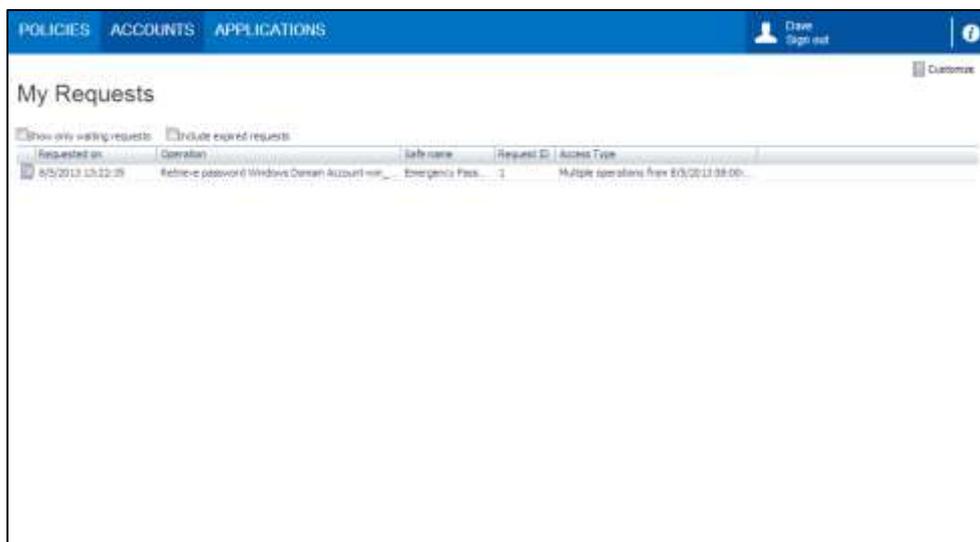
After you have sent a request, you can view its status at any time. You can also delete requests that are no longer relevant or invalid.

To View your Requests

1. In the Accounts List, the Requests View enables you to view the requests you have sent for authorization.



2. Click **My Requests**; the My Requests page appears.



Note: The Request ID is unique to each Safe.

This page lists the requests that you have created and sent for authorization. The icon next to each request indicates the status of the request:

Icon	Indicates ...
	The request has not yet been authorized.
	The request has been authorized.
	The request has become invalid.

3. Select **Show only waiting requests** to display your requests that have not yet been confirmed.
4. Select **Include expired requests** to display invalid requests in the requests list.
5. Click a request to display more information in the Request Details page.

Deleting a Request

The user who created a request can also delete it.

To Delete a Request

1. In the Request Details page, click **Delete on the toolbar**; you are prompted to confirm that you want to delete the request.
2. Click **Yes** to delete the request,
or,
Click **No** to leave the request in the Requests list and return to the Request Details page.

If the ENE is configured to send notifications when requests are deleted, a notification is sent to all the authorized users who are required to confirm the request.

- If **Require multi-level password access approval** was enabled, a notification is sent to all authorized users at the level that is currently required to confirm this request.
- If **Only direct manager can approve password access requests** was enabled, a notification is sent to your direct managers .

For more information about configuring the ENE to send notifications, refer to *Configuring the ENE*, page 805.

Accessing Accounts

As soon as your request has been handled by an authorized user, you can see it in the Accounts List.

To Use Confirmation

1. In the Accounts List, My Requests counter displays the total number of approved, declined and waiting requests. The tooltip displayed when you place your mouse over 'My Requests' displays the number of each type of request.

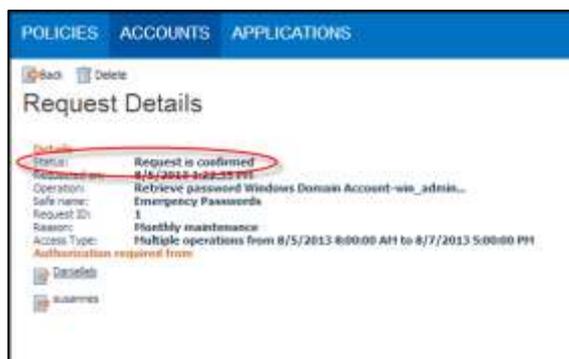


- Click the link to the request objects; the Access Requests page appears and displays the My Requests list.



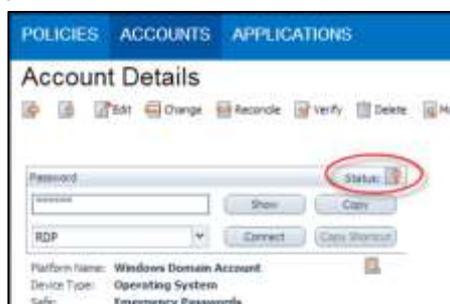
Confirmed requests are marked with the confirmed request icon so that you can identify them at a glance.

- Select **Show only waiting requests** to display requests that have not yet been authorized.
- Select **Include expired requests** to display invalid requests.
- Select the confirmed request; the Request Details page appears.



This page displays the status of the request.

- Click the name of a user who is authorized to confirm the request to display more information.
- Click the name of the account that appears in the Account Details; the Account Details page for that account appears. The status icon now indicates that the request to retrieve this account was confirmed and you can now use the password.



If the confirmed request is for a single operation, after you have used it to access an account or file, the request becomes invalid.

Confirming Requests

This section describes how to confirm requests for access to privileged accounts that you have received. It is specifically for users who are authorized to confirm requests.

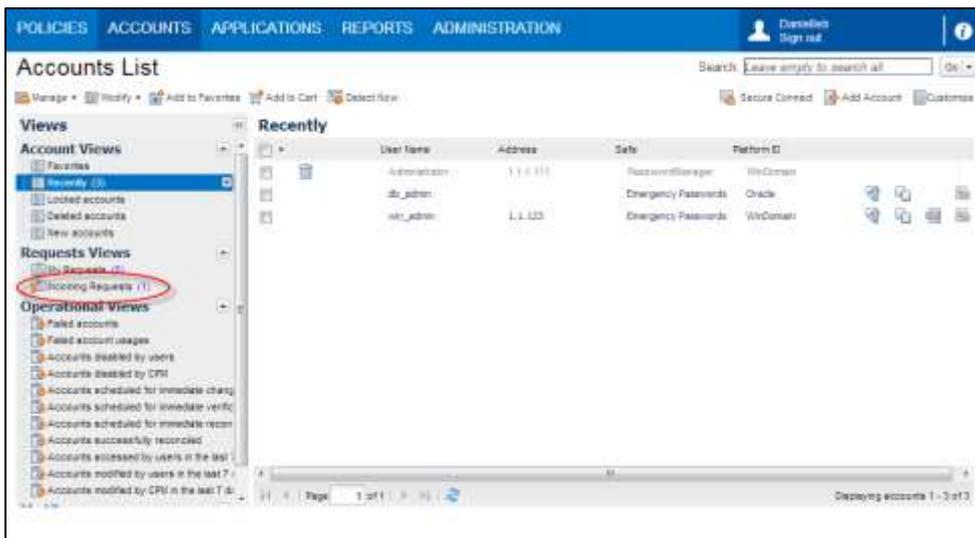
Safe Owners who have the Authorize password requests permission for a specific Safe can authorize requests to permit other users to access an account in that Safe. The instructions below are for these Safe members.

When a request must be authorized by multiple users, these users can do so in any order. However, if a request requires multi-level confirmation, the first level of authorized users receive the request for confirmation immediately after it is created. The second level of authorized users only receive the request after the required number of users at the first level have confirmed it. If any users at the first level deny the request, it is not sent to users at the second level.

Note: The **first** group member who confirms or rejects a request does so on behalf of the entire group. If more than one confirmation is required, each group is equivalent to a single authorized user and will count as a single confirmation/rejection. This is relevant to both basic and multi-level confirmation.

To Confirm a Request

1. In the Accounts List, you can see how many requests are waiting for you to authorize.



2. Click **Incoming Requests**; the Incoming Requests page appears.

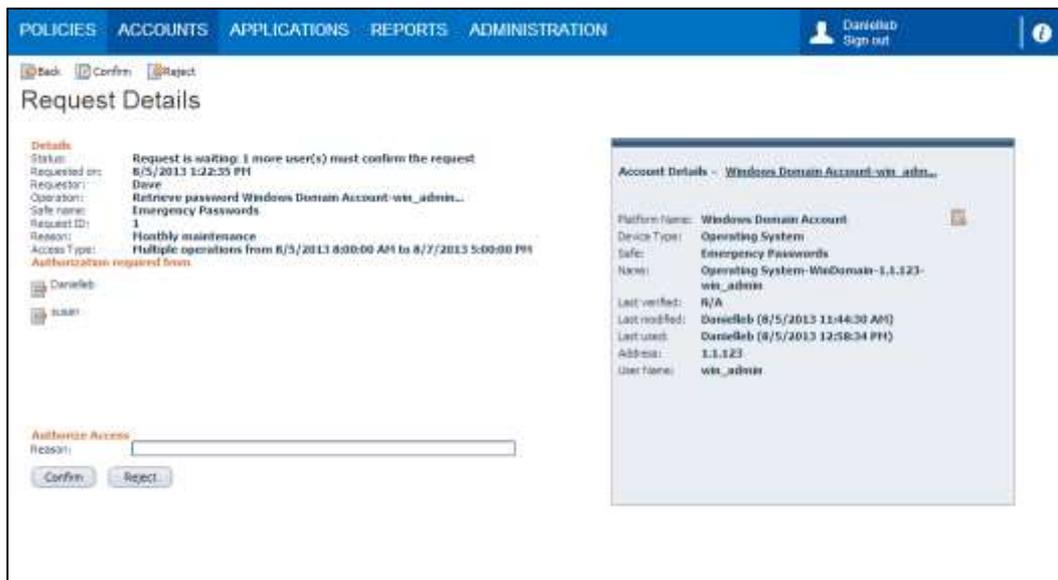


Note: The Request ID is unique to each Safe.

- By default, this page displays the requests that are waiting for you to authorize or reject. Clear **Show only requests waiting for my confirmation** to display all the requests that you have authorized or rejected.

Note: This option may be hidden, so that you can only view requests that are waiting for you to authorize or reject.

- Select **Include expired requests** to display invalid requests.
- Click a request to display more information; the Request Details page for the authorized user appears.



This page displays the details of the request as well as the buttons that enable the user to confirm or reject the request.

- If this request requires multi-level confirmation, the following additional information is displayed:
 - In the Request Status, the number of users from each level that still have to confirm the request is displayed.
 - In the list of users who are required to authorize this request, the names of the authorized users are displayed.
 - If this request requires confirmation by a direct manager, the following additional information is displayed:
 - In the Request Status, a single required confirmation is specified, as only the direct managers group can confirm this request.
 - In the list of users who are required to authorize this request, the name of the LDAP group who can confirm the request is displayed.
- After reading the request, specify the reason for authorizing or rejecting the request.
 - Click **Confirm** to confirm the request,
or,
Click **Reject** to reject the request and prevent the user who created the request from accessing the account or file.

The Incoming Requests page appears again. If **Show only requests waiting for my confirmation** is selected, the request that was handled does not appear in the list.

After you have confirmed or denied the request, a notification is sent to all the authorized users who are required to confirm the request.

- If the advanced **Require multi-level password access approval** setting was enabled:
 - After each confirmation or denial, a notification is sent to all authorized users at the level to which the user who that has just confirmed it belongs.
 - After the first level of authorized users have confirmed a request, a notification about the request is sent to the second level of authorized users.
 - After the final confirmation, a notification is sent to both levels of authorized users.

Defining Confirmers

This sections describes how to define users who are authorized to confirm requests for access to privileged accounts. It is specifically for Vault administrators who manage Vault users and define the CyberArk workflows.

Setting the Confirmation Requirement

When creating a Safe or changing Safe properties, you can specify that Users who wish to retrieve passwords and files require confirmation from an authorized Safe Owner.

To Set the Confirmation Requirement

1. Click **POLICIES** to display the Master Policy.
2. In **Privileged Access Workflows**, select **Require dual control password access approval**.
3. In the Rule Preview pane, click **Edit Settings**; the Edit Rule Settings window appears.

The screenshot shows the 'Edit Rule Settings' window for the 'Require dual control password access approval' policy. The window is titled 'Edit Rule Settings' and has a close button (X) in the top right corner. Below the title bar, there is a breadcrumb trail: 'Privileged Access Workflows | Require dual control password access approval'. The main content area is divided into two sections: 'Basic Policy Rule' and 'Advanced Settings'. In the 'Basic Policy Rule' section, there is a toggle for 'Require dual control password access approval' which is currently set to 'Inactive'. In the 'Advanced Settings' section, there are three settings: 'Require multi-level password access approval' (Inactive), 'Only direct managers can approve password access requests' (Active), and 'Number of confirmers required to authorize requests' (set to 1). At the bottom of the window, there are three buttons: 'Save', 'Save & Close', and 'Cancel'.

-
4. Set the policy settings as described below:
 - **Require dual control password access approval** – This setting determines whether this master policy rule is applied, using the following values:
 - **Active** – This rule will be applied at Master Policy level to all platforms, unless an Exception overrides it.
 - **Inactive** – The rule will not be applied at all.
 - **Require multi-level password access approval** – This advanced setting enforces an access control workflow in which end users require two levels of authorization before they can access privileged accounts. Each list of Safe members must include users who are authorized to confirm requests at both levels. The number of required confirmers set in the advanced **Number of confirmers required to authorize requests** setting applies to each level of authorized users. For more information about setting different levels of authorized users, refer to *Creating an Authorized User*, page 264.
 - **Only direct managers can approve password access requests** – This advanced setting enforces an access control workflow in which end users require authorization from their direct managers before they can access privileged accounts. This workflow requires the Digital Vault to recognize and integrate with your Active Directory. For more information, refer to the Privileged Access Security Installation Guide.

Note: This mode cannot be enabled together with multi-level confirmation, or with multiple required confirmers (more than one), as requests will never be confirmed and will not be usable.

For more information about configuring the Vault for direct manager confirmation, refer to *Configuring Confirmation*, page 514.
 - **Number of confirmers required to authorize requests** – The number of authorized users who are required to confirm requests
 - If **Require multi-level password access approval** was enabled, select a number to determine the number of authorized users **at each level** who are required to confirm requests.
 - If **Only direct manager can approve password access requests** was enabled, you cannot specify a multiple number of direct managers for this mode.
 5. Click **Save** to save the new rule settings and remain in the Edit Rule Settings window,
or,
Click **Save & Close** to save the new rule settings and display the Master Policy page. You can see the Advanced Settings in the Rule Preview pane.
-

The screenshot displays the CyberArk console interface for configuring a Master Policy. The top navigation bar includes 'POLICIES', 'ACCOUNTS', 'APPLICATIONS', 'REPORTS', and 'ADMINISTRATION'. The user is logged in as 'administrator' and can 'Sign out'. The main content area is titled 'Policies > Master Policy' and shows a table of 'Privileged Access Workflows'.

Policy Rule	Value	Exceptions
Require dual control password access	Active	-
Enforce check-in/check-out exclusions	Inactive	-
Enforce one-time password access	Inactive	-
Allow EPV transparent connections	Active	-
Require users to specify reason for access	Active	-

Below the table, there are expandable sections for 'Password Management', 'Session Management', and 'Audit'. On the right side, the 'Rule Preview' shows 'Require dual control password access' is active. The 'VALUE' section has a blue 'Active' button. The 'ADVANCED SETTINGS' section includes 'Require multi-level password access approval' (inactive), 'Require managerial password access approval' (inactive), and 'Number of confirmers required to authorize requests' set to 1. The 'EXCEPTIONS' section is currently empty.

At the bottom right, there are buttons for 'Edit Settings' and 'Add Exception'. The CyberArk logo is visible in the bottom left corner.

The system is now configured for dual control, and users who wish to retrieve accounts are required to request access confirmation from at least one authorized user. By default, requests are retained in the Safe for 30 days.

Users can specify default settings for requests in the Dual Control settings in the Web Access Options. For more information, refer to *Configuring the System through PVWA*, page 918.

Creating an Authorized User

Authorized Users can confirm requests by other users who require access to passwords. When you add a user as an Owner of the Safe, you can give them the 'Authorize password requests' authorization, or you can update an existing Safe member's properties.

Note: Any changes in the confirmers settings (eg, removing confirmers, changing confirmer levels, etc.) makes all existing requests obsolete. All existing active requests must be deleted and re-created.

To Create an Authorized User

If the user is not yet an owner of the Safe where the accounts are stored:

1. In the Add Safe Members page, select the Safe member/group to configure as an authorized user.
2. Select the authorizations that the Safe member/group will have in the Safe. Specifically, select **Authorize password requests**; the Confirmation Level options appear.

Add Safe Member

Search: Search In:

Selected Search: Vault Display 5 result(s)

Name	Business Email	Full Name
ManagerialGroup		
PasswordManager		
RequestorManagerial	RequestorManag...	RequestorManag...
ManagerialGroupVault		
ManagerManagerial	ManagerManage...	ManagerManage...

Account Management
 Safe Management
 Monitor
 View Audit log
 View Safe Members
 Workflow
 Authorize password requests
 Level 1
 Level 2
 Access safe without confirmation

The options enable you to define two different levels of authorized users to confirm requests. This is relevant to the **Require multi-level password access approval** setting, which is an advanced setting of the Master Policy's **Require dual control password access approval** rule.

3. If **Require multi-level password access approval** was enabled, select the level to which the user/group will belong.
 - Level 1 – The first level of users who are authorized to confirm requests.
 - Level 2 – The second level of users who are authorized to confirm requests.

These users can only confirm requests at the specified levels. In addition, the first level of authorized users must confirm requests before any authorized users from the second level can confirm them.

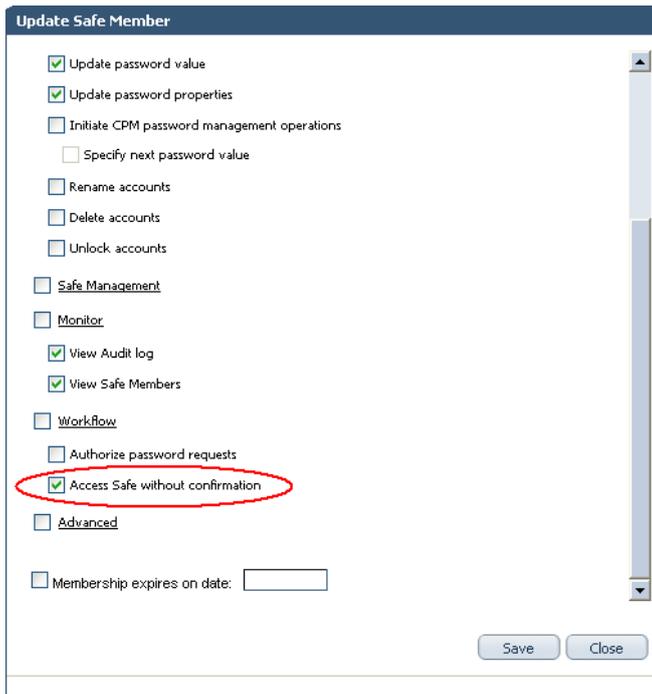
Note: Users who belong to multiple groups, and one group is defined as the first level of confirmers and the other is defined as the second level of confirmers will be considered as the second level of confirmers.

4. Click **Add**; the Safe member is added to the Safe with the authorization to confirm requests from other users to access passwords in this Safe.

Enabling a Safe Owner to Access a Safe without Confirmation

Although confirmation is specified in the Safe properties window and therefore applies to all Safe Owners, it is possible to give certain users authorization to access the Safe without requiring confirmation from other Safe Owners.

1. In the Safe Details page, display the Safe Members tab and select the Safe member who you will permit to access accounts without confirmation, the **Update Safe Members** page appears.
2. Select **Access Safe without Confirmation**, then click **Save**; this Safe Owner is now able to access the Safe or file without confirmation.



The screenshot shows the 'Update Safe Member' dialog box with the following options:

- Update password value
- Update password properties
- Initiate CPM password management operations
 - Specify next password value
- Rename accounts
- Delete accounts
- Unlock accounts
- Safe Management
- Monitor
 - View Audit log
 - View Safe Members
- Workflow
 - Authorize password requests
 - Access Safe without confirmation
- Advanced
- Membership expires on date:

Buttons: Save, Close

A Safe Owner who has an open request for confirmation can be given authorization to access the Safe without confirmation. However, he will not be able to access the Safe in this way until all open requests have been confirmed.

Privileged Single Sign-On

Privileged Single Sign-On (SSO) enables users to connect to remote devices without needing to know or specify the required password or key. This increases usability as the user does not have to open a login session and then copy and paste credentials into it and also prevents the password or key from being exposed to the user.

Users are required to authenticate to the PWWA before they can use passwords or keys stored in the Password Vault to access sensitive remote devices. This enforces strong authentication for accessing managed devices and restricts user access to passwords and keys according to access control.

Users can connect transparently to a remote machine in the following pages:

- Account lists
- Account versions
- Account details

Users can also directly access the Connect window used to log onto a remote device through a direct URL or a desktop shortcut.

Transparent connections can be defined for implementations that use the CPM as well as implementations that do not use the CPM. In addition to making it easier for users to log onto remote devices, passwords and keys that are used through transparent connections still benefit from the following CyberArk security features:

- **Dual Control** – If the Master Policy enforces dual control access approval, users require confirmation from authorized users before they can use a password or key to connect to a remote machine, in the same way as when credentials are retrieved directly.
- **Reason** – If the Master Policy requires users to specify a reason before a password or key can be retrieved, this also applies before a password or key can be retrieved for use in a transparent connection, in the same way as when they are retrieved directly.
- **Ticketing System Integration** – Requests to connect to remote devices can also require users to enter a ticket ID that will be validated against the enterprise ticketing system before connecting to the remote device.

The Privileged Account Security solution offers the following methods for privileged SSO:

- **PSM Connection** – Users connect to remote target applications and systems through the PSM proxy machine via the PWWA, using a secure protocol. The PSM isolates all privileged sessions to remote machines, and can record all the activities that occur during a privileged session and store them in the tamper-proof Vault for monitoring and auditing. The PSM server creates a single point of control for all privileged session activity that cannot be bypassed by even the most skilled users. Auditors and security teams can track privileged sessions through a unique interface and view recordings according to authorizations.

While EPV Transparent Connections are mainly used for convenience and usability purposes, the PSM Connection provides complete isolation of target systems and ensures that privileged credentials can never reach users or their devices. PSM Connections prevent users from bypassing controls to gain unmonitored access to privileged accounts and prevent malware or other attack techniques from obtaining and misusing privileged credentials.

Users can use PSM Connections to transparently log onto applications or remote machines, including Windows machines, VMWare machines, databases, SSH devices such as UNIX, Linux, routers and switches, and more.

- **PSM SSH Proxy (PSMP) Connection** – Users connect to remote Unix servers through the PSM SSH proxy machine, using a native Unix user experience. Like the PSM, the PSM SSH proxy isolates all privileged sessions to remote machines, and records all session activities and stores them in the Vault for monitoring and auditing.

The PSMP can also integrate with Microsoft's Active Directory (AD) to provision users transparently on UNIX systems, streamlining user management and reducing administrative overhead.

The PSMP enables Unix administrators to use SCP protocol to copy files securely from remote machines to your local machine and visa versa. Using the SCP protocol with PSMP ensures that users are not exposed to the privileged credentials on the target systems (privileged SSO) while copying the files.

In addition to other authentication methods, users can authenticate to the Vault through PSMP with private SSH key files. These private SSH keys can be kept on smart card devices and facilitate an even stronger authentication policy. The combination of this strong authentication with Active Directory integration allows transparent user provisioning on Unix systems, based on their strong authentication to the Vault.

- **EPV Transparent Connection** – Users connect to remote target applications and systems directly from the browser via the PVWA using preconfigured connection components (the password is transparently sent to the end-user browser without being exposed to the user). This method increases usability and users benefit from a seamless workflow.

Users can utilize the EPV transparent connections to transparently log onto remote Windows machines and SSH devices, such as UNIX, Linux, routers, switches and more.

Note: RDP transparent connections from the PVWA are not automatically supported when connecting from a non-Windows environment. For more information, refer to *Configuring PSM Connections and EPV RDP Connections that Require an External Tool*, page 550.

For information about configuring transparent connections through Password Vault Web Access, refer to *Configuring Transparent Connections*, page 527.

Users require the following authorizations in the Safe in order to be able to connect transparently to remote machines:

- **Use accounts** – To connect transparently through PSM and PSMP.
- **Retrieve accounts** – To connect transparently through EPV.

The following table summarizes the requirements and characteristics of each privileged SSO logon method:

Privileged SSO Logon Method	EPV Transparent Connection	PSM/P Privileged SSO
Privileged SSO to target device	Yes	Yes
Password is not exposed to the end user	Yes	Yes

Privileged SSO Logon Method	EPV Transparent Connection	PSM/P Privileged SSO
Password never reaches the end user's machine	No	Yes
Requires PSM/P server installation	No	Yes
Required permissions	Retrieve account	Use account
Enables session recording	No	Yes
Enables secure remote access	No	Yes
Cross network access to privileged devices (over HTTPS)	No	<ul style="list-style-type: none"> ▪ PSM-Yes ▪ PSMP-Connections are made over SSH port (22)
Controlled session duration	No	<ul style="list-style-type: none"> ▪ PSM – Yes ▪ PSMP – No
Separation between end user and target devices	No	Yes

Connecting Transparently to Remote Devices

Regardless of whether the privileged SSO method is implemented for EPV, PSM or PSMP, users can transparently log on to target applications and systems from the PVWA interface using preconfigured connection components. If more than one connection component has been configured for the platform that this account is associated with, you can select the connection component to use.

Connecting Transparently with PSM

Using PSM privileged SSO, users can transparently log onto a variety of systems and applications, including Windows machines, SSH devices such as UNIX, Linux, routers and switches, VMWare machines, databases and more.

The following table lists the end user platforms that you can use to transparently connect through the PSM, and the supported connection methods for each one:

End user platform	Supported	Connection methods
Windows	✓	ActiveX, RDPFile, external tool
Mac	✓	RDPFile, external tool
Unix/Linux	Supported with an external tool.	External tool

To connect from Mac platforms with an RDP file:

- Use the official Microsoft Remote Desktop app, which can be downloaded from: <https://itunes.apple.com/us/app/microsoft-remote-desktop/id715768417?mt=12>
- Clipboard redirection is not supported.

The following methods are available to establish PSM connections:

- **ActiveX** – This method is available when connecting through an IE browser, using Windows RDP ActiveX.

Note: This method does not support connections to target systems where NLA is enabled on the PSM server.

- **RDP File** – This method is available when connecting through non-IE browsers from Windows environments or when connecting from Mac environments. This method does not require any installation on the client side. The user will be prompted to download an RDP file and will have to open this file to establish the connection. It is recommended to allow these RDP files to open automatically in order to streamline the connection process. This is the default method.



Notes:

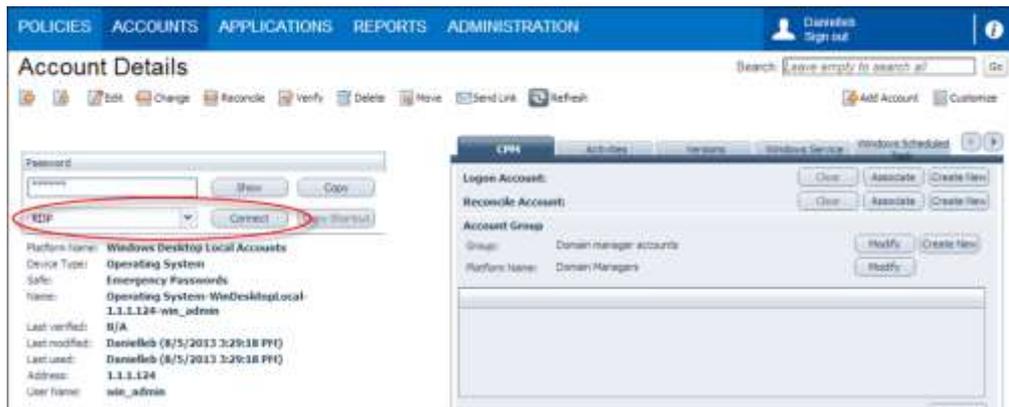
- This method does not support connections to target systems where NLA is enabled on the PSM server or RD Gateway is enabled in the environment.
- For each connection, an RDP file is downloaded to the Downloads folder of your browser. To use these RDP files, either open them manually from the browser or configure them to open automatically as recommended above. After the session has ended, its RDP file isn't valid anymore and cannot be reused. In addition, these RDP files are valid for connections for only a short time after they are downloaded and cannot be used after this time. You can delete these files manually from the Downloads folder.
Make sure that the folder to which the RDP files are downloaded is a private and protected storage, which is only accessible to the user who downloaded these files.
- **External Tool** – This method is available when connecting from non-Windows environments or when NLA is enabled on the PSM server. For more information, refer to *Configuring PSM Connections and EPV RDP Connections that Require an External Tool*, page 550.

Note: This method does not support connections to target systems where RD Gateway is enabled.

For more information about configuring PSM connection methods and their impact on the user experience, refer to *Configuring the PSM Session User Experience*, page 610.

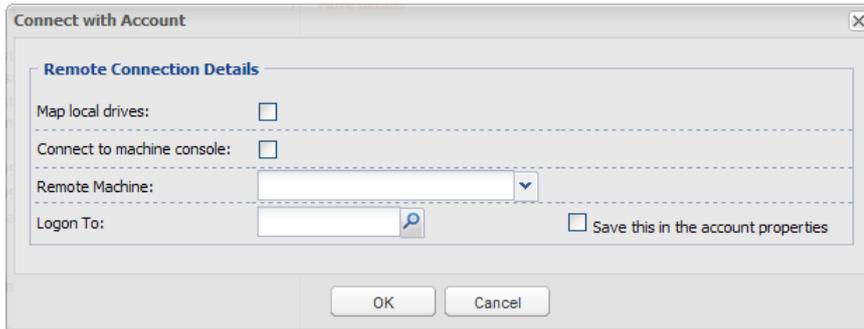
Connecting to a Remote Windows Device Transparently (RDP)

- In the Accounts Details page:
 1. Display the Accounts Details page of the account to use to log onto the remote device.
 2. If multiple connection components have been configured for this account, from the connection component drop-down list, select the connection component to use to log on.
The built-in connection component for **RDP connections via PSM** is PSM-RDP, which is automatically invoked for Windows accounts and does not require the user to select it manually.
 3. Click **Connect**.



or,

- In the Accounts List page or the Versions tab of the Account Details page:
 1. In the Accounts List page, display the account to use to log onto the remote database,
or,
In the Account Details page of the account to use to log onto the remote database, display the Versions tab.
 2. From the connection component drop-down list, select the connection component to use to log on, then click **Connect**.
 - If you are required to provide additional information before you can use the password, a window prompts you for the relevant information. For more information, refer to *Accessing Accounts*, page 243.
 - If you do not need to provide any additional information, the password will be used to log you onto the remote device.
 3. If you try to connect to the remote device with a domain/NIS user that requires you to specify the name or address of the remote device, the Connect with Account window appears to enable you to specify the required details. The following example shows the Connect with Account window that appears when you log onto Windows Domain accounts.



- i. To connect your local drives to the remote computer, select **Map local drives**.

Note: This is not supported for remote devices that run on Windows 2000.
- ii. To connect to the machine console, select **Connect to machine console**.
- iii. In **Remote Machine**, specify the remote machine to connect to. A drop-down list displays the most recent remote machine addresses that this account was used to connect to transparently with your user account. You can either select one of the listed addresses, or specify a different one.

If you are connecting to a remote Windows device with a local user, you will not be asked to specify the remote machine that will be logged onto transparently.
- iv. In **Logon To**, specify the NETBIOS domain that this user belongs to. For example, mycompany_dom.

The PWWA can try to detect the NETBIOS domain name automatically based on the address property of the account. For example, a domain whose full name is **mycompany.com** might have a NETBIOS name **mycompany_dom**, which users would specify here.

4. If you are required to create a request for confirmation before you can use this password, and you are prompted to specify one or more machines in the request, you will only be able to log onto the machine(s) you specified in the request after you receive confirmation.

You can specify multiple machine addresses in either of the following ways:

- **Any machine** – In **Remote Machine**, specify '*' (asterisk).
- **Multiple machines** – In **Remote Machine**, specify multiple machine addresses separated with a comma. For example, 1.1.1.174, 1.1.1.228, 1.1.1.235. The next time you are prompted for remote connection details, these remote machine addresses will be listed in a drop-down list.

For more information about requests, refer to *Dual Control*, page 251.

5. If your system requires a special tool to connect to a remote device, the first time you connect, the following window prompts you for a confirmation to run this tool.

For more information about when this tool is needed, refer to *Configuring PSM Connections and EPV RDP Connections that Require an External Tool*, page 550.

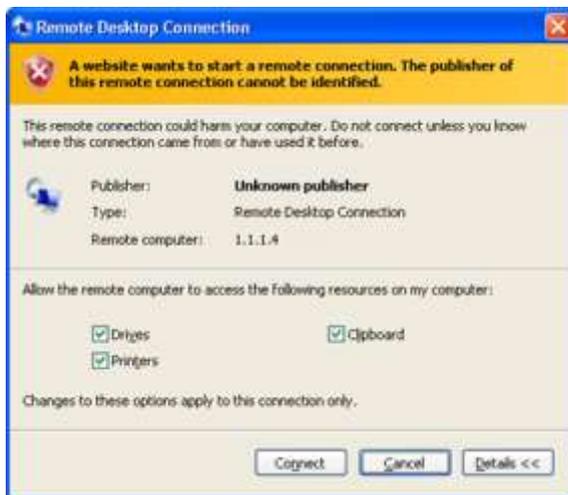


Click **Run** to make the remote connection and begin the privileged session.

6. If the transparent connection is configured to connect your local drives to the remote computer, one of the following windows will appear depending on the version of the RDP application on the remote machine:
- If the following window appears, make sure that **Connect your local disk drives to the remote computer** is selected, then click **OK**.



- If the following window appears, check **Drives**, then click **Connect**.



The PWA will use the remote connection details to logon to the remote device.

Note: To pass 'Alt + Ctrl + Del' to the target system, use the following alternative sequence: 'Alt + Ctrl + End'

Connecting to a Remote SSH Device Transparently

- In the Accounts Details page:
 1. Display the Accounts Details page of the account to use to log onto the remote device.
 2. If multiple connection components have been configured for this account, from the connection component drop-down list, select the connection component to use to log on.
 3. Click **Connect**.
- or,
- In the Accounts Details page or the Versions tab of the Account Details page:
 1. In the Accounts List page, display the account to use to log onto the remote database,
or,
In the Account Details page of the account to use to log onto the remote database, display the Versions tab.
 2. From the connection component drop-down list, select the connection component to use to log on, then click **Connect**.
 - If you are required to provide additional information before you can use the password, a window prompts you for the relevant information. For more information, refer to *Accessing Accounts*, page 243.
 - If you do not need to provide any additional information, the password will be used to log you onto the remote machine.
 3. If you try to connect to with a domain/NIS user that requires you to specify the name or address of the remote machine, the Connect with Account window appears to enable you to specify the required details.



4. Specify the address of the remote machine to log onto, then click **Connect**; the PVWA will use the remote connection details to logon to the specified remote machine.
5. If you are required to create a request for confirmation before you can use this password, and you are prompted to specify the machine in the request, you will only be able to log onto the machine you specified in the request after you receive confirmation. For more information about requests, refer to *Dual Control*, page 251.

Connecting to Remote Devices with X-Forwarding through the PSM

Users can connect to remote Unix devices through the PSM using X-Forwarding in addition to SSH protocol. As in all PSM connections, users do not need to know the privileged password or key content and the entire session can be recorded for auditing.

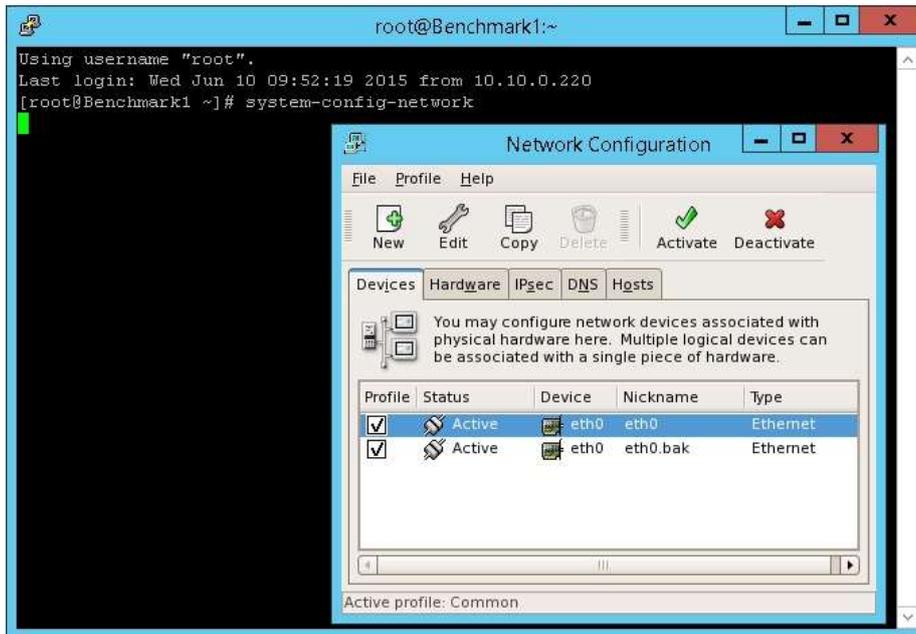
To Enable X-Forwarding

1. In the ADMINISTRATION page, click **Options**; the Web Access Options are displayed.
2. Expand **Connection Components**, then expand **PSM-SSH**, and then **Client Specific**; the dynamic parameters for specific clients are displayed.
3. Set the **EnableXForwarding** parameter to **Yes**.
4. Click **Apply** to apply the new configurations immediately,
or,
Click **OK** to save the new configurations and return to the System Configuration page.

To Connect to a Remote Device with X-Forwarding

- In the Accounts Details page:
 1. Display the Accounts Details page of the account to use to log onto the remote device.
 2. If multiple connection components have been configured for this account, from the connection component drop-down list, select the connection component to use to log on.
 3. Click **Connect**.
- or,
- In the Accounts Details page or the Versions tab of the Account Details page:
 1. In the Accounts List page, display the account to use to log onto the remote database,
or,
In the Account Details page of the account to use to log onto the remote database, display the Versions tab.
 2. From the connection component drop-down list, select the connection component to use to log on, then click **Connect**.
 - If you are required to provide additional information before you can use the password, a window prompts you for the relevant information. For more information, refer to *Accessing Accounts*, page 243.
 - If you do not need to provide any additional information, the password will be used to log you onto the remote machine.
 3. The PSM opens a second window in which you issue commands to the remote device. You can type any X commands that the logged on user is authorized to perform.
Note: There is no need to specify the DISPLAY variable.

The following example shows the network configuration X application screen displayed on the PSM transparent connection window to the remote device.



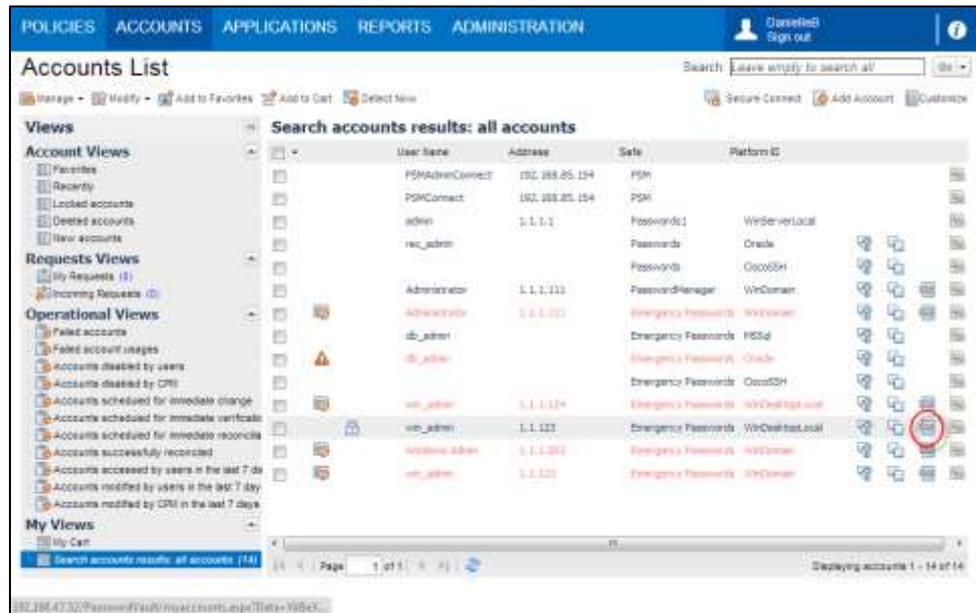
Note: To switch between open X windows, use 'Alt + Page up' or 'Alt + Page down'.

Connecting to Databases through the PSM

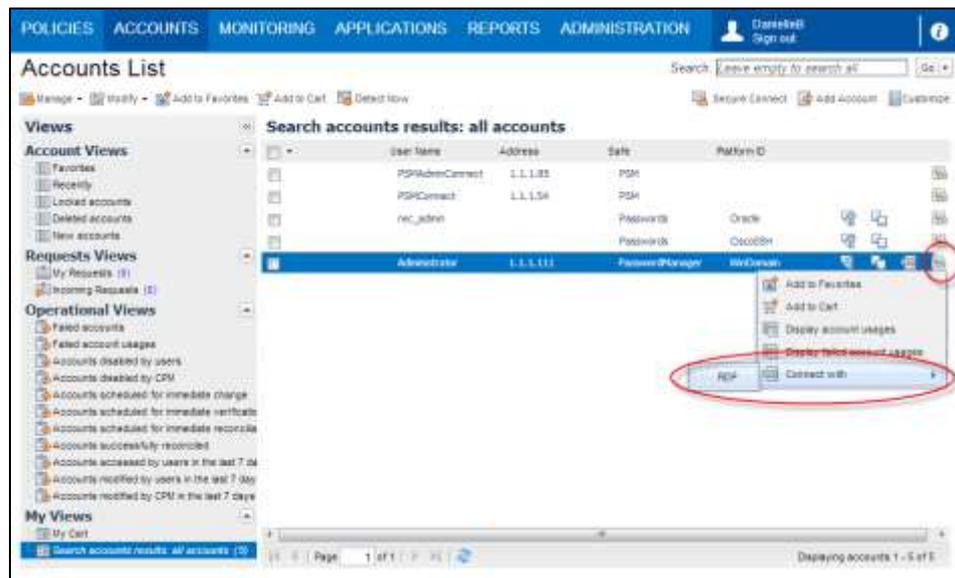
The PWA enables you to log onto remote Oracle databases using a different user during the transparent logon procedure.

To Connect to a Database Transparently

- In the Accounts List:
 1. In the Accounts List, display the account to use to log onto the remote database.
 2. From the connection component drop-down list, select the connection component to use to log on.
 - If there is only one available connection component, click the **Connect with** icon:



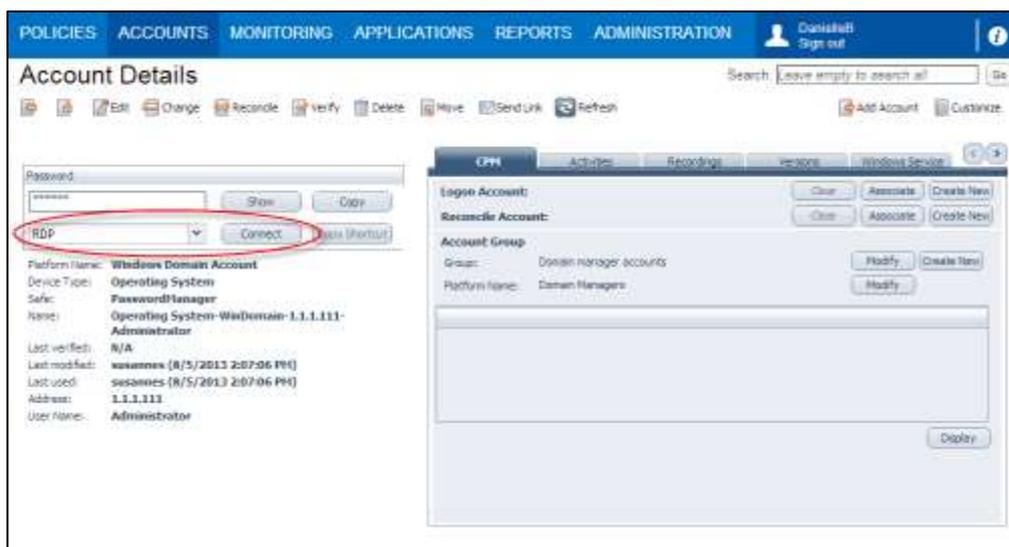
- If there is more than one available connection component, click the **Action menu** icon, then click **Connect with**, and select the connection component to use to connect to the remote machine:



The PWA will use the specified details to logon to the remote database using the specified PSM connection component.

or,

- In the Accounts Details page:
 - Display the Accounts Details page of the account to use to log onto the remote database.
 - From the connection component drop-down list, select the connection component to use to log on.
 - Click **Connect**.



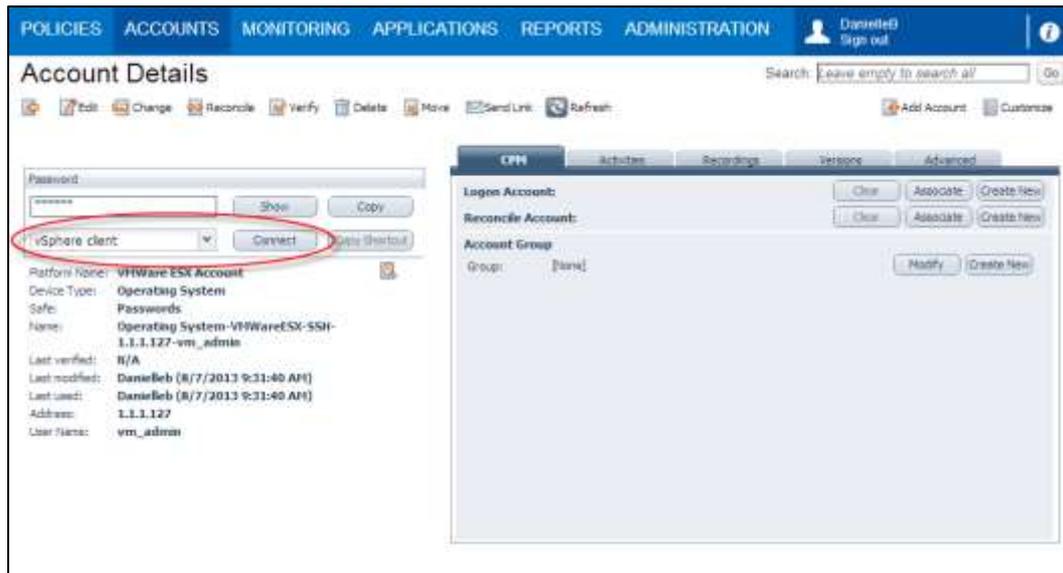
If the connection component enables this user to log onto the remote database with a different user, the Connect with Account window appears.

4. When connecting with the SYS user or any other registered privileged user, a Connect As drop-down list is displayed. From the Connect As drop-down list, select the role that will be used to connect to the remote database.
5. Click **OK**; the PVWA will use the specified details to logon to the remote database using the specified PSM connection component.

Connecting to VMWare Administrative Tools through the PSM

To Connect to a VMWare ESX Machine Transparently

- In the Accounts List page:
 1. In the Accounts List page, display the account to use to log onto the remote machine.
 2. From the connection component drop-down list, select the connection component to use to log on.
- or,
- In the Accounts Details page:
 1. Display the Accounts Details page of the account to use to log onto the remote machine.
 2. From the connection component drop-down list, select the connection component to use to log on, then click **Connect**.



The PWA will log onto the remote ESX using the specified PSM connection component.

To Connect to a vCenter Transparently using a Personal Account

- In the Accounts List page:
 1. In the Accounts List page, display the machine account to use to log onto the remote machine.
 2. From the connection component drop-down list, select the connection component to use to log on.

or,

- In the Accounts Details page:
 1. Display the Accounts Details page of the machine account to use to log onto the remote machine.
 2. From the connection component drop-down list, select the connection component to use to log on, then click **Connect**.

The user is prompted for their password again and then is logged onto the remote vCentre machine using the specified PSM connection component.

To Connect to a vCenter Transparently using a Shared Account

- In the Accounts List page:
 1. In the Accounts List page, display the machine account to use to log onto the remote machine.
 2. From the connection component drop-down list, select the connection component to use to log on.

or,

- In the Accounts Details page:
 1. Display the Accounts Details page of the machine account to use to log onto the remote machine.
 2. From the connection component drop-down list, select the connection component to use to log on, then click **Connect**.

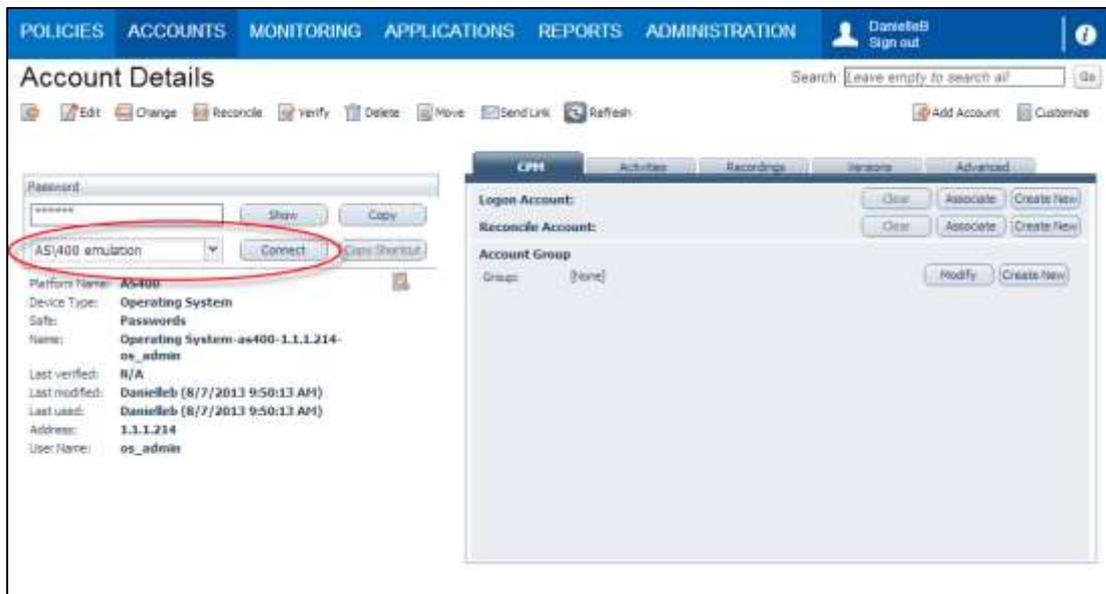
The PWA will log onto the remote vCenter machine with the shared account, using the specified PSM connection component.

To Connect to a Remote Machine Transparently using an AS\400 Account

- In the Accounts List page:
 - In the Accounts List page, display the machine account to use to log onto the remote machine, then click the **Connect with** button.
 - If more than one connection component has been defined for this platform, select the connection component to use to log on.

or,

- In the Accounts Details page:
 1. Display the Accounts Details page of the machine account to use to log onto the remote machine.
 2. From the connection component drop-down list, select the connection component to use to log on, then click **Connect**.



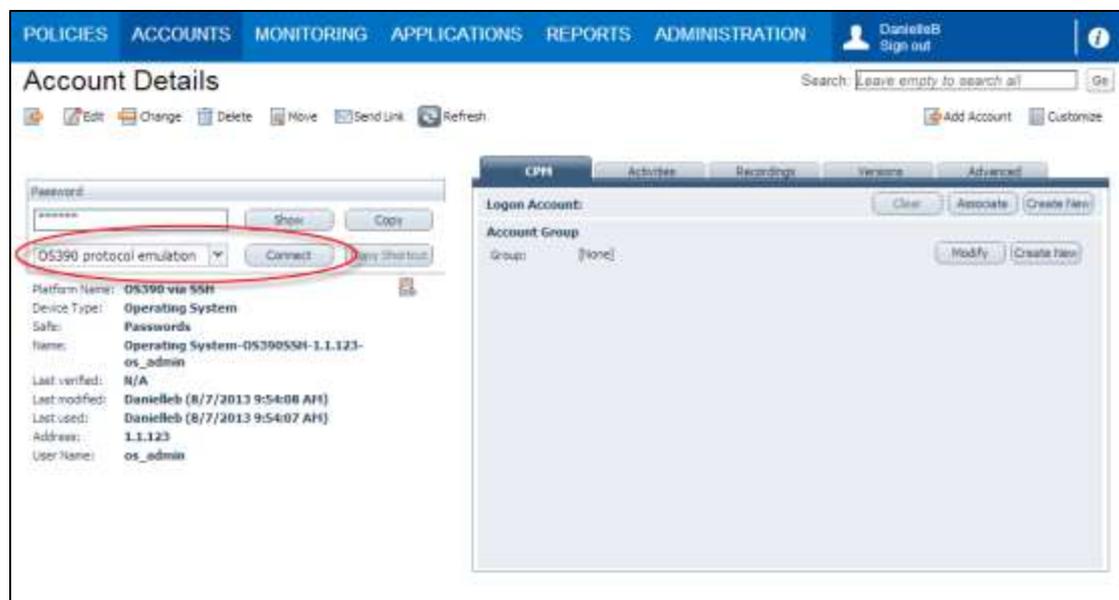
The PWA will log onto the remote machine with the AS\400 account, using the specified PSM connection component.

To Connect to a Remote Machine Transparently using an OS/390 (Z/OS) Account

- In the Accounts List page:
 1. In the Accounts List page, display the machine account to use to log onto the remote machine.
 2. From the connection component drop-down list, select the connection component to use to log on.

or,

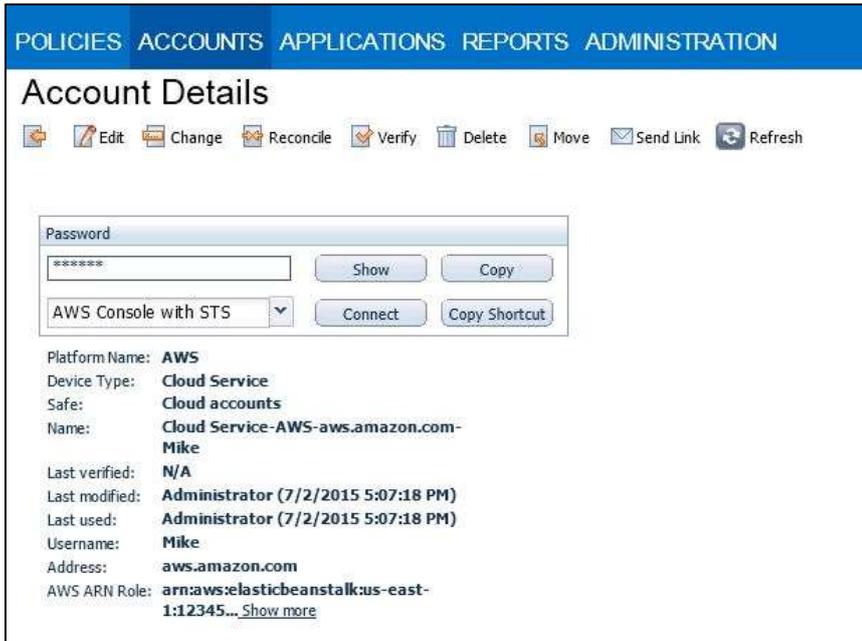
- In the Accounts Details page:
 1. Display the Accounts Details page of the machine account to use to log onto the remote machine.
 2. From the connection component drop-down list, select the connection component to use to log on, then click **Connect**.



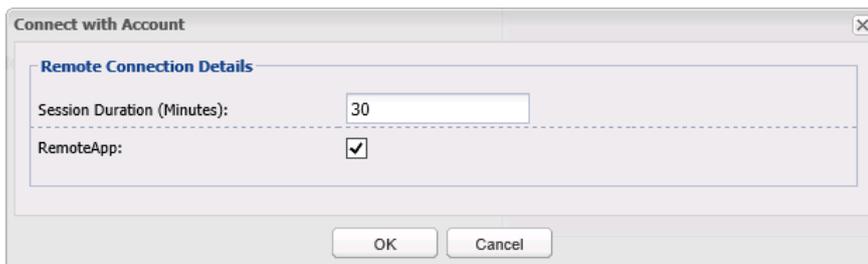
The PWA will log onto the remote machine with the OS/390 (Z/OS) account, using the specified PSM connection component.

Connecting to an Amazon Web Services (AWS) Platform Transparently through the PSM

1. Display the account to use to log onto the remote machine.
 - In the Accounts List page, display the account to use to log onto the remote machine.
 or,
 - Display the Accounts Details page of the account to use to log onto the remote machine.
2. From the connection component drop-down list, select **AWS Console with STS**.



3. Click **Connect**; the Connect with Account window appears.



4. Click **OK** to start the remote session and log onto the remote machine.

Connecting Transparently with PSMP

The PSM SSH Proxy (PSMP) enables you to connect to remote Unix devices with a native user experience through any SSH client, such as plink, PuTTY, SecureCRT, etc.

You can authenticate to the Vault through PSMP with CyberArk passwords, LDAP, RADIUS or with a private identity file (a private SSH key) whose corresponding public key was assigned to your user in the Vault.

For information about configuring authentication methods that will be available for PSMP connections in your environment, refer to *Configuring Authentication Methods*, page 693

Connecting to a Remote SSH Device

The following sections describe the PSMP command syntax and parameters, and how to use this command to connect to your target systems using password (CyberArk, LDAP or RADIUS) or private SSH key authentication.

Accessing Target Machines using the PSMP Command

The PSMP can be used by any ssh client using one of the following syntaxes:

Option 1:

```
<ssh client> [-i private_key_file]
vaultuser@targetuser#domainaddress@targetmachine#targetport@targetpas
sword@proxyaddress
```

Notes:

- Parameters are separated by '@'.
- Required parameters are separated from optional parameters by '#' (hash sign).
- You can customize the default delimiters that are used by the PSMP ('@', '#'). For example, you can change the '@' delimiter to enable connections with a domain or other user names that include this character. For information about configuring PSMP syntax delimiters, refer to *Configuring PSMP Syntax Delimiters*, page 717.
- This syntax is not supported when PSMP is installed on SUSE. Use Option 2 instead.
- To use this syntax, the CyberArk SSHD service must be installed. For more information about installing this service, refer to *Installing the Privileged Session Manager SSH Proxy* in the Privileged Account Security Installation Guide.

Option 2:

```
<ssh client> [-L <srcport>:localhost:target_port] -t
PSMConnect@<proxyaddress> <vaultuser> <targetuser> <targetmachine> [-
protocol <telnet|ssh>] [-port <port>] [-vp <vault-password>] [-tpw <
targetpassword>] [-tunnel <target_port>]
```

Notes:

- This syntax is only supported when the PSMP is installed in RHEL environments.
- SSH tunneling is only supported when using this syntax. To use SSH tunneling, the CyberArk SSHD service must be installed. For more information about installing this service, refer to *Installing the Privileged Session Manager SSH Proxy* in the Privileged Account installation Guide.
- For a full description of the parameters used in this syntax, refer to *Appendix F: Accessing Target Machines through PSMP*, page 985.

The PSMP Parameters

The following table explains the parameters used in option 1, above.

Parameter	Description	Required
[-i private_key_file]	<p>The path of the file from which the private key for SSH key authentication is read. This is an optional parameter and must be specified when SSH key authentication is used.</p> <p>For more information about this parameter and the different ways to specify private SSH keys, refer to SSH documentation.</p> <p>For information about SSH key authentication to the Vault, refer to <i>Authenticating to the Vault through PSMP using a Private SSH Key</i>, page 287.</p>	No
vaultuser	The name of the Vault user running this command.	Yes
targetuser	<p>The name of the account that will be used on the target system. For example, root.</p> <p>Note: This parameter is not required to connect through AD Bridge.</p>	Yes (not for AD Bridge)
Domainaddress	<p>The IP address or DNS of the domain server in the domain where the target machine resides.</p> <p>Note: For centralized account management, this parameter can be used to access multiple target systems with one account, even if they're not on the same domain. In this case, this parameter specifies the address in the centralized account and not the domain server.</p>	No
targetmachine	<p>The address of the target system in any of the following formats:</p> <ul style="list-style-type: none"> ▪ IPv4 – For example, 1.1.1.1 ▪ IPv6 – For example, 1000-1000-1000-1000-1000-1000-0055 <p>Note: Use hyphens instead of colons as separators.</p> <ul style="list-style-type: none"> ▪ DNS – For example, 'myhost' <p>As the PSMP resolves DNS names to IP addresses when necessary, you can specify either the machine's DNS name or an IP address, regardless of whether the account of the target machine was defined with an IP address, subnet or DNS name.</p>	Yes
targetport	<p>The connection port used to access the system. If this is not specified, it will be taken from the account properties. If neither of these ports is specified, the default port is used.</p> <p>Default values are:</p> <ul style="list-style-type: none"> ▪ SSH – 22 (used by default if no port is specified) ▪ Telnet – 23 <p>The protocol (SSH or Telnet) is set according to the specified port.</p>	No

Parameter	Description	Required
targetpassword	The password of the target account. This parameter is only relevant when privileged SSO is not enabled and the password is not managed in the Vault. If this password is not specified, the user is prompted for it.	No
proxyaddress	The IP address or DNS of the PSMP machine. For example, 1.1.1.1 or 'myhost'.	Yes

Authenticating to the Vault through PSMP using a Password

To connect to your target systems through PSMP while authenticating with a password (CyberArk, LDAP or RADIUS):

1. At a command line, run the command to access a target machine through the PSMP.

```
<ssh client>
vaultuser@targetuser#domainaddress@targetmachine#targetport@target
password@proxyaddress
```

2. You will automatically be prompted for your Vault password and any parameters, mandatory or optional, that you did not specify in the command line.

Example 1: Running sessions with Privileged SSO

- The following example initiates an SSH privileged SSO session. The command contains all the information that is required to log onto the target system through the PSMP.

```
ssh john@root@ciscorouter.com@psmp.proxymachine.com
```

In this example, a Vault user called **john** will access the Vault and retrieve an account for the **root** user on the target system, **target.ciscorouter.com**. As this command does not specify a port, the default port, **22**, and protocol, **SSH**, will be used.

John will be prompted for his Vault password so that the PSMP can retrieve information that is required to connect to the target machine. The account stored in the Vault for the target system is configured for Privileged SSO and contains the password or private SSH key that is required to access the target system. Therefore, the user will be logged on to the target system transparently without needing to specify any more credentials.

- The following example initiates a Telnet privileged SSO session.

```
ssh john@root@ciscorouter.com#23@psmp.proxymachine.com
```

Similar to the previous example, a Vault user called **john** will access the Vault and retrieve an account for the **root** user on the target system, **target.ciscorouter.com**. However, this command specifies port **23**, which indicates **Telnet** protocol.

John will be prompted for his Vault password so that the PSMP can retrieve information that is required to connect to the target machine.

As in the previous example for Privileged SSO, the account stored in the Vault for the target system contains the password or the private SSH key that is required to access the target system and the user will be logged on transparently without needing to specify any other credentials.

Example 2: Running sessions without Privileged SSO

The following example initiates a non-privileged session.

```
ssh
john@root@ciscorouter.com#2222@targetciscorootpass@psmp.proxymachine.com
```

In this example, a Vault user called **john** will access the Vault and retrieve an account for the **root** user on the target system, **target.ciscorouter.com**. This command specifies port **2222**, so SSH protocol will be used.

This example shows a non-privileged SSO session, meaning that the account stored in the Vault for the target system is not configured for Privileged SSO and does not contain the password. Therefore, the password of the target system is specified in the command, **targetciscorootpass**. If this password is not specified in the command, the user is prompted for it so that the PSMP can complete the connection to the remote machine. John will also be prompted for his Vault password so that the PSMP can retrieve information that is required to connect to the target machine.

Example 3: Accessing Target Machines with a Domain/NIS Account

You can connect directly to a target machine with a UNIX domain/NIS account through the PSMP. To access target machines with a domain/NIS account, specify the domain machine in the command.

The following example shows how to access a target machines with a Domain/NIS account.

```
ssh
john@root#mycompany.com@target.mycompany.com@psmp.proxymachine.com
```

In this example, a Vault user called **john** will access the Vault and retrieve a domain account for the **root** user in the **mycompany.com** domain to access the target system, **target.mycomany.com**.

John will also be prompted for his Vault password so that the PSMP can retrieve information that is required to connect to the target machine.

If the target user is not specified, you will be prompted for it and then can specify the target user and the domain machine as shown in the following example:

```
Target user is required (to use domain account, specify
<target_user>#domain_address).
Target user: root#mycompany.com
```

Example 4: Specifying the Vault Password in the PSMP Command

If the SSH client supports the ability to pass the connecting user's password, for example plink, you can specify the Vault password as the SSH password, as shown in the following example:

```
plink -pw <vault password>
john@root@ciscorouter.com#2222@psmp.proxymachine.com
```

In this example, a Vault user called **john** will access the Vault and retrieve an account for the **root** user on the target system, **target.ciscorouter.com**. This command specifies port **2222**, so SSH protocol will be used.

The account stored in the Vault for the target system is configured for Privileged SSO and contains the password. Therefore, it's not necessary to specify the target password in this command. John's Vault password is included in the command, so he will not be prompted for it.

Authenticating to the Vault through PSMP using a Private SSH Key

You can connect to target systems through PSMP by authenticating to the Vault with a private SSH key file. This key can be provided with any standard SSH tool or client configuration. A corresponding public SSH key must be assigned to your user in the Vault to allow authentication.

Vault users can be assigned one or more public SSH keys that are kept for them in the Vault. If one of these keys matches the private SSH key provided by the user during authentication, the connection through PSMP will be approved and the user will be able to access their target system.

You can manage the public SSH keys that are authorized for Vault users with dedicated SSH Key web services. For information about these web services, refer to *Managing Authorized Public SSH Keys for Users* in the Privileged Account Security Web Services SDK Implementation Guide.

To connect through PSMP using SSH Key Authentication

1. At a command line, run the command to access a target machine through the PSMP.

```
<ssh client> [-i private_key_file]
vaultuser@targetuser#domainaddress@targetmachine#targetport@target
password@proxyaddress
```

Note: The private SSH key can be provided with the `-i` option or with any standard SSH tool or client configuration.

2. You will automatically be prompted for any parameters, mandatory or optional, that you did not specify in the command line. If the SSH key authentication is successful, you will not be prompted for a password.

Note: If SSH key authentication was refused, and password authentication is allowed for SSH connections on the PSMP server, you will be prompted for a password.

Example: Running a session with Privileged SSO and SSH key authentication

- The following example initiates an SSH privileged SSO session using SSH key authentication.

```
ssh -i ~/.ssh/id_rsa
john@root@ciscorouter.com@psmp.proxymachine.com
```

In this example, a Vault user called **john** will authenticate to the PSMP with a private SSH key stored in the `~/.ssh/id_rsa` file. Then, he will access the Vault and retrieve an account for the **root** user on the target system, **target.ciscorouter.com**.

The account stored in the Vault for the target system is configured for Privileged SSO and contains the password or private key that is required to access the target system. Therefore, the user will be logged on to the target system transparently without needing to specify any more credentials.

As this command does not specify a port, the default port **22** and **SSH** protocol will be used.

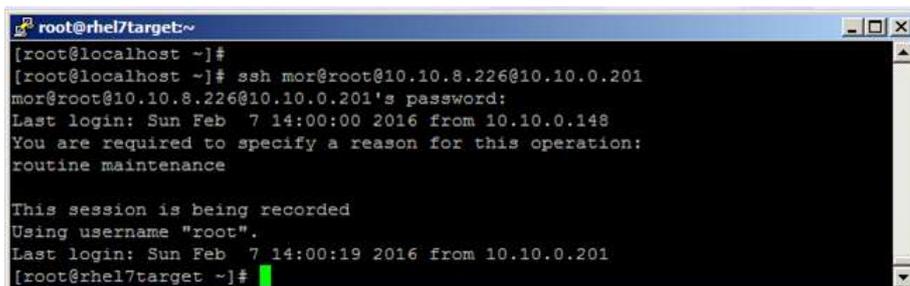
Specifying a Reason for Accessing Accounts through PSMP

A rule in the Master Policy determines whether users can only retrieve passwords or SSH keys after they specify a reason that explains why they need to retrieve them. If the rule is active, the user is prompted to provide the relevant information before the remote session begins.

Note: When copying files through PSMP, users will not be prompted for a reason.

To Specify a Reason for Accessing Accounts

1. After running the command to access a target machine through the PSMP, you will be prompted to type a reason for connecting. Specify the reason and press **Enter**.



```
root@rhel7target~
[root@localhost ~]#
[root@localhost ~]# ssh mor@root@10.10.8.226@10.10.0.201
mor@root@10.10.8.226@10.10.0.201's password:
Last login: Sun Feb  7 14:00:00 2016 from 10.10.0.148
You are required to specify a reason for this operation:
routine maintenance

This session is being recorded
Using username "root".
Last login: Sun Feb  7 14:00:19 2016 from 10.10.0.201
[root@rhel7target ~]#
```

2. The PSMP will now retrieve the password or SSH key, and the reason you specified will be stored in the audit log.

Copying Files Securely through PSMP

You can copy files securely from your local machine to target machines or visa versa in your native environment, using the PSMP together with SCP protocol.

The PSMP provides Privileged Single Sign-On when copying files with SCP and thus eliminates the need to know or specify the passwords or keys to the target machine.

When using SCP through PSMP, the PSMP will not prompt you for any required parameters that you do not specify. Make sure that you specify all mandatory parameters in the command.

Make sure that the client you are using to copy files uses SCP protocol. For example, PSCP (the PuTTY secure copy client) uses the SFTP protocol by default, so you must use pscp with the [-scp] option to enforce the use of SCP protocol.

To Copy Files to a Remote Machine

Use the following syntax to copy files securely from your local machine to a target machine:

```
scp <path-on-end-user-machine>  
vaultuser@targetuser%domainaddress@targetmachine%targetport@targetpas  
sword@proxyaddress:<path-on-target-machine>
```

Notes:

- Parameters are separated by '@'.
- In SCP syntax, '#' (hash) cannot be used as a delimiter. By default, required parameters are separated from optional parameters with '#' (hash). To use optional parameters in SCP, configure a different delimiter to replace the '#' (hash). In the above syntax, '%' is used as a delimiter to separate required and optional parameters. For more information, refer to *Configuring PSMP Syntax Delimiters*, page 717.

For a complete explanation of the PSMP syntax parameters, refer to *The PSMP Parameters*, page 284.

In the following example, a Vault user called **john** will connect as user **root** to the target machine, which is **10.10.10.5**, through a proxy machine whose IP address is **10.10.10.200**, and copy a file called **readme.txt** from the **/home** directory on the user's local machine to the **/tmp** directory on the target machine.

```
scp /home/readme.txt john@root@10.10.10.5@10.10.10.200:/tmp
```

To Copy Files from a Remote Machine

On your local machine, use the following syntax to copy files securely from a remote machine to your local machine:

```
scp  
vaultuser@targetuser#domainaddress@targetmachine#targetport@targetpas  
sword@proxyaddress:<path-on-target-machine> <path-on-end-user-  
machine>
```

Notes:

- Parameters are separated by '@'.
- In SCP syntax, required parameters are separated from optional parameters by a preconfigured delimiter which must be different from '#' (hash sign). In the above syntax, '%' is used as a delimiter. To use optional parameters in SCP, configure a character to use as a delimiter by adding the **PSMP_TargetAddressPortAdditionalDelimiter** parameter to the **sshd_config** configuration file. For more information, refer to *Configuring PSMP Syntax Delimiters*, page 717.

For a complete explanation of the PSMP syntax parameters, refer to *The PSMP Parameters*, page 284.

In the following example, a Vault user called **john** will connect as user **root** to the target machine, which is **10.10.10.5**, through a proxy machine whose IP address is **10.10.10.200**, and will copy all files and directories recursively from the **/tmp** directory on the target machine to the **/home** directory on the user's local machine.

```
scp -r john@root@10.10.10.5@10.10.10.200:/tmp /home
```

Connecting through PSMP with Active Directory Users

Users can connect to a UNIX machine through PSMP using their AD credentials. This automatically synchronizes their AD user with a corresponding user in the Vault.

To Access Target Machine using AD Bridge Capabilities:

- Use the following syntax to access the target machine using AD Bridge capabilities:

```
<ssh client> vaultuser@targetmachine#targetport@proxyaddress
```

Example: Running sessions with AD Bridge Capabilities

You can use AD Bridge capabilities to provision users transparently on a target machine and connect to it through the PSMP.

The following example shows how to initiate a privileged session using AD Bridge Capabilities.

```
ssh john@10.10.10.5@10.10.10.200
```

In this example, a Vault user called **john** will access the Vault and retrieve an account to access a machine whose IP address is 10.10.10.5 through a proxy machine whose IP is 10.10.10.200. If this user does not exist on the target machine, it will be created transparently and this user will be able to access the target machine through the PSMP.

Note: If this user does not exist in the Vault, it will be created transparently according to its AD credentials. For more information, refer to *Integrating with AD Bridge Capabilities*, page 724.

Connecting Transparently with EPV

Connecting to a Remote Windows Device Transparently (RDP)

- In the Accounts Details page:
 1. Display the Accounts Details page of the account to use to log onto the remote device.
 2. If multiple connection components have been configured for this account, from the connection component drop-down list, select the connection component to use to log on.
 - The following connection methods are available for **EPV Transparent Connections through RDP**. The system automatically selects the relevant connection, based on the browser type and the operating system from which the user connects:

Connection Method	Use when Connecting from	Comments
RDP (non-ActiveX)	Any browser on Windows	Requires Java installation on the client side.
RDP	IE (Windows)	Uses the default Windows RDP ActiveX. This method does not support connecting to target systems in which NLA is enabled on the PSM server.
RDP (from non-Windows)	Non-Windows environment	This method is not supported automatically. For more information, refer to <i>Configuring PSM Connections and EPV RDP Connections that Require an External Tool</i> , page 550.

3. Click **Connect**.



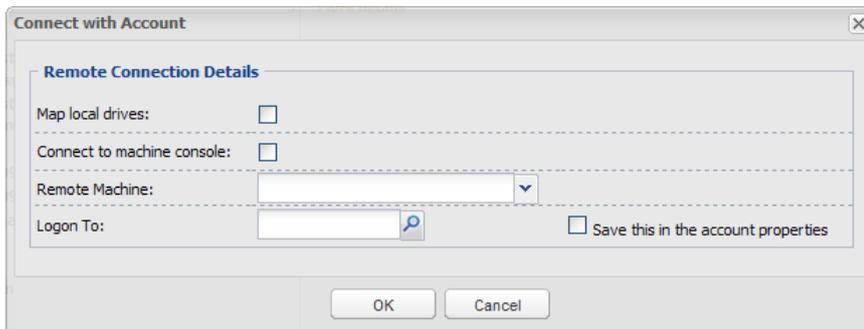
or,

- In the Accounts List page or the Versions tab of the Account Details page:
 1. In the Accounts List page, display the account to use to log onto the remote database,
 - or,
 - In the Account Details page of the account to use to log onto the remote database, display the Versions tab.

2. From the connection component drop-down list, select the connection component to use to log on, then click **Connect**.

For information about the available connection options, refer to the table in step 2 on page 291.

- If you are required to provide additional information before you can use the password, a window prompts you for the relevant information. For more information, refer to *Accessing Accounts*, page 243.
 - If you do not need to provide any additional information, the password will be used to log you onto the remote device.
3. If you try to connect to the remote device with a domain/NIS user that requires you to specify the name or address of the remote device, the Connect with Account window appears to enable you to specify the required details. The following example shows the Connect with Account window that appears when you log onto Windows Domain accounts.



- i. To connect your local drives to the remote computer, select **Map local drives**.

Note: This is not supported for remote devices that run on Windows 2000.
 - ii. To connect to the machine console, select **Connect to machine console**.
 - iii. In **Remote Machine**, specify the remote machine to connect to. A drop-down list displays the most recent remote machine addresses that this account was used to connect to transparently with your user account. You can either select one of the listed addresses, or specify a different one.

If you are connecting to a remote Windows device with a local user, you will not be asked to specify the remote machine that will be logged onto transparently.
 - iv. In **Logon To**, specify the NETBIOS domain that this user belongs to. For example, mycompany_dom.

The PWWA can try to detect the NETBIOS domain name automatically based on the address property of the account. For example, a domain whose full name is **mycompany.com** might have a NETBIOS name **mycompany_dom**, which users would specify here.
4. If you are required to create a request for confirmation before you can use this password, and you are prompted to specify one or more machines in the request, you will only be able to log onto the machine(s) you specified in the request after you receive confirmation.

You can specify multiple machine addresses in either of the following ways:

- **Any machine** – In **Remote Machine**, specify '*' (asterisk).
- **Multiple machines** – In **Remote Machine**, specify multiple machine addresses separated with a comma. For example, 1.1.1.174, 1.1.1.228, 1.1.1.235. The next time you are prompted for remote connection details, these remote machine addresses will be listed in a drop-down list.

For more information about requests, refer to *Dual Control*, page 251.

5. If your system requires a special tool to connect to a remote device, the first time you connect, the following window prompts you for a confirmation to run this tool.

For more information about when this tool is needed, refer to *Configuring PSM Connections and EPV RDP Connections that Require an External Tool*, page 550.

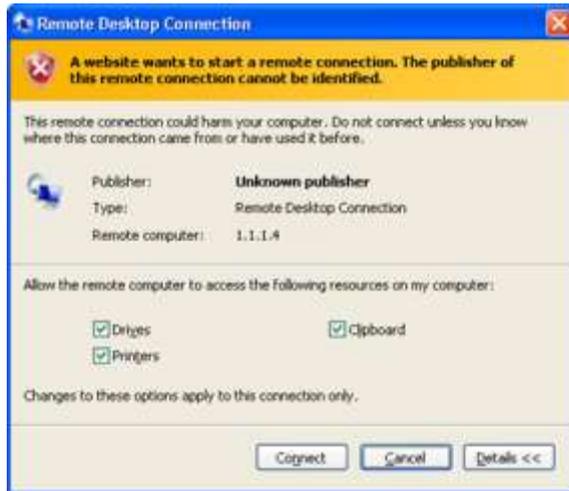


Click **Run** to make the remote connection and begin the privileged session.

6. If the transparent connection is configured to connect your local drives to the remote computer, one of the following windows will appear depending on the version of the RDP application on the remote machine:
 - If the following window appears, make sure that **Connect your local disk drives to the remote computer** is selected, then click **OK**.



- If the following window appears, check **Drives**, then click **Connect**.



The PWA will use the remote connection details to logon to the remote device.

Note: RDP transparent connections are not displayed in full screen view if the browser zoom is greater than 100%.

Connecting to a Remote SSH Device Transparently

- In the Accounts Details page:
 1. Display the Accounts Details page of the account to use to log onto the remote device.
 2. If multiple connection components have been configured for this account, from the connection component drop-down list, select the connection component to use to log on.
 3. Click **Connect**.
- or,
- In the Accounts Details page or the Versions tab of the Account Details page:
 1. In the Accounts List page, display the account to use to log onto the remote database,
 - or,
 - In the Account Details page of the account to use to log onto the remote database, display the Versions tab.
 2. From the connection component drop-down list, select the connection component to use to log on, then click **Connect**.
 - If you are required to provide additional information before you can use the password, a window prompts you for the relevant information. For more information, refer to *Accessing Accounts*, page 243.
 - If you do not need to provide any additional information, the password will be used to log you onto the remote machine.
 3. If you try to connect to with a domain/NIS user that requires you to specify the name or address of the remote machine, the Connect with Account window appears to enable you to specify the required details.



4. Specify the address of the remote machine to log onto, then click **Connect**; the PVWA will use the remote connection details to logon to the specified remote machine.
5. If you are required to create a request for confirmation before you can use this password, and you are prompted to specify the machine in the request, you will only be able to log onto the machine you specified in the request after you receive confirmation. For more information about requests, refer to *Dual Control*, page 251.

Connecting with Secure Connect

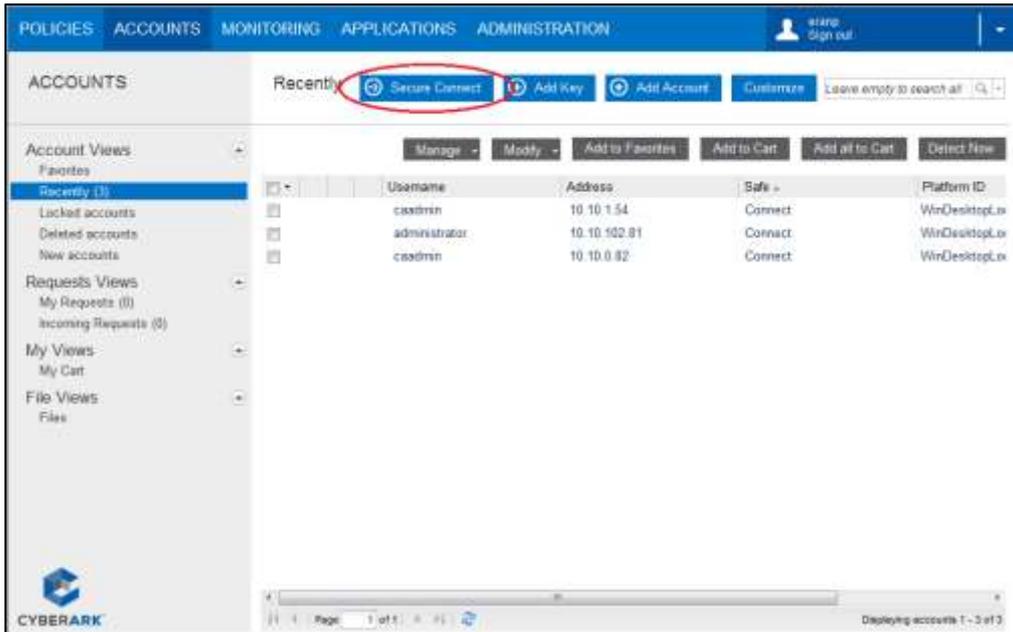
Users can connect to any machine through PSM using any account, including those that are not managed in the CyberArk Vault. All secure connect sessions benefit from the standard PSM features, including session recording, detailed auditing, and standard audit records. In addition, authorized users can monitor live sessions in real time, assume control, and terminate them when necessary.

You can configure multiple secure connect platforms, and define different settings for each one, such as recording Safes or a different PSM server. This way, you can create secure connect platforms that suit the network structure and your organizational business needs.

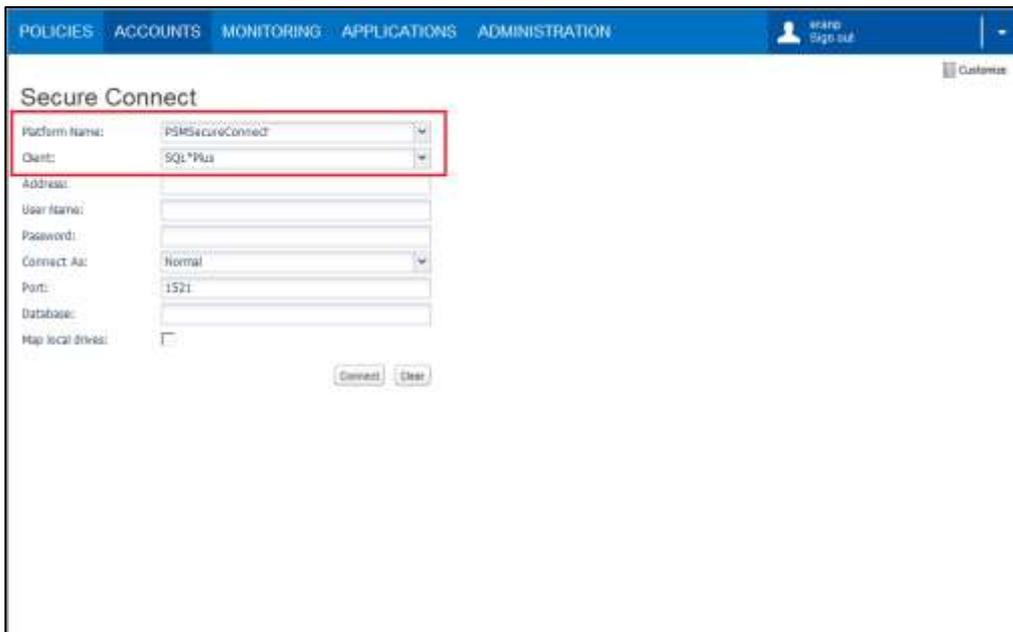
In the Secure Connect page, users select the secure connect platform and a client that enables them to log onto the remote device. Then they specify the address of the remote machine, and the user name and password that are required to log on, but which are not managed in the Vault.

To Connect to a Remote Device with Secure Connect

1. In the Accounts List, click **Secure Connect**; the Secure Connect page appears.



- From the Platform Name drop-down list, select the secure connect platform that will be used to connect to the remote machine.
- From the Client drop-down list, select the Secure Connect client that will be used; the information that is required for each client is displayed. The following example shows the Secure Connect page for the SQL *Plus client.

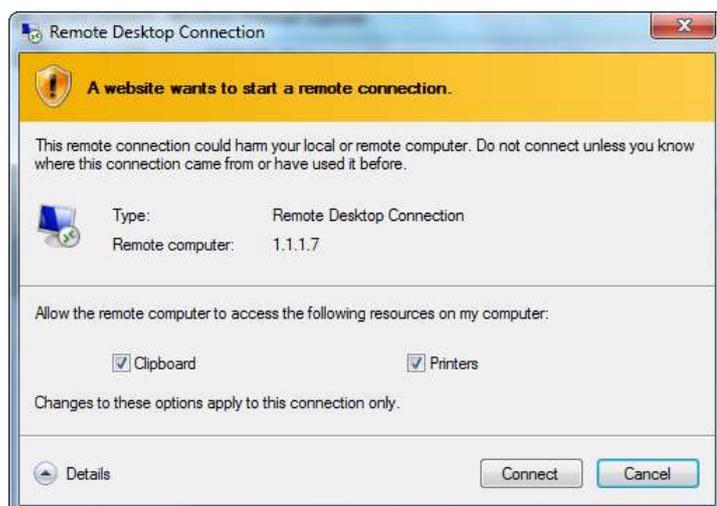


4. Specify the information that is required to create a secure connection to the remote machine. The following table lists the information that can be specified, as well as additional information for specific clients.

Information	Description
All Clients:	
Required information:	
Platform Name	The name of the secure connect platform that will be used to connect to the remote machine.
Client	The connection component that will be used to log onto the remote device.
Address	The IP/DNS address of the remote machine that the user will log on to.
User Name	The name of the user who will log onto the remote machine.
Password	The password that will be used to log onto the remote machine.
Optional information:	
Map local drives	Connects your local drives to the remote computer.
SQL Plus and Toad Clients:	
Required information:	
Connect As	The specific user role that will be used to log onto the remote machine.
Port	The port that will be used to log onto the remote machine.
Database	The remote database that the user will log onto.
SSH and WinSCP Clients:	
Required information:	
Port	The port that will be used to log onto the remote machine.
RDP Clients:	
Required information:	
Logon To	The specific user role that will be used to log onto the remote machine.
Port	The port that will be used to log onto the remote machine.
Optional information:	
Connect to machine console	Connects your local machine to the machine console.
PSM-SQLServerMgmtStudio Clients:	
Required information:	
Database	The remote database that the user will log onto.

Click **Clear** at any time to clear the details that you have specified and to redisplay the default values.

5. Click **Connect**; the following window appears:



6. Click **Connect**; the remote connection is made through the PSM and the secure connect session begins.

Viewing Secure Connect Sessions

You can view privileged sessions that have been initiated through Secure Connect in either of the following ways:

- **Active Sessions** – View active sessions in real-time in the Live Sessions view in the MONITORING page. For more information, refer to *Monitoring Live Sessions*, page 333.
- **Session recordings** – View finished secure connect session recordings in the MONITORING page. For more information, refer to *Monitoring Privileged Sessions*, page 315.

Accessing the Connection Window (Direct Access to Target Systems)

Users can directly access the Connect window used to log onto a remote devices through a direct URL or a desktop shortcut.

If a reason for access, a ticketing system, or dual control is enforced for the account, the relevant window will appear for the user to provide the required information. After the user has provided the correct information or has received authorization to access the account specified in the direct line, the Connection window will appear.

If a browser blocks pop-ups in the PVWA, enable the pop-up to display the Connect window.

You can create a shortcut in either of the following ways:

Specifying the shortcut manually

The following URL displays the Connection Component window, without needing to access the Account Details page:

```
http://<host name>/PasswordVault/directaccess.aspx?objectdetails.aspx&safe=<Safe name>&folder=<folder name>&object=<account name>&OpenConnectWindow=Yes
```

If a connection component is specified for the account referred to in the URL, the unique ID of the connection component can be included in the URL.

```
http://<host name>/PasswordVault/directaccess.aspx?objectdetails.aspx&safe=<Safe name>&folder=<folder name>&object=<account name>&OpenConnectWindow=Yes&ConnectionClient=<ConnectionComponentId>
```

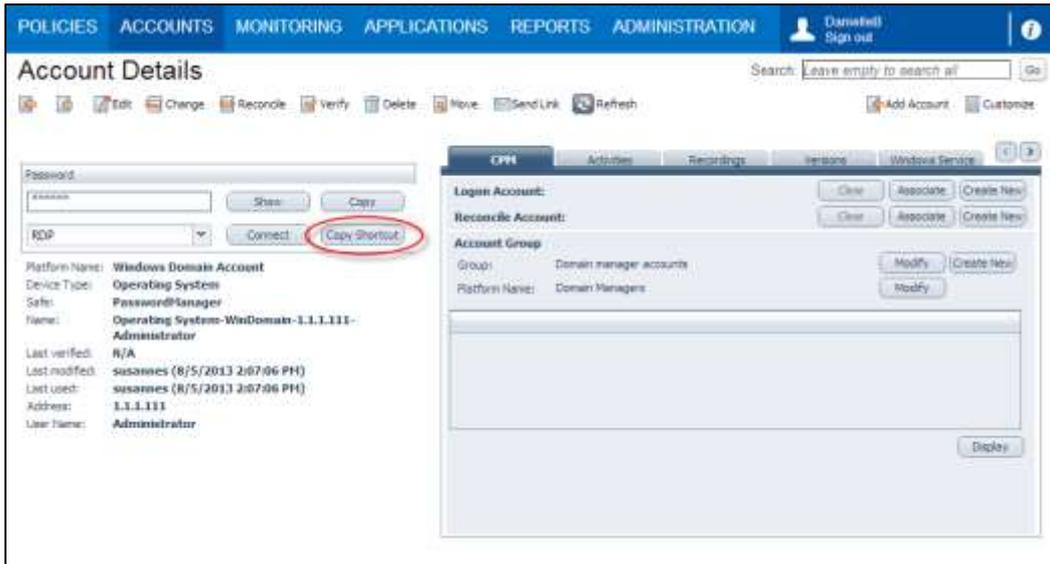
If the connection component is not specified, the default connection component will be used.

- Either use this URL to access the Account Details page directly,
or,
Create a desktop shortcut and specify this URL in the location of the shortcut item.

Copying the shortcut automatically

1. Display the Account Details page for the account to use to connect to the remote terminal.
2. Click **Copy Shortcut**; the PVWA creates a link that includes the transparent connection component that is displayed and copies it to the Connection window.

Note: This feature is active in Internet Explorer and Firefox.



3. On the desktop, create a new shortcut. When you are asked for the location of the shortcut item, paste the copied link into the edit box.

Working in Split Password Mode

Passwords in the PWA can be accessed in Split Password mode. This mode is recommended only when passwords are managed and changed by the CPM, when end users do not need the “Update password value” authorization. In cases where the CPM does not manage the account and change the password in it, it is recommended to save the password in two different objects in the Vault, and assign the relevant permissions to end users, based on the half of the password they need to access or change

The Split Password mode restricts users to accessing either the first half of a password or the second half. In this mode, users access passwords according to group membership which defines which half of the password they can access as well as their Safe authorizations. Users who have access to both halves of the password will be able to see the entire password.

Split password mode is managed by platforms which enables this mode and defines the user groups. For more information, refer to *Configuring Split Password Mode*, page 569.

Passwords that are configured for split password mode cannot be used in the following scenarios:

- Logging onto remote machines transparently.
 - Note:** Users who have the ‘Use accounts’ authorizations can log onto remote machines transparently through the PSM in split password mode.
- Exclusive password mode

Viewing Passwords

In Split Password mode, users access passwords in the same way as in the regular mode, but only the half of the password that they are permitted to see is displayed.

To View Passwords in Split Password Mode

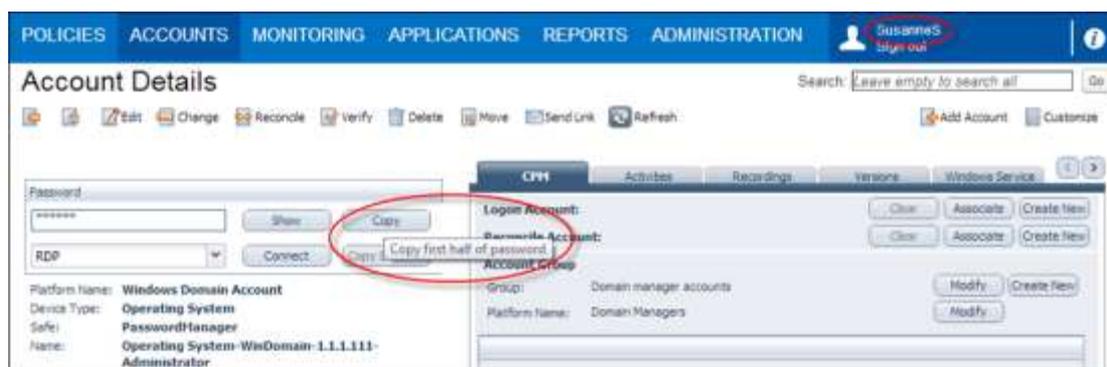
- In the Account Details page, click **Show**; the half of the password that the user is permitted to see is displayed.

Note: The tooltip for the Show button indicates which half of the password the user can see.

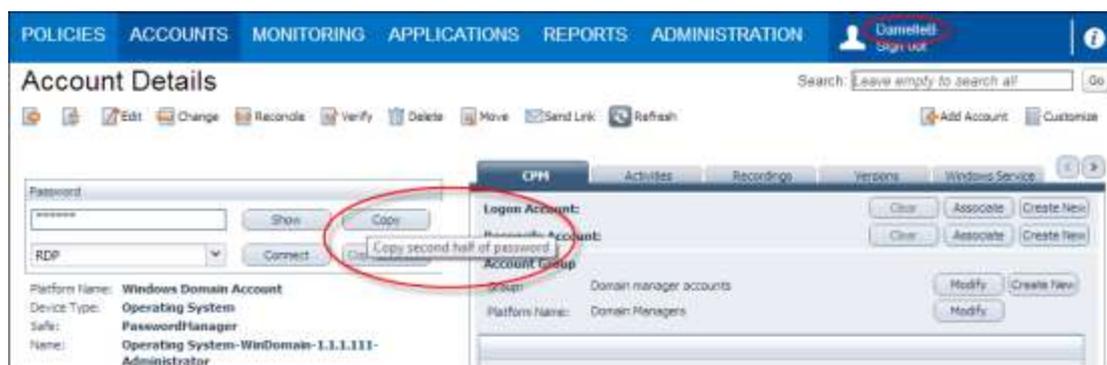
Copying Passwords

Users can only copy the half of the password that they are permitted to see. Tooltips on the copy icon in the Accounts List and on the Copy button in the Account Details page show which half of the password will be copied.

In the following example, Susanne can only copy the first half of the password. This is due to the fact that she is a member of the group that is configured to see the first half of the password.



On the other hand, Danielle can only copy the second half of the password. This is due to the fact that she is a member of the group that is configured to see the second half of the password.



Password Version Control

Authorized users can view versions of passwords in the Safe. The Versions tab in the Account Details page displays the different versions of the passwords that are currently retained in accounts in the Safe. In order to see the Versions tab, users require the following Safe member authorization:

- Retrieve accounts

Password versions are saved according to one of the following Safe configuration:

- **Previous versions** – A predetermined number of password versions are saved in the Safe.
- **Previous days** – All password versions from a predetermined number of days are saved in the Safe.

When an account is managed by the CPM, you may see temporary password versions (with a special indication), during the password change process. When a password change process ends successfully, the temporary version becomes a real version. If a password change process fails, the CPM reverts the password to the previously correct password. The temporary version will still be available in the versions list for troubleshooting purposes.

By default, temporary versions are not displayed in the list of password versions.

To View Password Versions

1. In the Accounts list, select the account that contains the password you wish to inspect; the Account Details page appears.
2. Select the **Versions** tab; a list of the versions of the selected password that are retained in the Safe is displayed in this pane.
3. By default, temporary password versions are not displayed in the list. Clear **Do not display CPM temporary password versions** to display both real and temporary password versions.
4. In the row of the required password version, click the relevant icon to show it, copy it, or connect with it to a remote machine.

Accessing Accounts through the Mobile PVWA

Users can access accounts stored in the Password Vault through mobile devices, providing ongoing access regardless of physical location.

The Mobile PVWA enables users to search for accounts and view passwords, as well as confirm requests for other users' access.

Logging onto the Mobile PVWA

The PVWA offers several authentication options for logging on to the mobile PVWA:

- CyberArk
- RADIUS
- RSA SecurID
- LDAP

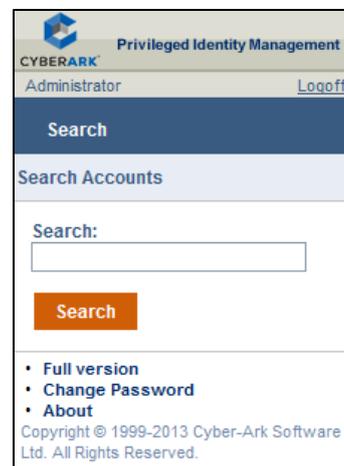
When you access the mobile PVWA logon page, the relevant logon window is displayed.

To Authenticate to the Mobile PVWA

1. On your mobile device, access the PVWA URL; the Mobile PVWA logon page appears.
2. Specify your username and password, then click **Login**; the PVWA logs you onto the Vault and displays the Search page.



The Mobile PVWA Portal
Logon page



The Mobile PVWA Portal
Search page

Finding Accounts

1. In the Search Accounts page, in the Search field, specify a keyword to search for. You can specify up to four keywords.

Specify focused search criteria to optimize the search, resulting in quick and accurate results.

You can carry out a search for all the accounts in the Vault that you have access to by leaving the Search field empty. However, this might take a while as the process searches the entire Vault.

2. Click **Search**; the search is carried out in all the Safes in the Vault that you have access to. All accounts that meet the specified criteria are displayed in the Accounts Results list.



This page displays a summary of each account to enable you to identify the account you wish to view.

At the top of the list of accounts, a message displays the number of accounts that are displayed and the number of accounts that meet the search criteria.

If the PWWA has been configured to display a maximum number of accounts, and the search resulted in more accounts than this number, the message will display the number of accounts that the PWWA is configured to show as well as the total number of accounts found in the search.

Viewing Passwords

When you identify the account that contains the password you require, you can view the password.

1. In the Accounts Results list, click the name of the account to view; the account details page is displayed.
2. Click **Show** to display the password stored in this account.



A phonetic listing is displayed under the password. This makes it easier for you to identify the characters in the password, despite their small size on the mobile device.

The password is displayed for a predetermined number of seconds, after which the Account Details page is displayed again.

Changing Passwords

When you are logged onto the Mobile PVWA, you can change your user account password.

1. In almost any of the Mobile PVWA pages, click **Change Password**.

The Change Password page appears.



2. In the **Current Password** edit box, specify your current user account password.
3. In the **New Password** edit box, specify your new password. Make sure that the password you specify meets your enterprise password criteria.
This is the password you will use to log onto the PWWA next time you access it, whether from a mobile interface or in the full PWWA.
4. In the **Confirm Password** edit box, specify your new password again, then click **OK**; the PWWA changes your password.

Creating Requests

Before a user can retrieve an account from a Safe that requires confirmation, a request must be sent to all authorized Safe members and must be confirmed. Users can create request to access this type of accounts in the mobile PWWA.

1. In the Account Details page, click **Show**; the Get Password page appears and enables the user to specify the access information that is required to create a request.
2. Specify the reason for accessing the account.
3. If you require access during a period of time, click **Access timeframe**; the access timeframe details are displayed.
4. Specify the dates and times during which you will require access to the account.
5. If you will need to access the Safe or account several times during the specified timeframe, select **Multiple access required**.
6. Click **OK**; the PWWA creates a request for access to this account and sends a notification to Safe owners who are authorized to confirm requests.

You can see the status of the request at the bottom of the account properties. The following example shows that the request has been created and sent to authorized users and is waiting for confirmation.



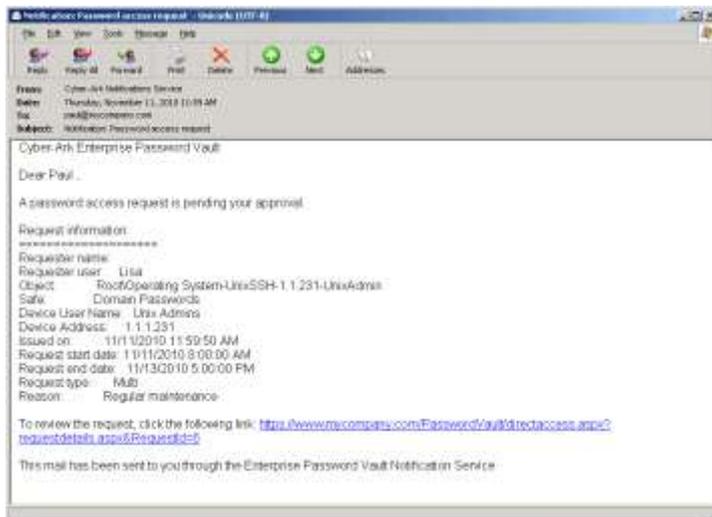
If a user tries to access the same account or file again before receiving confirmation, the Request Details page appears. A second request is not sent as the previous request is still unanswered.

Confirming Requests

Specified Safe Owners can authorize requests to permit other users to access an account. These requests can either be confirmed in the mobile PVWA or the full version of the PVWA.

The instructions below are for Safe members who have this authorization.

1. After a request has been created in the Mobile PVWA, an email notification is sent to Safe owners who can authorize this request.



This request is also displayed in the Incoming Requests page in the full version of the PVWA and can also be confirmed there, as described in *Confirming Requests*, page 260.

2. Use the link in the notification to access the Request Details page.



This page displays the details of the request as well as properties of the account that the request is for.

3. Scroll down the Request Details page to see all the information that is displayed.
4. After reading the request and all the account details, at the bottom of the page in the Authorize Access area, specify the reason for authorizing or rejecting the request.

5. Click **Confirm** to confirm the request,

or,

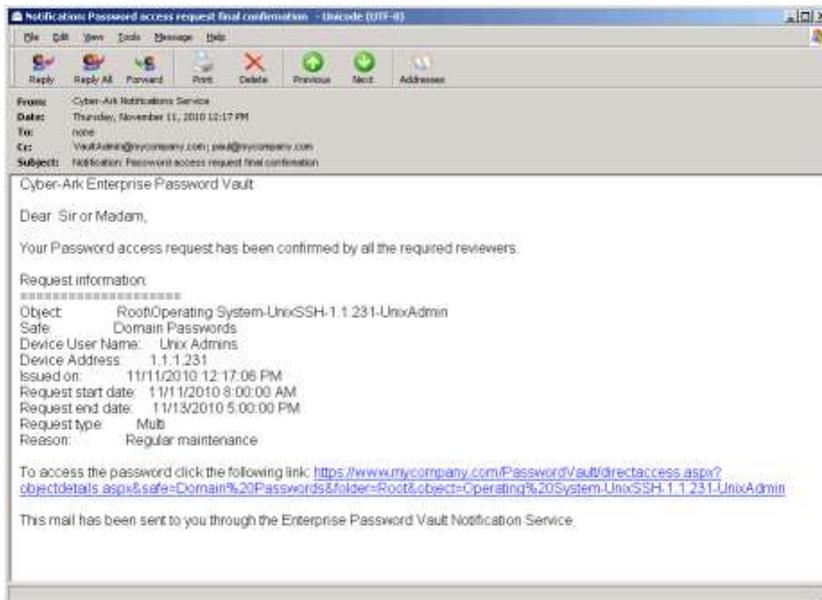
Click **Reject** to reject the request and prevent the user who created the request from accessing the account or file.

The PVWA sends an email notification to the user who created the request with details of the request confirmation or rejection, and displays the Search page again.

Receiving Confirmation

As soon as your request has been handled by an authorized user, an email notification is sent to the user who created the request. You can use this email to access the account that you now have confirmation to access and display the password.

1. After your request has been confirmed or rejected by an authorized user, you will receive an email notification.



2. Use the link in the notification to access the Account Details page. You can see in the Account Details Status that the request has been confirmed and you can now use the password.
3. Click **Show** to view the password stored in the account.

If the confirmed request is for a single operation, after you have used it to access an account, the request becomes invalid.

For more information about creating and using requests in the PVWA, refer to *Dual Control*, page 251.

Accessing the PVWA Mobile Version

Hand-held devices that are not automatically directed to the Mobile PVWA can access the Mobile version from the main PVWA logon page.

- In the main PVWA logon page, click **Mobile version**; the PVWA is directed to the Mobile PVWA and you can use all its features as described above.



Or,

- In your browser, specify the following URL:
`http://<host name>/passwordvault/m`

On-Demand Privileges for UNIX Environments

Users can perform highly sensitive UNIX tasks from their own UNIX accounts using on-demand privileges which elevate them to root or any other privileged account, according to preconfigured permissions and commands in the Vault. This enables users to perform a wide range of tasks, some of which are restricted, seamlessly and with optimum productivity.

The On-Demand Privileges Manager (OPM) uses the Vault technology to allow end users to perform super-user tasks with their own personal account, whilst maintaining the least-privilege concept. The entire procedure benefits from all the security and tracking features of the CyberArk Vault.

In addition to CyberArk's typical privileged account management features, each privileged session is recorded in text and/or video format and uploaded to the Vault where it can be viewed at any time by authorized users. For more information about viewing recorded sessions, refer to *Monitoring Privileged Session Recordings*, page 316.

Using the On-Demand Privileges Manager

Users can perform privileged tasks from their own UNIX account with a privileged user by using the 'pimsu' utility. This utility can be run by users who have been defined in the Vault together with the privileged task that they can perform.

Users can execute a privilege command using the 'pimsu' utility in which they specify the privileged command. The utility authenticates the end-user to the Vault and executes this command for him, according to the user's permissions and the command that has been defined in the Vault.

In order to run the 'pimsu' utility, users must be a member of the Safe where the privileged account is stored and must have the following permission in the Safe or for the specific account:

- Use accounts

In addition, the On-Demand Privileges Manager must be enabled in the Vault for the platform that manages the account, and the user must be allowed to issue it. For more information, refer to *Defining Privileged Commands*, page 748.

For example, to execute the 'kill' command users will issue the following pimsu command:

```
pimsu kill
```

By default, the current logged-on OS user name is used for Vault authentication. This user name can be overwritten by the `-vu` command line parameter, as shown in the following example. A user called John is currently logged onto a UNIX machine. If Fred issues the command below, the OPM will authenticate a Vault user, called Fred to the Vault.

```
pimsu -vu Fred kill
```

Authentication is either user/password based authentication, or Single Sign-On (SSO) based that relies on the underlying OS authentication. The required authentication methods can be configured in the platform definition.

Specifying pimsu without a command

Pimsu can be specified without a command, as shown in the following example:

```
pimsu
```

When pimsu is specified without a command, if the user is entitled to run this shell, their default shell will be opened as an elevated shell.

The 'pimsu' utility resides in the `/opt/CARKaim/bin` folder. To avoid the need to specify the full path, update the PATH environment variable or copy it to a directory that is already included in the PATH environment variable.

The pimsu utility has the following usage:

```
pimsu [-u <privilegedaccountname>]
      [-vu <vaultaccountname>]
      [-p <vaultpassword>]
      [-port <port>]
      [-t <timeout>]
      [-E]
      [-l <logsfolder>]
      [-d <debuglevel>]
      [command]
      [arg(s)]
      [-h]
```

The pimsu utility has the following syntax:

Note: All the following parameters are optional.

Option	Specifies	Default	Acceptable values
-u <privilegedaccountname>	The name of the privileged account that will be used to execute the privileged command.	root	String up to 255 characters
-vu <vaultaccountname>	The name of the user who will authenticate to the Vault and whose permission will be used. If this option is specified, the Vault password must also be specified. When this option is not specified, the OPM uses the OS user name to authenticate to the Vault and applies this user's permissions.	OS username	Vault username up to 128 characters
-p <vaultpassword>	The password required to authenticate to the Vault. If this option is not specified and the Vault password is required, the user will be prompted for it.	None	Vault password up to 170 characters

Option	Specifies	Default	Acceptable values
-port <port>	The number of the port of the OPM on the local machine.	18924	Valid port between 1 and 65535)
-t <timeout>	The maximum number of seconds that the pimsu utility will wait for a response during each request to the OPM.	40	Numeric up to 10 characters
-E <preserve environment>	Overrides the ResetEnvironmentVariables command restriction, and prevents it from taking effect. This parameter can only be used if the SetEnvironmentVariables and KeepEnvironmentVariables restrictions are enabled. For more information, refer to <i>Restrictions</i> , page 757.	False	True/False
-l	Displays the list of all the ACLs of the current user when he uses the privileged user. <ul style="list-style-type: none"> ▪ Specify -l -u to display ACLs for a specific privileged Vault user. ▪ Specify -l -vu to display ACLs for another Vault user. 		String
-sr	Displays the user's default restrictions.		
-lt <logsfolder>	The folder where the log file will be stored. The name of the log file will be always according to the unique ID (UUID) of the session.	.	UNIX folder path
-d <debuglevel>	The level of debug to write.		1
<command>	The command to execute on behalf of the privileged user, according to the user's permissions in the Vault. If specified, this parameter and the arg(s) parameter must be specified at the end of the command.	Default shell specified for the end user	Any valid UNIX command

Option	Specifies	Default	Acceptable values
arg(s)	Command argument(s). If specified, this parameter and the <command> parameter must be specified at the end of the command.	None	
-h	Displays help for the pimsu utility.		

Playing Session Recordings

You can play session recordings of privileged sessions in one of the following ways:

- **PVWA** – You can view details of your session recordings in the PVWA, and play them either from the beginning or from a specific command. For more information, refer to *Monitoring Privileged Session Recordings*, page 316.
- **OPMPlayer utility** – You can play session recordings using the OPMPlayer utility. You can also dump session recordings into text files that you can later view in an editor, and remove all control characters from the recording. For more information, refer to *Playing Session Recordings on Unix*, below.

Playing Session Recordings on UNIX

The OPMPlayer utility enables you to view OPM sessions as they appeared on the screen, making it easier to review recordings and search for any commands included in them.

During installation, the OPMPlayer utility is copied to the OPM installation folder.

This utility has the following syntax:

```
OPMPlayer <recording file> [-skipdelays] [-showoutput]
```

Command	Specifies
-skipdelays	Whether or not the OPMPlayer will add idle time in the playback. This parameter enables you to replay recordings while skipping idle time.
-showoutput	Whether or not the OPMPlayer will display the output for commands that were run with stdout and/or stderr redirection. Specify one of the following valid values: <ul style="list-style-type: none"> ▪ True – The OPMPlayer will print stdout/stderr data to the terminal. This is the default value. ▪ False – The OPMPlayer will not display the output for commands that were run with stdout and/or stderr redirection.

To Replay an OPM Session

1. Save a session recording on your UNIX environment:
 - i. In the MONITORING page, download a recording to your local machine.
 - ii. If you need to transfer the recording file from a different machine to your Unix machine, make sure it is transferred in binary mode.
2. Use the OPMPPlayer utility to playback the recording.
3. Press **Enter** to skip idle parts in the playback.

To Clean Up an OPM Session to View it in an Editor

1. Save a session recording on your UNIX environment:
 - i. In the MONITORING page, download a recording to your local machine.
 - ii. If you need to transfer the recording file from a different machine to your Unix machine, make sure it is transferred in binary mode.
2. Run the CleanOutput script.

This script has the following syntax:

```
./cleanoutput.sh <recording_file> <out_file>
```

Command	Specifies
recording_file	The full pathname of the recording file downloaded from the PVWA and saved in binary format on the Linux machine.
out_file	The file where the recording is stored as a text file without the control characters from the recording.

3. View the clean recording text file.

Monitoring Privileged Sessions

Privileged Session Manager (PSM) enables organizations to secure, control and monitor privileged access to network devices by using the Vault technology to manage privileged accounts and create detailed session audits and video recordings of all IT administrator privileged sessions on remote machines.

The PSM Suite also includes the PSM SSH Proxy (PSMP), which preserves the benefits of PSM such as isolation, control and monitoring, whilst enabling users to connect transparently to target UNIX systems from their own workstation without interrupting their native workflow.

PSM provides the following features:

- **Recorded Privileged Sessions** – All the activities in each privileged session can be recorded in text and/or video format, and stored in the Vault, compressed, for future auditing. These recordings are transparent to users and cannot be bypassed.
- **SQL Command Level Audit** – All the command activities carried out in SQL privileged sessions can be recorded and stored in the Vault as audit records, and viewed at any time by authorized users. Session recordings can also be searched according to the SQL commands that were invoked during sessions.

Note: This is only applicable to the PSM.

- **SSH Keystrokes Audit** – All the keystrokes that are typed during SSH privileged sessions can be recorded and stored in the Vault as audit records, and viewed at any time by authorized users. Users can search session recordings for specific content according to these SSH keystrokes.
- **Universal Keystrokes Audit** – All the keystrokes that are typed during privileged sessions, such as on Mainframe, SQL clients, Windows and all other supported platforms can be recorded and stored in the Vault as audit records, and viewed at any time by authorized users.

In addition, keystrokes that are typed during privileged sessions that are initiated by the Universal Connector are also recorded. For more information, refer to *Configuring a PSM Universal Connector Connection Component*, page 668.

Users can search session recordings for specific content according to these keystrokes.

- **Windows Events Audit** – Titles of windows that were accessed during a privileged session are recorded and stored in the Vault as audit records. Authorized users can view these audit records at any time and can search for session recordings in which a particular window title was accessed. Users can also search recordings according to the name of the process that the audited windows titles belong to.
- **Privileged Remote Access** – Users can initiate privileged sessions to the PSM proxy machine using HTTPS protocol. This meets standards for secure remote access by ensuring encrypted sessions and by not requiring the corporate firewall to be opened to additional native protocols.
- **Privileged Single Sign-On** – Users connect transparently to remote target applications and systems through the PSM proxy machine via the PVWA.
- **Centralized Management** – In the PVWA, users can see all the recordings archives, where auditors can retrieve and view comprehensive recordings of privileged sessions. Search features enable auditors to locate specific recordings.

- **Transparent Integration** – The PSM can be integrated transparently and seamlessly into existing enterprise infrastructures, including a variety of authentication, monitoring, ticketing, and workflow systems.
- **Monitoring and Terminating Live Sessions** – Authorized users can monitor live privileged sessions in real-time, viewing them or taking part in controlling them according to predefined configurations. This enables authorized users to supervise live sessions and also enables two users to perform a procedure concurrently. In addition, authorized users can terminate suspicious sessions immediately, when necessary.

Note: This is only applicable to the PSM.

PSM can be used by the following organizational roles:

- **IT administrators** who need to perform administrative tasks on remote network devices and managed hosts. Users can transparently log onto remote devices directly without disturbing their workflow by needing to retrieve and copy the passwords. The entire privileged session can be recorded automatically without any human intervention.
- **Auditors and security officers** who require access to audit information and privileged session recordings. These users benefit from centralized administration that is displayed in a simple, intuitive interface.
- **Administrators** who needs to configure, manage and administer PSM related activities.

Monitoring Privileged Session Recordings

The PWA acts as a centralized access point for privileged session recordings. In order to display information about privileged session recordings and be able to play the session recordings, users require the following authorizations:

- Membership in the **Auditors** group

Or,

- Membership in the relevant Password Safes and Recording Safes with the following authorizations:
 - In the relevant account Safes:
 - List accounts/files

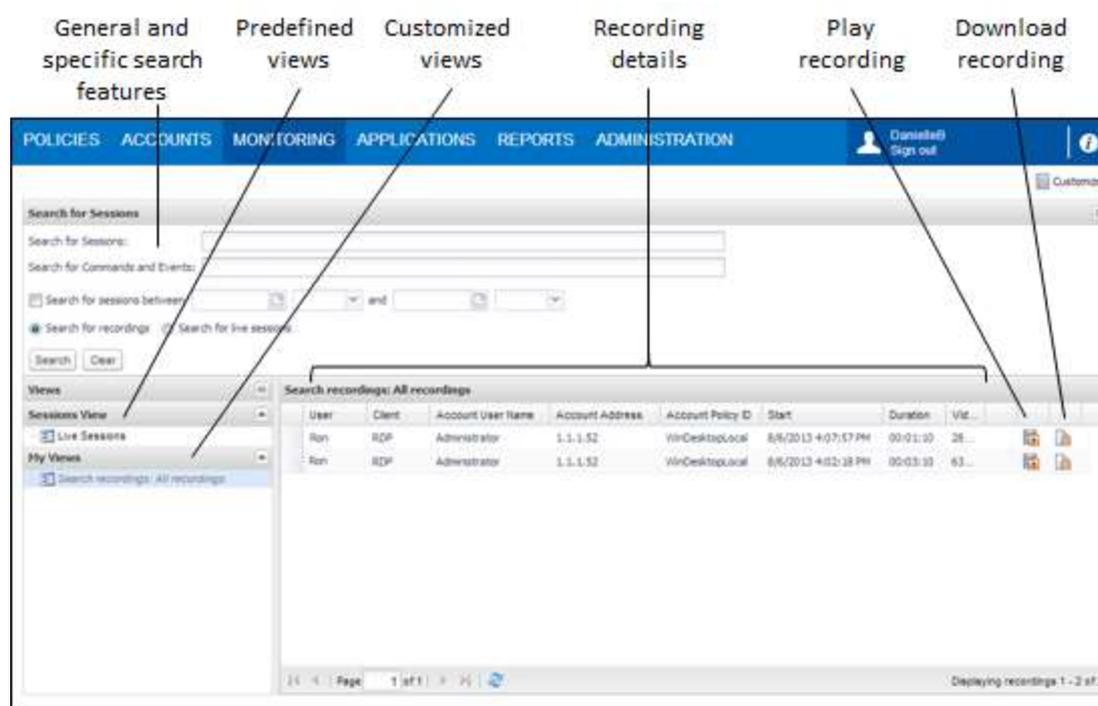
Note: This authorization specifically enables users to access recordings from the Account Details page.
 - In the relevant recording Safes:
 - Retrieve accounts/files
 - List accounts/files
 - View audit

Authorized users can view the recordings in any of the following ways:

- The **MONITORING** page enables intuitive access to all privileged session recordings. This page is visible to authorized users after the first recording has been uploaded to the Vault.
- The **Recording Details** page enables a more thorough view of a specific session recording.
- The **Account Details** page provides access to recordings for individual passwords.

Privileged Session Recordings

The Recording page enables authorized users to search for and access privileged session recordings in a centralized point.



- **Displaying privileged session activity** – Authorized users can search for video and text recordings according to session information, such as address or username, or by a command or event that was performed during the session and is stored in the recording. For more information, refer to *Finding Session Recordings*, page 318.
- **Customize views** – You can create a set of customized views that display a list of recordings in one quick step, increasing accessibility and efficiency. You can save these personalized views and even mark one so that it is displayed as the default view the next time you log on and display the MONITORING page. For more information about customizing views, refer to *Customizing your own Views*, page 337.

- **Access recordings** – You can access video and text recordings of privileged sessions, view their details and their contents. You can also see which other users are authorized to access these recordings and any activities that they performed on them, as well as detailed information and properties of the recording file. For more information, refer to *Displaying Session Recording Details*, page 322.
- **View commands and events** – You can view a list of the commands and events that were issued during specific privileged sessions. This enables you to audit every keystroke or command, facilitating total accountability. For more information about auditing commands and events, refer to *Displaying Session Recording Details*, page 322.
- **Playing Privileged Session Recordings** – By configuring the PSM for direct playback, you can play privileged session video recordings directly from the PVWA using an embedded video player. Alternatively, you can open or download video recordings and view them using your default media player. Likewise, you can either open text recordings and view them immediately or download them and view them at your convenience. For more information, refer to *Accessing Privileged Session Recordings*, page 322.

Finding Session Recordings

The MONITORING page displays the following recordings:

- **Video Recordings** – Video recordings of privileged PSM, PSMP and OPM sessions.
- **Text Based Recordings** – Text recordings of privileged PSM, PSMP and OPM sessions.

You can search for these recordings using a free text search according to the properties that are associated with the privileged session (e.g. password, user, address, device, machine, ticket ID, or any other account keyword). You can also search for recordings according to SQL commands, SSH or SCP (Secure Copy) commands, keystrokes typed during sessions on any platform, Windows events that were recorded during sessions.

You can limit search results according to dates, which adds an extra dimension to the tracking facility and enables a quick search and full audit of all password activity according to keywords over a specific period of time.

After each search, a definition of the search is listed in the Views list, enabling you to access the results of different searches without the need to repeat them. For more information about customized views, refer to *Customizing your own Views*, page 337.

You can change the columns that are displayed in the recordings list to display different properties of the displayed recordings and reorganize the displayed list recordings so that you can locate recordings quickly and easily. For more information, refer to *Managing the Recordings List*, page 338.

Note: The recordings are stored in the Safe in a compressed format. The size of the recording that is displayed in the Recordings list indicates the size of the compressed recording file, and not its actual size.

To Find Privileged Session Recordings

1. Display the MONITORING page; the Monitoring – Sessions List page appears.
2. In the search edit boxes, specify keywords that will be used in the search. These keywords may include the following:
 - **Search for Sessions** – Any information about the privileged session that was recorded. This includes the name of the user or the privileged account user, the name or IP address of the remote machine, the platform name, port or database name.
 - **Search for Commands and Events** – You can search for specific events that were executed during PSM sessions. Events and commands that are issued during sessions can be recorded in OPM commands and PSM connections.

Note: Specify all or part of a search word. Do not specify wildcards.

For example, you can specify a single command to display recordings of privileged sessions during which that command was issued, or a command and an IP address to display recordings of all the privileged sessions run from a particular IP address during which that command was issued.

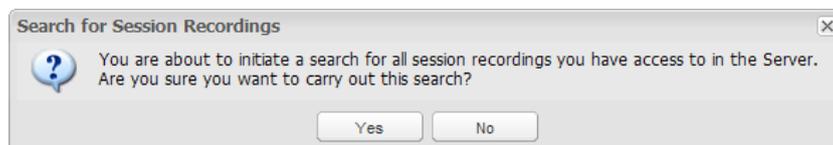
Leave the search edit boxes empty to display all the recordings you are authorized to view.

3. To specify a timeframe for the search, select **Search for session recordings between**; the date and time drop-down boxes are enabled.

Leave this checkbox clear to check all dates and times.

4. Select a date and time to begin and end the search.
5. Click **Search** or press **Enter**.

If you did not specify any search criteria, the following message will appear:



6. Click **Yes** to begin the search,
or,

Click **No** to return to the Search for Sessions page where you can specify search criteria.

The Search is performed in the Safes where the recordings that you are authorized to access are stored, and all the Session Recordings that meet the search criteria are displayed.

7. To view recordings:

- **Video Recordings** – In the search results, click one of the following icons:

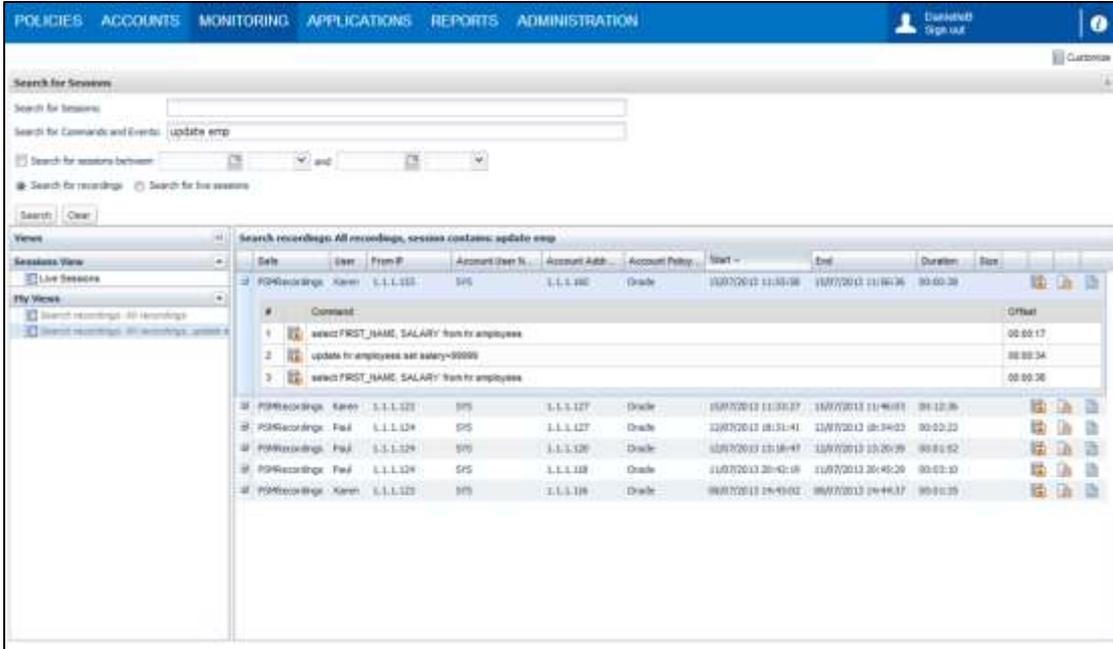
Icon	Name	Description
	Play recording	Plays the session recording immediately in a direct playback. You can either play the recording from the beginning or from a specific command. This icon is displayed when Direct Playback is enabled.
	Download recording	Enables you to open the session recording or save it in a different location.

- **Text Based Recordings** – In the search results, click the following icon:

Icon	Name	Description
	Save text recording	Enables you to open the session text recording or save it in a different location. You can view commands and events that were executed during a PSM or OPM session.

8. To display the **Recording Details** page, and view the contents of the **Events** tab, click the recording line in the search results.

The following example shows the results of a search in recordings for two words, **update** and **emp**, that were issued during recorded sessions. These words are not connected by surrounding quotation marks.



The screenshot shows the Oracle Enterprise Manager interface with the search results for recordings. The search criteria are "update emp". The results table shows several recordings with a preview of commands containing "update" and "emp".

Views	Search recordings: All recordings, session contains: update emp												
<ul style="list-style-type: none"> Session View Live Sessions My Views Search recordings: All recordings Search recordings: All recordings, session c... 	<table border="1"> <thead> <tr> <th>#</th> <th>Command</th> <th>Offset</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>select FIRST_NAME, SALARY from hr.employees</td> <td>00:00:17</td> </tr> <tr> <td>2</td> <td>update hr.employees set salary=99999</td> <td>00:00:24</td> </tr> <tr> <td>3</td> <td>select FIRST_NAME, SALARY from hr.employees</td> <td>00:00:26</td> </tr> </tbody> </table>	#	Command	Offset	1	select FIRST_NAME, SALARY from hr.employees	00:00:17	2	update hr.employees set salary=99999	00:00:24	3	select FIRST_NAME, SALARY from hr.employees	00:00:26
#	Command	Offset											
1	select FIRST_NAME, SALARY from hr.employees	00:00:17											
2	update hr.employees set salary=99999	00:00:24											
3	select FIRST_NAME, SALARY from hr.employees	00:00:26											

This search returns all the recordings that include at least one command that contains both words, although not necessarily consecutively. The commands preview displays several commands that were issued during the session that contain at least one of these words, in this case, **update** or **emp**. Commands that do not contain either of the words specified in this search are not included in the preview.

Commands or events that are surrounded by quotation marks, for example “**update emp**” will initiate a search for recordings of sessions during which these two words were issued consecutively, separated by one space, at least once.

The following example shows the results of a search for recordings in which windows with a title that included the words “**new**” or “**user**” were displayed.

The screenshot shows the 'Monitoring - Sessions List' interface. The search criteria are set to 'Search for recordings' and 'Search for Commands and Events: new user'. The search results table shows two sessions:

User	Client	Account User Name	Account Address	Account Policy	Start	Duration	View Size
administrator	RDP	adm	1.1.1.119	WinServerLocal	2016/01/14 13:41:04	00:00:00	1729
administrator	RDP	adm	1.1.1.119	WorkServerLocal	2016/01/14 13:38:02	00:04:02	9448

A preview window for the second session shows a 'Command' list with one entry: 'new user: New User'.

This search returns all the recordings in which a window with a title that included the words “**new**” or “**user**” was displayed at least once during the session. Each search result includes a preview that displays all window titles displayed in this recording that match the search criteria.

The following example shows the results of a search for recordings in which the **scp** command was issued to copy files securely through PSMP.

The screenshot shows the 'Monitoring - Sessions List' interface. The search criteria are set to 'Search for recordings' and 'Search for Commands and Events: scp'. The search results table shows multiple sessions:

User	Client	Account User Name	Account Address	Account Policy	Start	Duration	View Size
root	PSMP-SCP	root	10.10.8.228	UnixSSH	2/8/2016 3:11:57 PM	00:00:07	
root	PSMP-SCP	root	10.10.8.228	UnixSSH	2/8/2016 3:11:00 PM	00:00:01	

A preview window for the second session shows a 'Command' list with one entry: 'scp to [tmp] on 10.10.8.228'.

This search returns all the recordings of sessions in which the **scp** command was used.

Accessing Privileged Session Recordings

Authorized auditors can view the privileged session recordings to see exactly what happened during each session. Users can play recordings directly from the PWWA or download them and play them using a media player.

Recordings can be played or viewed in any of the following pages:

- Monitoring – Sessions List
- Recording Details page
- Account Details page

Users must have the **View Audit** authorization in the Safe where the recordings are saved or they must belong to the **Auditors** group.

For more information about viewing OPM sessions, refer to *Viewing Text Recordings*, page 330.

Displaying Session Recording Details

The Recording details page enables you to see details about privileged session recordings, including details about the account that was used, a list of all the events that took place during the recorded session, an attestation list of activities performed on the recording, and a list of users who are authorized to access the recording.

To Display Session Recording Details

1. In the list of Session Recordings, click a specific recording; the Recording Details page appears.

This page displays all the details about the recording, including the following:

- **Events** – A list of commands and keystrokes that were performed during the privileged session and the time from the beginning of the session that they were carried out. You can also play a recording from the point of a specific event.
- **Attestation** – Activities that were carried out on the recording files.
- **Permissions** – Users who have permission to access the recording files through object level access. For more information, refer to *Object Level Access Control*, page 76.
- **Advanced** – Detailed information and properties of the recording file, including the compressed size and the actual size of the recording files in the Vault.

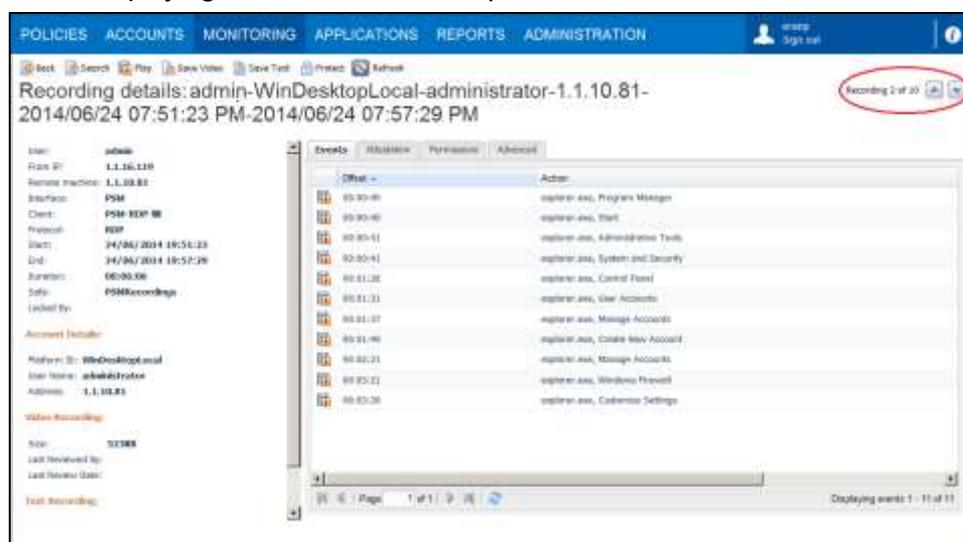
In addition to viewing all the information about the recording, you can do the following:

- **Play recording** – If this is a video recording, you can play the recording immediately. This option is available when Direct Playback is enabled. For more information about playing video recordings, refer to *Playing Video Recordings*, page 324.
- **Download recording** – If this is a video recording, you can download it and save it in a different location. For more information about playing video recordings, refer to *Downloading Video Recordings*, page 329.
- **Save text recording** – If this is a text recording, you can save it and view the contents of the recording. For more information about viewing text recordings, refer to *Viewing Text Recordings*, page 330.

- **Protect or Unprotect the recording** – You can protect important recordings from being deleted automatically after the Safe retention period on the Recordings Safe has expired.
 - To protect a recording, click **Protect** on the toolbar; the recording will be stored in the Safe either until you delete it or until you remove the protection.
 - To unprotect a recording, click **Unprotect** on the toolbar; the recording will be deleted from the Safe the next time that expired Safe history is erased from the Safe.

The retention period and the automatic purge settings can be modified in the Safe properties.

- **Browse between search results** – You can easily browse other recordings found during the same search to review their content and recorded commands/events without having to return to the Search results page each time, simplifying the auditor's review process.



Displaying Recordings for Individual Accounts

In the Account Details page for accounts whose platform is configured to use PSM, PSMP or OPM session recording, users can see video and text recordings of every privileged session during which a specific account was used. This provides a complete audit of individual accounts, what they were used for, and on which machine.

Users must have the **View Audit** authorization in the Password Safe or they must belong to the **Auditors** group.

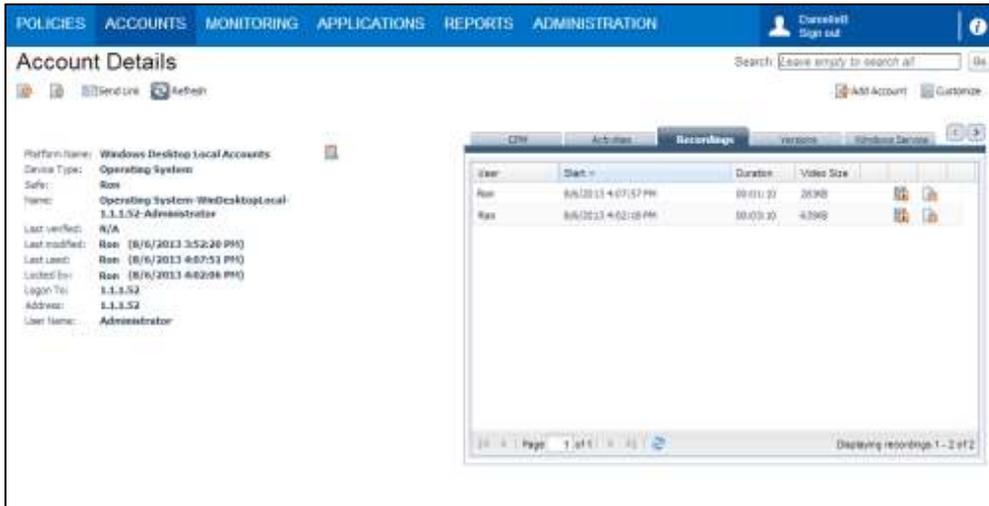
To Display Account-Related Recordings

1. In the Accounts list, click the account whose recording you want to view; the Account Details page appears.

- In the Account Details page, display the **Recordings** tab; all the recorded privileged sessions for this account appear.

The following icons in the Recordings tab indicate which recordings are available for this session recording:

Icon	Indicates
	A video recording
	A video recording
	A text recording



- You can view more information about a recording by clicking on the specific recording in the Recordings tab.

Playing Video Recordings

Authorized auditors can play privileged session recordings and see an exact replica of the tasks that were performed during a privileged session in a VCR-like playback. Session recordings are AVI files that, by default, are played with Windows Media Player, although they can be played with other media applications.

To Play Privileged Session Recordings

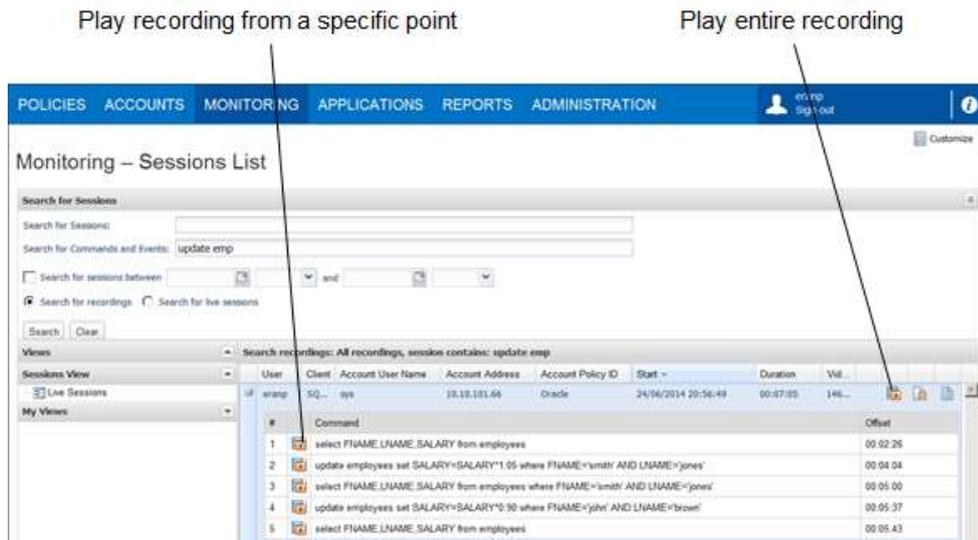
- Display the recording to play, then click the relevant button to start the **Direct Play**. This may be in any of the following pages:

In the Monitoring – Sessions List:

- In the details of the recording to view, click the **Play recording** icon to play the entire recording,

or,

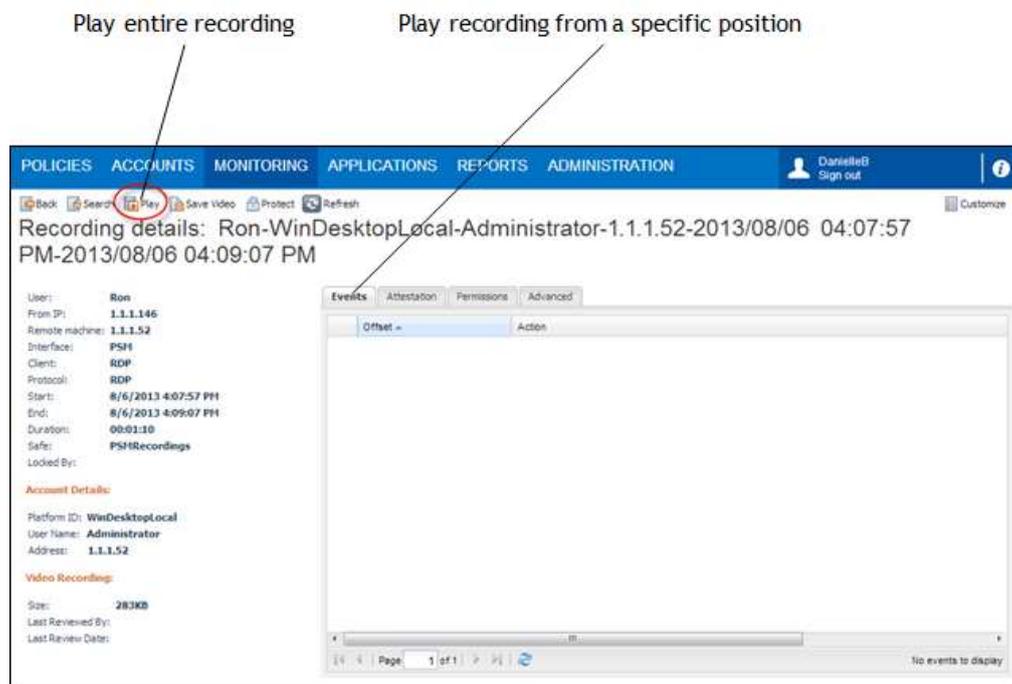
- Click **Start playback from this position** to play the recording from a specific point.



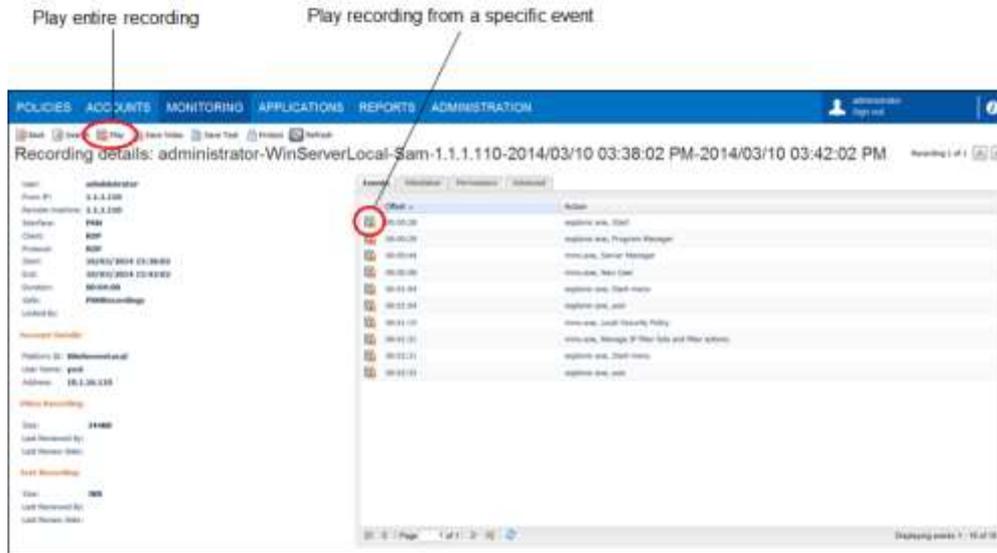
In the Recordings Details page:

- On the toolbar, click **Play** to play the entire recording,
- or,
- In the Events tab, click **Start playback from this position** to play the recording from a specific point.

The following example shows the Recording Details page of a privileged SQL session. You can start the recording from any specified command.

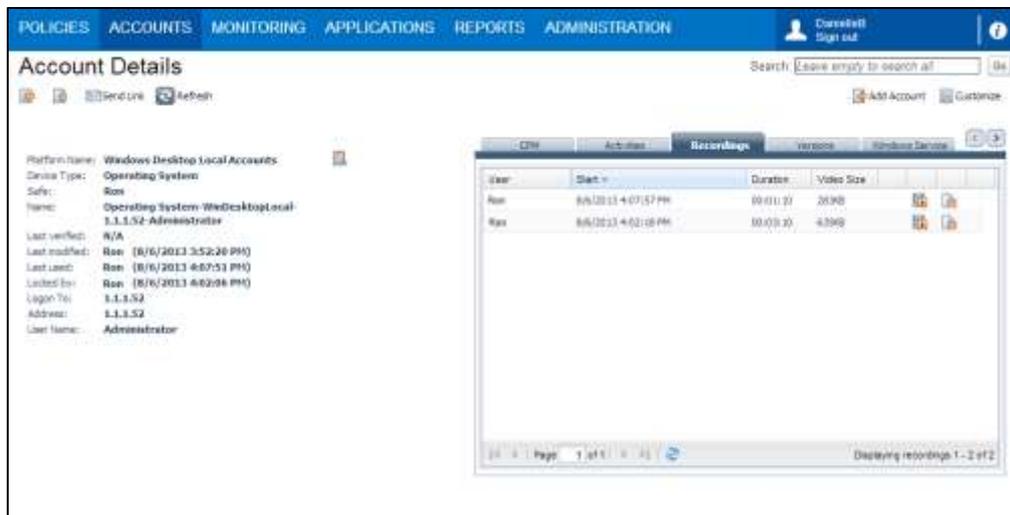


The following example shows the Recording Details page of a privileged Windows session. You can start the recording from any specified Windows event that has been recorded.

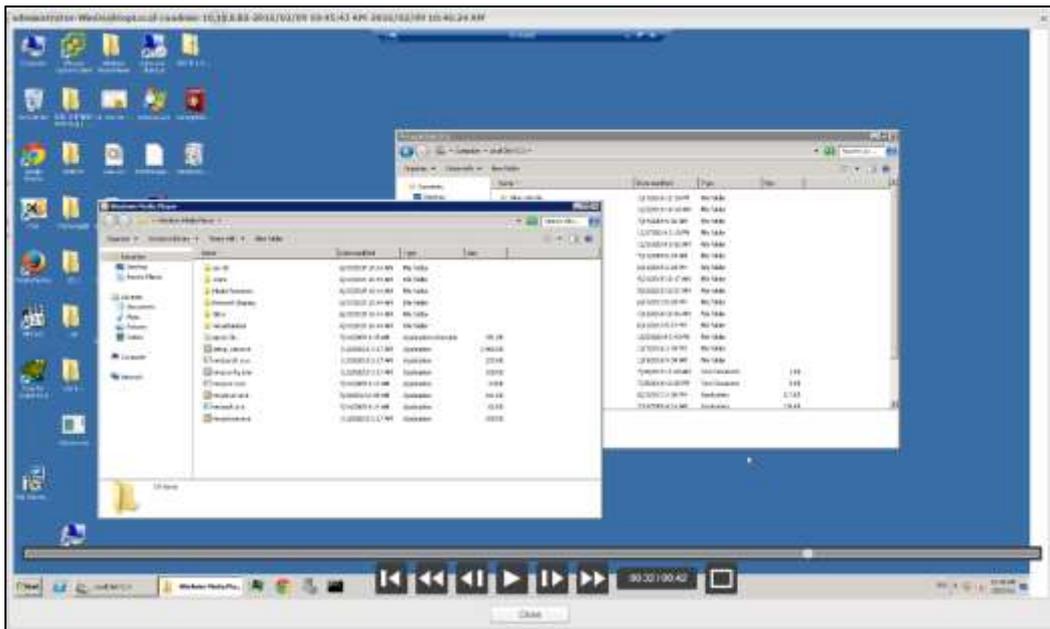


In the Account Details page:

- In the Recordings tab, in the details of the recording to view, click the **Play recording** icon.



2. When you click any of the Play buttons described above, the Direct Play window appears.



If you selected **Start playback from this position** in the Events tab of the Recording Details page, the recording will start playing from the selected action.

You can play and replay the recording in the same way as any media file. Use the play control buttons to replay different parts of the recording or to skip to a different point.

When a recording contains time when a user was not active, you can skip to the next action by clicking the **Next frame** button.



When you play PSMP or OPM session recordings, a Direct Play window appears, as shown in the following example:



To Play Additional Recordings with Direct Playback

After a direct playback has finished running, you can start another one without having to close the embedded video player.

1. After the direct playback has finished running, **do not** click **Close**.
2. Toggle to the PVWA and in the details of the next recording to play, click **Play**; the video player immediately begins playing the recording in the same window.

Click 'Play recording' to play the next privileged session recording

The screenshot shows the 'Monitoring - Sessions List' interface. A terminal window is open in the foreground, displaying a 'Welcome to CentOS' message. Below the terminal, a table lists recording sessions. An arrow points from the text above to the 'Play recording' button in the table row for the session starting on 09/06/2015 at 16:59:06.

ID	Start	Duration	File	Actions
	18/06/2015 10:50:17	00:01:38	207B	[Icons]
	17/06/2015 15:06:42	01:03:32	529B	[Icons]
	17/06/2015 15:04:55	00:00:56	60B	[Icons]
	15/06/2015 22:55:05	00:01:07	106B	[Icons]
	15/06/2015 21:52:45	00:00:50	60B	[Icons]
	15/06/2015 21:32:19	00:07:07	1.19	[Icons]
	15/06/2015 21:12:08	01:16:51	4.49	[Icons]
	15/06/2015 20:49:24	00:18:14	2.04	[Icons]
	15/06/2015 20:32:28	01:56:24	7.55	[Icons]
	15/06/2015 20:28:38	00:59:06	529B	[Icons]
	15/06/2015 20:08:52	00:03:34	272B	[Icons]
	15/06/2015 20:05:11	00:03:06	839B	[Icons]
	09/06/2015 16:59:06	00:00:48	36B	[Icons]
	09/06/2015 16:57:41	00:02:12	36B	[Icons]
	09/06/2015 16:56:41	00:03:17	17B	[Icons]
	09/06/2015 16:55:36	00:04:01	29B	[Icons]
	09/06/2015 16:55:20	00:04:41	30B	[Icons]
	09/06/2015 16:55:03	00:04:59	33B	[Icons]

You can also click **Start playback** from this position to play the recording from a specific point.

Downloading Video Recordings

Authorized auditors can download privileged PSM session recordings and view them according to their convenience.

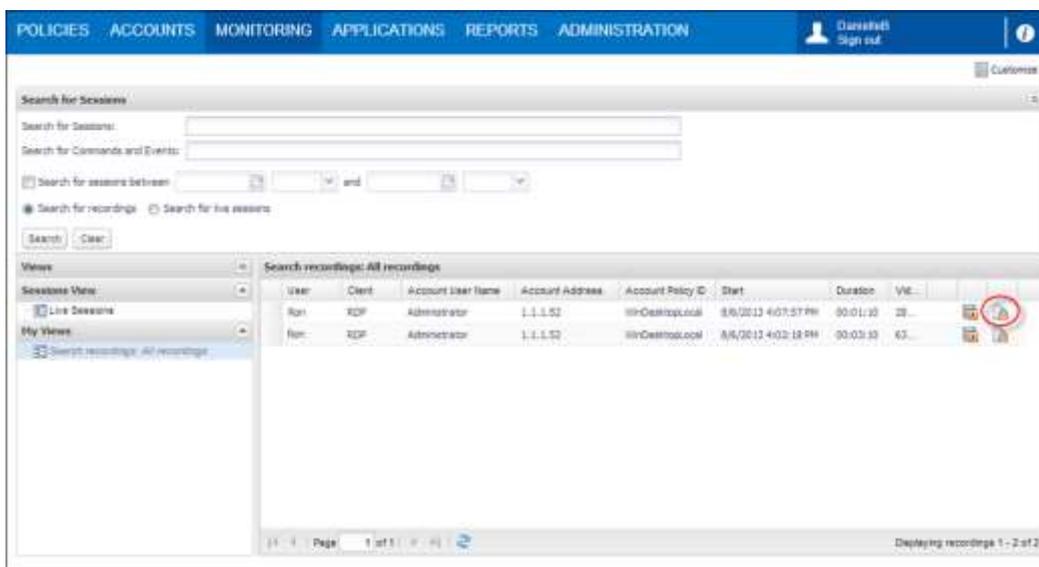
Notes:

- Make sure that the PSM codec for high compression session recordings is installed on your desktop. This codec is included in the in the PSM installation package, and enables you to download and play session recordings with a regular media player. Administrator permissions are required in order to install this codec.
- Currently, you can only download video recordings for PSM sessions, but not for PSMP or OPM sessions.

To Download Privileged Session Recordings

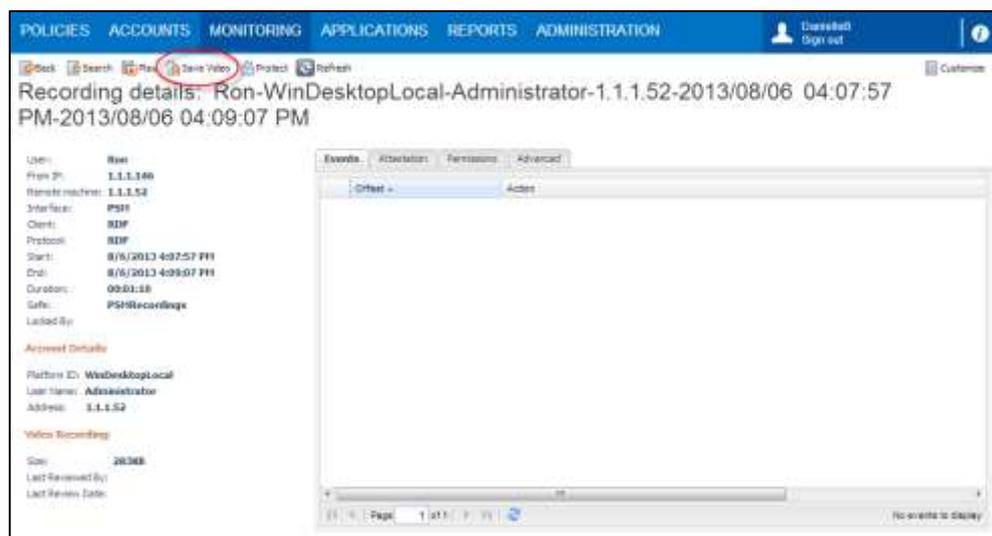
- Display the recording to download, then click the **Download recording** button.

In the Recordings List:



In the Recordings Details page:

- On the toolbar, click **Save Video**.



In the Account Details page:

- In the Recordings tab, in the details of the recording to view, click the **Download recording** icon.
2. The File Download window appears.
 - Click **Open** to play the recording,
 - or,
 - Click **Save** to save the recording in another location.
- Note:** If you save the recording in a location outside the Safe, it will not be secure and unauthorized users will be able to access it.
3. If you click open, the media player will begin to play the recorded session.

You can play and replay the recording in the same way as any media file.

Viewing Text Recordings

Authorized auditors can view privileged session text recordings and see all the commands that were executed during a privileged session. Auditors can view the following text recordings:

- **Privileged SSH sessions** – The entire session as textual lists of commands.
- **Privileged SQL commands** – A list of SQL commands issued in a privileged session.
- **Privileged Windows sessions** – A full textual log of the windows titles that were opened by the user during the session.

To View Text-based PSM Session Recordings

1. Display the text recording to view, then click **Save Text**. This may be in any of the following pages:

In the Recordings List:

- In the details of the recording to view, click the **Save text recording** icon.

In the Recordings Details page:

- On the toolbar, click **Save Text**.

2. The File Download window appears.

Click **Open** to view the recording,

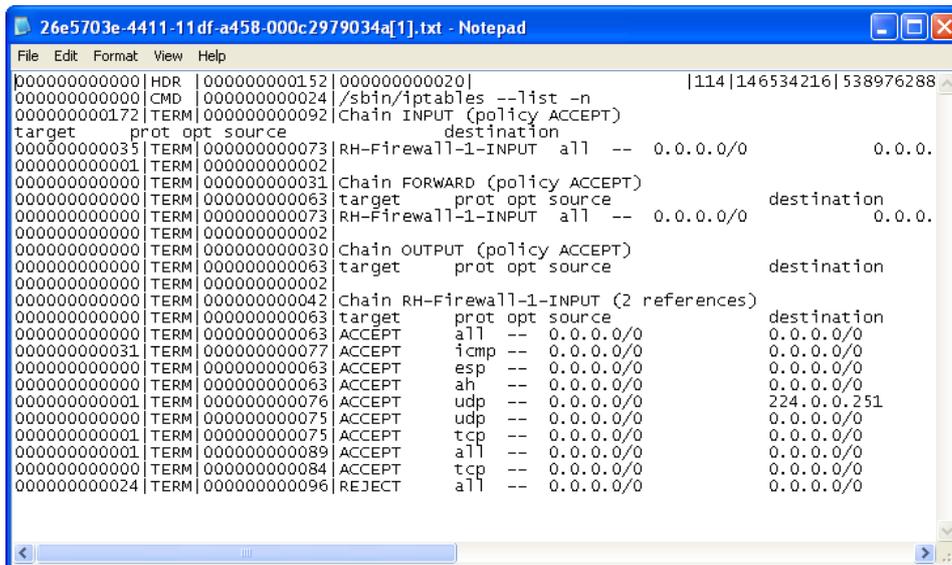
or,

Click **Save** to save the recording in another location.

Note: If you save the recording in a location outside the Safe, it will not be secure and unauthorized users will be able to access it.

3. If you click open, the recording is displayed as a text file.

The following example shows a text recording of a privileged SSH session:



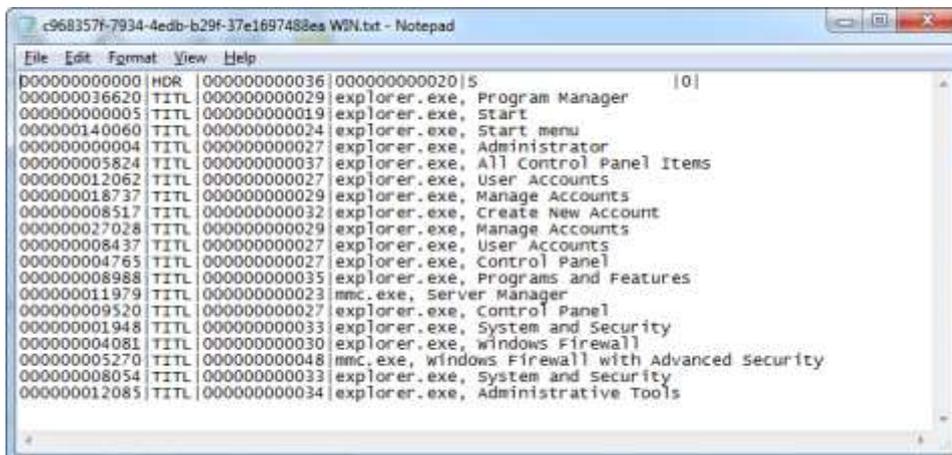
```

26e5703e-4411-11df-a458-000c2979034a[1].txt - Notepad
File Edit Format View Help
000000000000 HDR 000000000152 000000000020 |114|146534216|538976288
000000000000 CMD 000000000024 /sbin/iptables --list -n
000000000172 TERM 000000000092 Chain INPUT (policy ACCEPT)
target prot opt source destination
000000000035 TERM 000000000073 RH-Firewall-1-INPUT all -- 0.0.0.0/0 0.0.0.0/0
000000000001 TERM 000000000002
000000000000 TERM 000000000031 Chain FORWARD (policy ACCEPT)
target prot opt source destination
000000000000 TERM 000000000063 RH-Firewall-1-INPUT all -- 0.0.0.0/0 0.0.0.0/0
000000000000 TERM 000000000002
000000000000 TERM 000000000030 Chain OUTPUT (policy ACCEPT)
target prot opt source destination
000000000000 TERM 000000000063
000000000000 TERM 000000000002 Chain RH-Firewall-1-INPUT (2 references)
target prot opt source destination
000000000000 TERM 000000000063 ACCEPT all -- 0.0.0.0/0 0.0.0.0/0
000000000031 TERM 000000000077 ACCEPT icmp -- 0.0.0.0/0 0.0.0.0/0
000000000000 TERM 000000000063 ACCEPT esp -- 0.0.0.0/0 0.0.0.0/0
000000000000 TERM 000000000063 ACCEPT ah -- 0.0.0.0/0 0.0.0.0/0
000000000001 TERM 000000000076 ACCEPT udp -- 0.0.0.0/0 224.0.0.251
000000000000 TERM 000000000075 ACCEPT udp -- 0.0.0.0/0 0.0.0.0/0
000000000001 TERM 000000000075 ACCEPT tcp -- 0.0.0.0/0 0.0.0.0/0
000000000001 TERM 000000000089 ACCEPT all -- 0.0.0.0/0 0.0.0.0/0
000000000000 TERM 000000000084 ACCEPT tcp -- 0.0.0.0/0 0.0.0.0/0
000000000024 TERM 000000000096 REJECT all -- 0.0.0.0/0 0.0.0.0/0

```

This text file contains a record of the text that was generated during the privileged session. Depending on the type of session and the PSM connection, this file might contain commands that were issued and the channels that were used during the session.

The following example shows a textual log of a privileged Windows session.



```

c968357f-7934-4edb-b29f-37e1697488es WIN.txt - Notepad
File Edit Format View Help
000000000000 HDR 000000000036 000000000020 |S
000000036620 TITL 000000000029 explorer.exe, Program Manager |0|
000000000005 TITL 000000000019 explorer.exe, Start
000000140060 TITL 000000000034 explorer.exe, Start menu
000000000004 TITL 000000000027 explorer.exe, Administrator
000000005824 TITL 000000000037 explorer.exe, All Control Panel Items
00000012062 TITL 000000000027 explorer.exe, User Accounts
00000018737 TITL 000000000029 explorer.exe, Manage Accounts
000000008517 TITL 000000000032 explorer.exe, Create New Account
000000027028 TITL 000000000029 explorer.exe, Manage Accounts
000000008437 TITL 000000000027 explorer.exe, User Accounts
000000004765 TITL 000000000027 explorer.exe, Control Panel
000000008988 TITL 000000000035 explorer.exe, Programs and Features
000000011979 TITL 000000000023 mmc.exe, Server Manager
000000009520 TITL 000000000027 explorer.exe, Control Panel
000000001948 TITL 000000000033 explorer.exe, System and Security
000000004081 TITL 000000000030 explorer.exe, windows Firewall
000000005270 TITL 000000000048 mmc.exe, windows Firewall with Advanced Security
000000008054 TITL 000000000033 explorer.exe, System and Security
000000012085 TITL 000000000034 explorer.exe, Administrative Tools

```

This text file contains a full audit record of the processes that were run during the privileged session.

To View Text-based OPM Session Recordings

1. Display the text recording to view, then click **Save Text**. This may be in any of the following pages:

In the Recordings List:

- In the details of the recording to view, click the **Save text recording** icon to save the text in readable format.

In the Recordings Details page:

- On the toolbar, click **Save Text** to save the text in readable format.
- On the toolbar, click **Save Raw Text** to save all the text that appeared during the session, including keystrokes, control characters, terminal properties, etc.

2. The File Download window appears.

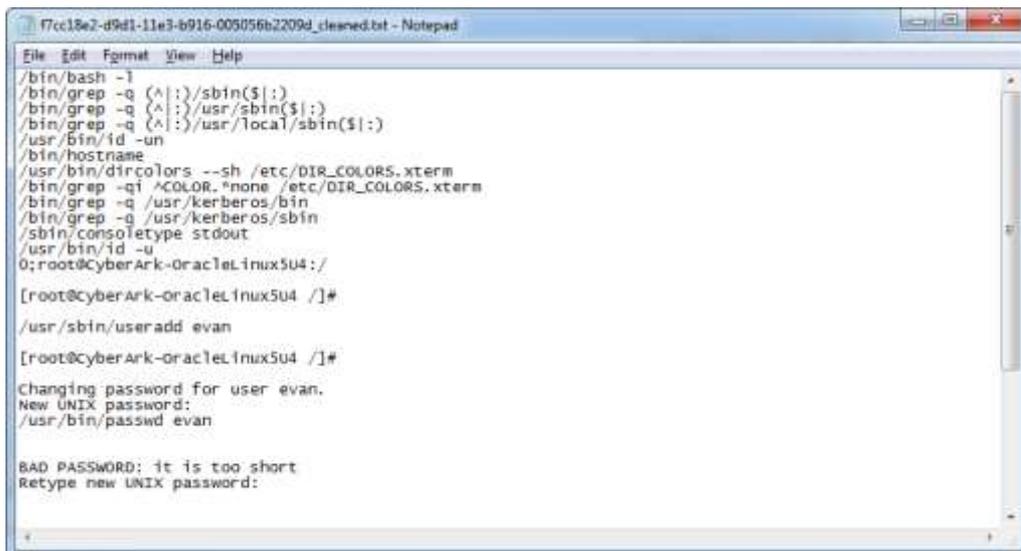
- Click **Open** to view the recording,
- or,
- Click **Save** to save the recording in another location.

Note: If you save the recording in a location outside the Safe, it will not be secure and unauthorized users will be able to access it.

3. If you click open, the recording is displayed as a text file.

The following examples show text recordings of OPM sessions:

Example 1: Viewing a readable text recording:



```

f7cc18e2-d9d1-11e3-b916-005056b2209d_cleaned.txt - Notepad
File Edit Format View Help
/bin/bash -l
/bin/grep -q (^:)/sbin($:)
/bin/grep -q (^:)/usr/sbin($:)
/bin/grep -q (^:)/usr/local/sbin($:)
/usr/bin/id -un
/bin/hostname
/usr/bin/dircolors --sh /etc/DIR_COLORS.xterm
/bin/grep -qi ^COLOR.*none /etc/DIR_COLORS.xterm
/bin/grep -q /usr/kerberos/bin
/bin/grep -q /usr/kerberos/sbin
/sbin/consoletype stdout
/usr/bin/id -u
0;root@CyberArk-oraclelinux504:/
[root@CyberArk-oraclelinux504 /]#
/usr/sbin/useradd evan
[root@CyberArk-oraclelinux504 /]#
Changing password for user evan.
New UNIX password:
/usr/bin/passwd evan
BAD PASSWORD: it is too short
Retype new UNIX password:

```

Example 2: Viewing a raw text recording:

```

f7cc18e2-d9d1-11e3-b916-005056b2209d.txt - Notepad
File Edit Format View Help
000000000000 HDR 000000000151 000000000020| [113|27906|5|1215|35387|32|3|28|127|21
000000000000 CMD 000000000012 /bin/bash -l
000000000379 RCMD 000000000028 /bin/grep -q (^|:)/sbin($|:)
000000000115 RCMD 000000000032 /bin/grep -q (^|:)/usr/sbin($|:)
000000000082 RCMD 000000000038 /bin/grep -q (^|:)/usr/local/sbin($|:)
000000000087 RCMD 000000000015 /usr/bin/id -un
000000000083 RCMD 000000000013 /bin/hostname
000000000091 RCMD 000000000045 /usr/bin/dircolors --sh /etc/DIR_COLORS.xterm
000000000083 RCMD 000000000048 /bin/grep -qI ^COLOR.*none /etc/DIR_COLORS.xterm
000000000076 RCMD 000000000030 /bin/grep -q /usr/kerberos/bin
000000000089 RCMD 000000000031 /bin/grep -q /usr/kerberos/sbin
000000000085 RCMD 000000000024 /sbin/consoletype stdout
000000000081 RCMD 000000000014 /usr/bin/id -u
000000000045 TERM 000000000034 -]0;root@CyberArk-OracleLinux5U4:/
000000000001 TERM 000000000001 *
000000000044 TERM 000000000034 [root@CyberArk-OracleLinux5U4 /]#
000000000908 KYBD 000000000001 u
000000000001 TERM 000000000001 u
000000000117 KYBD 000000000001 s
000000000001 TERM 000000000001 s
000000000134 KYBD 000000000001 e
000000000001 TERM 000000000001 e
000000000366 KYBD 000000000001 r
000000000001 TERM 000000000001 r
000000000294 KYBD 000000000001 a
000000000001 TERM 000000000001 a
000000000230 KYBD 000000000001 d
000000000001 TERM 000000000001 d
000000000167 KYBD 000000000001 d

```

These text files contain a record of the text that was generated during the privileged session. Depending on the type of session and the PSM connection, this file might contain commands that were issued and the channels that were used during the session.

Monitoring Live Sessions

Authorized auditors can monitor live sessions, take part in controlling these sessions, and even terminate them. In order to monitor live sessions, users must be owners of the **PSMRecordings** Safe with the following permission:

- Retrieve files/passwords

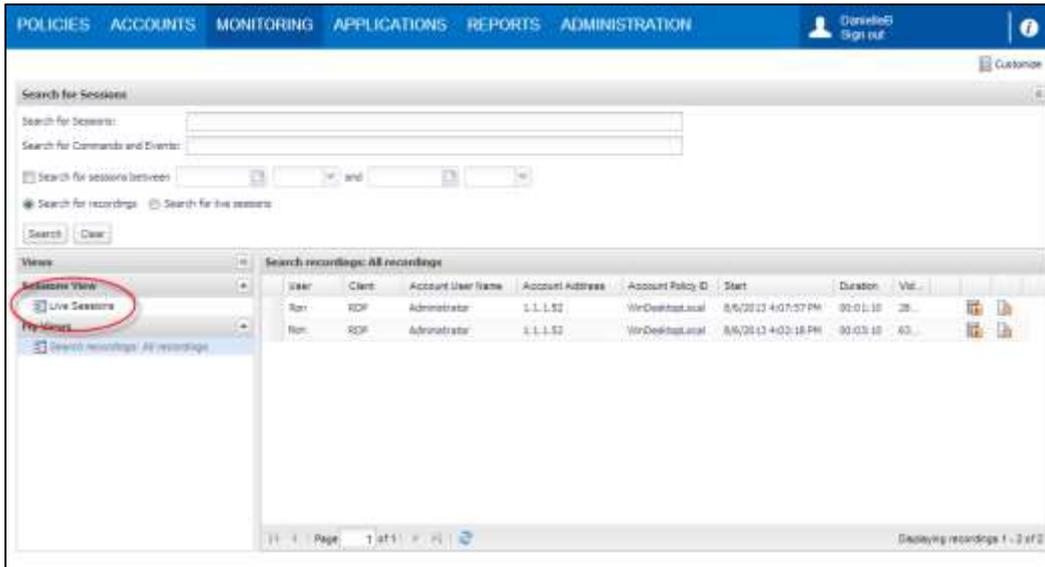
In addition, preconfigured settings determine whether authorized users can view or control live sessions, and/or terminate them.

Listing Live Sessions

The **Live Sessions** view displays all the current live sessions. By default, the list of live sessions is sorted according to the Start column, meaning that the most recently started session appears at the top of the list.

To List Live Sessions

- In the **MONITORING** page, click **Live Sessions**; a list of live sessions is displayed.



- Click on any of the column headings to sort the sessions according to that column. The Live Sessions list facilitates full sorting, meaning that when you sort the displayed sessions according to column, all the sessions are organized in the new order across all the pages in the list.

Finding Live Sessions

You can search for specific live sessions using a free text search according to the properties that are associated with the privileged session (e.g. account name, user, address, device, machine, or any other account keyword). You can also search for sessions according to commands and keystrokes that were performed during the session.

To Find Specific Live Sessions

- In the MONITORING page, specify the keywords that will be used in the search. These keywords may include the following:
 - Search for Sessions** – Information about the live session, such as the name of the user or the privileged account user, the name or IP address of the remote machine, the platform name, port or database name.
 - Search for Commands and Events** – Specific events that were executed during the session.
For example, you can specify a single command to display the live sessions during which that command was issued, or a command and an IP address to display live sessions that are being run from a particular IP address during which that command was issued.

Leave the search edit boxes empty to display all the live sessions.

Note: Specify all or part of a search word. Do not specify wildcards.

- Select **Search for live sessions**.
- Click **Search** or press **Enter**.
- Click **Yes** to begin the search; the Search is performed for all current live sessions.

Viewing Live Session Details

The Live Session details page enables you to see details about live sessions, including details about the account that is being used and a list of all the events that have been issued.

To View Live Sessions

- In the MONITORING page, display the Live Sessions view, then select a specific live session; the Live Sessions details page for that session appears.

This page displays all the details about the session, including the following:

- **Events** – A list of commands and keystrokes that were performed during the live session so far, and the time from the beginning of the session that they were performed.
- **Attestation** – Activities that were performed on the session recording, such as upload, view, recording, monitor, terminate, etc.
- **Advanced** – Detailed information and properties of the live session.

In addition to viewing the information about the live session, you can do the following:

- **Monitor** – Open a copy of this live session on your own workstation and view or take part in controlling the session, depending on the system or platform configurations. For more information, refer to *Viewing and/or Controlling Live Sessions*, page 335.
- **Terminate** – Terminates the live session, if you are authorized. For more information, refer to *Terminating Live Sessions*, page 336.

Viewing and/or Controlling Live Sessions

You can monitor a live session on your workstation and see exactly what tasks the original user is performing. Depending on the system or platform configurations, you can either view a session or actively participate in it.

You can monitor a live session from either of the following pages:

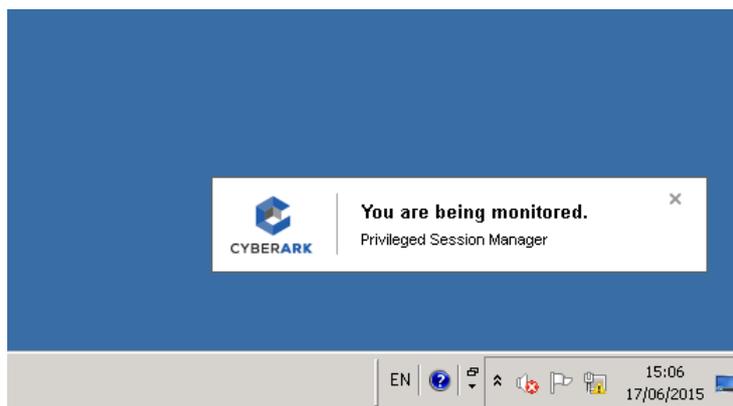
- Live Sessions view in the MONITORING page
- Live Session Details page

To Monitor Live Sessions

- In the MONITORING page:
 1. In the Live Sessions grid, display the live session to monitor.
 2. In the line of the session, click **Monitor**.
- In the Live Session Details page:
 1. Display the Live Session details page of the live session to monitor.
 2. On the toolbar, click **Monitor**.

An additional window opens on your own workstation and displays the live session that you are now monitoring. You can either view or take part in controlling the remote session, depending on how the live session feature is configured.

A notification can be displayed in the live session's screen, as shown in the following example.



Terminating Live Sessions

You can terminate live sessions from your own workstation.

To Terminate Live Sessions

- In the MONITORING page:
 1. In the Live Sessions grid, display the live session to terminate.
 2. In the line of the session, click the **Action menu** icon and then **Terminate**.
- In the Live Session Details page:
 1. Display the Live Session details page of the live session to terminate.
 2. On the toolbar, click **Terminate**.

A message appears prompting you for confirmation.

3. Click **Yes** to terminate the live session,

or,

Click **No** to leave the live session running and return to the Live Session details page.

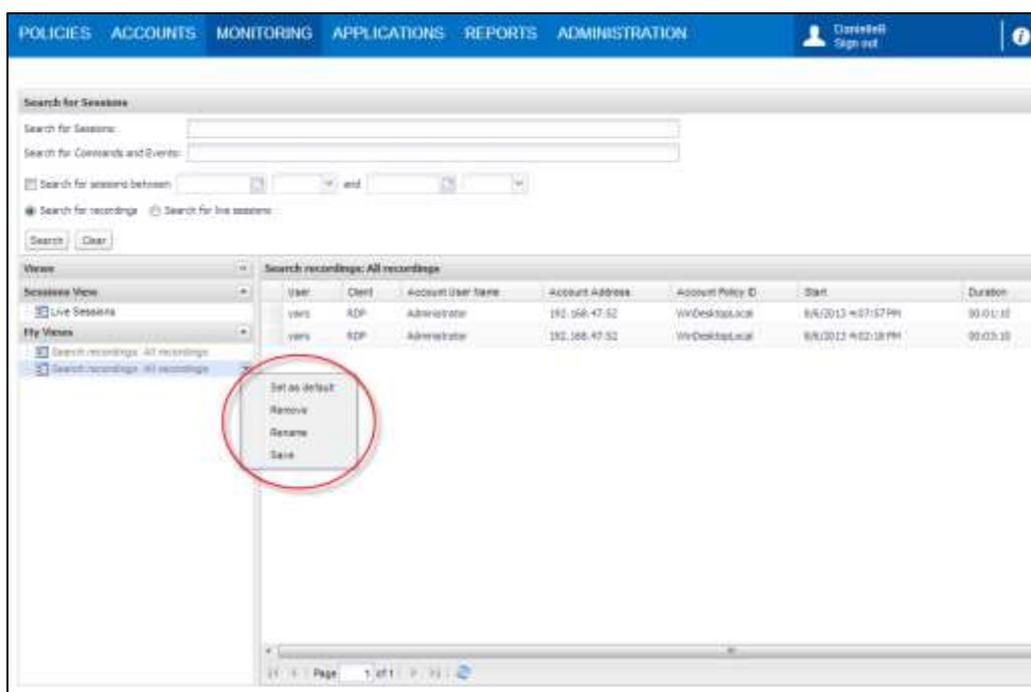
A new window is opened on your workstation and the live session is terminated; a message appears to confirm that the target session was terminated.

Customizing your own Views

Each time you perform a search, the results are listed in My Views where you can display them as often as you wish without repeating the search. Search results are listed temporarily while the user who performed the search is still logged on, after which they are removed from your customized views list. You can save search results so that they are listed again the next time you log on, and even set them as the default view when you display the MONITORING page. You can also rename them or remove them from the list.

To Manage Customized Views

1. In **My Views**, point to the view to display as default, then click the drop-down arrow in the selection.



A pop-up menu enables you to do the following activities:

- **Set as default** – Sets the selected view as the default view that will be displayed when the MONITORING page is displayed.
- **Remove** – Deletes the selected view from the list of customized views.
- **Rename** – Enables you to specify a name for the selected view.
- **Save** – Saves the selected view using the default name.

2. Select the relevant option.

Managing the Recordings List

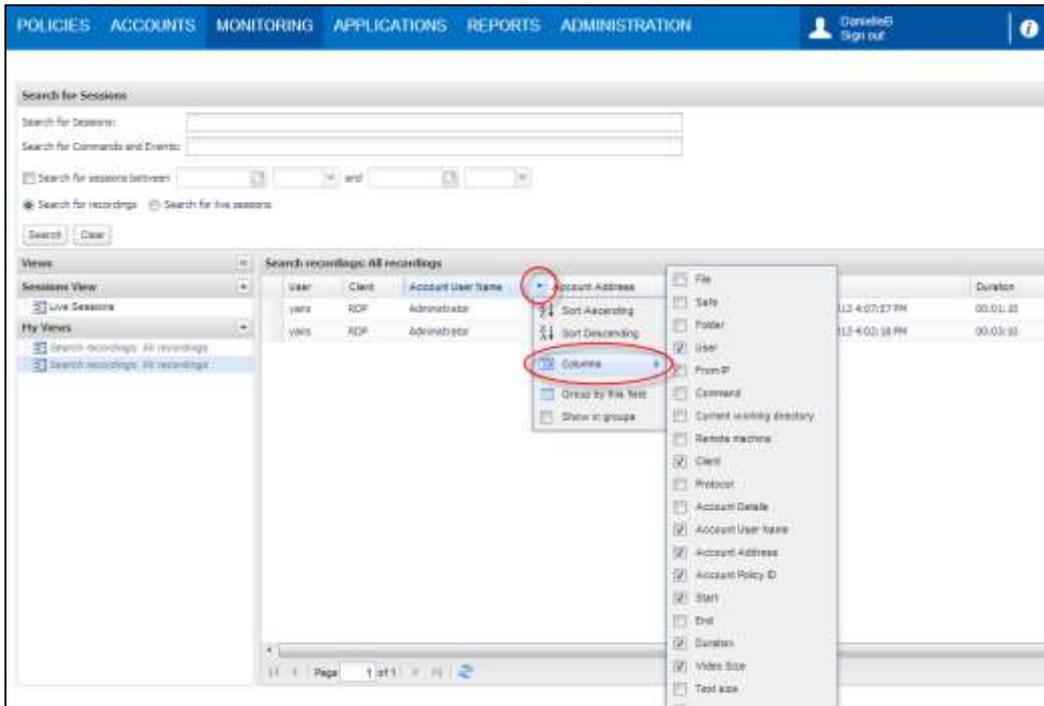
The MONITORING page displays a list of recordings according to their properties. You can customize this list so that it displays your personal preferences, and enables you to view recordings according to your specific needs.

Displaying Hidden Columns

Information about the recordings is displayed in columns. However, by default, not all the available columns are displayed. You can customize your own Recordings list and display the columns that are more useful for your needs.

To Hide and Display Columns in the Accounts List

1. In the Recordings list, click the drop-down button in one of the column titles.



2. From the drop-down menu, select **Columns**, then select or clear the name of the column to display.

Grouping Recordings by Properties

You can organize the displayed recordings according to the properties displayed in the column titles. This enables you to easily identify recordings that have the same properties.

To Group Recordings according to Properties

1. In the Recordings list, click the drop-down button in the title of the column that will determine the property by which recordings will be sorted.
2. From the drop-down menu, select **Group by this field**; the PWA reorganizes the displayed recordings according to the selected property (column title).

Auditing Accounts

Inspecting Accounts Activity

Authorized users can inspect activity that has been performed on accounts or files in the Safe. The Activity tab in the Account Details page displays the dates and times that an account or file is handled, as well as the names of Users who have retrieved, modified, or added it.

In order to see the Activity tab, users require the following Safe member authorization:

- View audit

If the account can only be retrieved after the user specifies a reason, requires confirmation from authorized users, or after the retrieval is validated by a ticketing system, all the relevant information is also displayed in the Activity tab.

To Inspect Account Activity

- In the Accounts list, select the account to inspect; the Account Details page appears.
- Select the **Activities** tab; all the activity for the selected account is displayed in this pane.

Time	User	Action	Client ID	Name of file	Reason
8/6/2013 4:02:08 PM	Run	PSM Disconnect	PSMApp	Application T...	
8/6/2013 4:07:59 PM	Run	PSM Connect	PSMApp	Application T...	
8/6/2013 4:07:51 PM	Run	Use Password	PWSA		(Action: Conn...
8/6/2013 4:07:43 PM	Run	Add File Category	PWSA	Logo/Conver...	Value=[132]...
8/6/2013 4:05:42 PM	Run	Retrieve password	PWSA		(Action: Show...
8/6/2013 4:04:38 PM	Run	Retrieve password	PWSA		(Action: Show...
8/6/2013 4:00:14 PM	Run	Update File Category	PWSA	Address	Value=[132]...
8/6/2013 4:05:14 PM	Run	Update File Category	PWSA	Username	Value=[Address...
8/6/2013 4:05:51 PM	Run	Update File Category	PWSA	Address	Value=[Local]...
8/6/2013 4:05:51 PM	Run	Update File Category	PWSA	Username	Value=[Local]...
8/6/2013 4:05:29 PM	Run	PSM Disconnect	PSMApp	Application T...	
8/6/2013 4:02:08 PM	Run	PSM Connect	PSMApp	Application T...	
8/6/2013 4:02:06 PM	Run	Use Password	PWSA		(Action: Conn...
8/6/2013 3:53:21 PM	Run	Add File Category	PWSA	Address	Value=[132]...

Auditing Privileged Single Sign-On

Whenever users connect to a remote machine transparently, the details of the connection are saved in the activity log of the account. If a domain/NIS password is used to log onto a remote machine, the IP address or DNS that the user specified manually is also saved in the activity log.

When PSM is used for privileged SSO, an additional audit record is written in the account activities log when the privileged session is disconnected. This provides a complete audit of all activities performed during the privileged session.

The Privileged Account Security Solution Reports

Reports are an essential part of the Privileged Account Security solution, as they enable you to see the amount and type of activity that is taking place in the Vault.

The Privileged Account Security solution works together with Excel and text files to give you a variety of report generation options. By generating reports directly into the application of your choice, you can mold the information to your specific output requirements. If you choose to generate a report into a text or CSV file, you can then import the information into a third party or report generator application to suit your unique requirements.

Report configurations can be saved to create a standard and consistent report that meets your enterprise needs. Reports can be scheduled for automatic generation on a weekly or monthly basis. This configuration includes the type of report, its content, users who will be able to access it, and whether or not these users will receive an automatic notification each time the report is generated.

The reports include a large variety of details that keep you informed of objects and activity in the Vault, and can be divided into two groups:

- **Operational reports** – These reports contain information about the information stored in the Vault, Safes and users, and the operational connections between them.
 - **Privileged Accounts Inventory** – Provides information about all the privileged accounts in the system, based on different filters.
 - **Applications Inventory** – Provides information about the application IDs in the system, based on different filters.
 - **Safes List** – A list of Safes and their properties according to location.
 - **Owners List** – A list of Owners of the specified Safe(s) and their permissions.
 - **Active/Non-active Safes** – A list of active or non-active Safes for activities over a specified period of time. The report includes a list of active or non-active Safes and some of their properties.

The active or non-active status of the Safe is determined by the administrative or data-related tasks that were carried out in it, and not by whether it was opened or closed.

Note: The 'Last Used' date is updated for both administrative and data-related tasks and also for Safe open and close operations.

- **License Capacity Report** – The licensed user types and objects in the Vault, the maximum number of licenses for each type or object, and the number of used licenses for each one.
- **Users List** – A list of Users and disabled Users according to their location, including the location's quota and the User's own quota.
- **User Activities** – Users' activities in the Vault, including those who have been disabled. The activities do not include data-related activities. These reports can be generated by User Managers and by the Auditor User.
- **Active/Non-active Users** – Active or non-active Users, including disabled Users, in the specified Vault over the specified period of time. Active Users are defined as those who have logged on to the Vault for whatever purpose.

- **Audit/compliance reports** – These reports contain information that enable you to track Safe activities and, specifically, password use in order to meet audit requirements.
 - **Privileged Accounts Compliance Status** – An inventory that indicates which accounts are compliant with their platforms, how accounts are managed in order to make them compliant, when password changes are planned, and their management status.
 - **Entitlement Report** – Users' entitlement rights in the Privileged Account Security solution regarding user, Safe, active platform, target machine, target account, etc. This report includes each user's effective access control and authorization level on each account that the user has access to in the Privileged Account Security solution.
 - **Activities Log** – A log of all the activities that have taken place in the Safe(s). This report can be filtered according to user, target system, specified period, and a variety of other criteria.

Some of these reports can be generated in the PVWA, while others can only be generated in the PrivateArk Administrative Client.

Generating Reports in the PVWA

Reports can be generated in the Reports page in the PVWA by users who belong to the group that is specified in the **ManageReportsGroup** parameter in the Reports section of the Web Access Options in the System Configuration page. By default, this is the PVWAMonitor group.

The following reports can be generated in the PVWA:

- **Privileged Accounts Inventory** – Provides information about all the privileged accounts in the system, based on different filters. Users who have the following authorizations can generate this report:
 - **List accounts** and **View Safe members** in the Safes that will be included in the report.

Note: Users who do not have the View Safe members authorization will only be able to view complete information about their own activities.

This report includes the following output:

Column	Description
Safe	The name of the Safe where the privileged account is stored.
Folder	The name of the folder where the privileged account is stored. Note: This column is not displayed by default.
Name	The name of the privileged account. Note: This column is not displayed by default.
PlatformId	The ID of the platform that is associated with the privileged account.
DeviceType	The type of device that the privileged account is allocated to.
Username	The name of the user that is authorized to use the privileged account on the remote device.
Address	The address of the remote device where the privileged account is used.
Group	The group that the privileged account belongs to, if any.

Column	Description
LastAccessedDate	The date when the privileged account was last accessed.
LastAccessedBy	The name of the last human user who accessed the privileged account.
LastModifiedDate	The date when the privileged account was last modified by any user, human or component.
LastModifiedBy	The name of the last human user who modified the privileged account.
VerificationDate	The last date when the privileged account was verified.
CheckoutDate	The last date when the privileged account was checked out.
CheckedOutBy	The name of the last user who checked out the privileged account.
Age	The number of days since the password was created.
ChangeFailure	Whether or not the password in the privileged account could be changed. Note: 'Yes' indicates that the password change failed.
VerificationFailure	Whether or not the password in the privileged account could be verified. Note: 'Yes' indicates that the password verification failed.
MasterPassFolder	The folder where the master account associated with the current usage is stored. Note: This column is not displayed by default.
MasterPassName	The name of the master account associated with the current usage. Note: This column is not displayed by default.
DisabledBy	Whether the privileged account was disabled by a human user or by the CPM. Note: This column is not displayed by default.
DisabledReason	The reason why the privileged account was disabled.

- **Applications Inventory**– Provides information about the application IDs in the system, based on different filters. Users who have the following authorizations in the Vault can generate this report:

- Audit Users

Note: Users can generate this report for users in the same level or lower in the Vault hierarchy.

This report includes the following output:

Column	Description
Default columns:	
Application ID	The unique ID of an application defined in the Vault.
Business owner	The full name of the application's business owner.
Location	The location of the application in the Vault's hierarchy.

Column	Description
Allowed machines	A list of machines that are defined in the Vault for the listed application and from where the application can request a privileged account.
OS User/s	A list of OS users that are defined in the Vault for the listed application and can request a privileged account.
Path/s	A list of paths that are defined in the Vault for the listed application and from where the application can request a privileged account.
Optional columns:	
Application description	A description of the application requesting the privileged account.
Business owner email	The email address of the application's business owner.
Business owner phone	The phone number of the application's business owner.
Disabled	Whether or not the application's Vault definition is disabled. If the application is disabled in the Vault, 'Yes' is displayed. If not, 'No' is displayed.
Hash	A list of unique hash values that has been created by the AIM utility to enable the application to authenticate to the Vault and retrieve a privileged account.
Access permitted	The date from when and until when the application is permitted to access the privileged account.
Expiration date	The date when the application's Vault definition expires.

- **Privileged Accounts Compliance Status** – Accounts compliance and management status. Users who have the following authorizations can generate this report:
 - The following permissions are required in the Safes that will be included in the report:
 - List passwords/files
 - View Audit or Confirm Safe request – in Safes that are configured for dual control
 - Membership in the following group:
 - PVWAMonitor
 - To enable users to run this report for the entire system, add them as members to the following group:
 - Auditors

This report includes the following output:

Column	Description
Default Columns – These columns are displayed in the output report by default.	
Target system user name	The name of user or group that users the account on the target system.
Target system address	The address of the target system.

Column	Description
Safe	The name of the Safe where the privileged account is stored.
Platform name	The unique platform name that is associated with the privileged account.
Compliance status	Whether or not the account is compliant with the Master Policy. Possible values are: <ul style="list-style-type: none"> Compliant Non Compliant
Non-compliance reason	If the account is not compliant, the reason why. If the account is not compliant for more than one reason, the report will display the reason for the most recent change that did not take place. Possible values are: <ul style="list-style-type: none"> Password expired One time password not changed N/A – if the account is compliant
Expiration due (days)	The number of days until the password is due to expire. If the password has already expired, a negative number will be displayed. If the password is due to expire in less than one day, a fraction will be displayed. Note: The password must be changed during this time in order to stay compliant.
Planned password change	The date and time when the password in the account will be changed in order to stay compliant. This may be affected by any of the following: <ul style="list-style-type: none"> Expiration change period taken from the platform One-time access Head-start interval Execution days and the relative timeframe If the password was not changed on the planned date/time, the missed date is displayed. Note: This is not necessarily the expiration date as the system will usually start changing the password before the expiration (due to the Headstart parameter). It is an optional date before the password expiration date so that the password does not expire before it is changed.
Change mode	The mode used to change the password. Possible values are: <ul style="list-style-type: none"> Automatic Automatic but disabled Automatic with a manual trigger Automatic with a manual trigger but disabled Manual
One-time password	Whether or not the Master Policy defines one-time passwords.
Expiration period	The number of days left until a password expires, according to the Master Policy.
Last modified date	The date and time when the password was last changed.
Last accessed by	The name of the last human user who accessed the privileged account.
Last accessed	The date and time when the password was last accessed.

Column	Description
Last access request timeframe	The timeframe specified in the request for the last access.
Optional Columns – These columns are not displayed in the output report by default, but can be added.	
Exclusive Access	Whether or not an account is locked each time it is retrieved, so that it cannot be retrieved by a second user.
Dual control	Whether or not this account can only be accessed after confirmation from one or more authorized users.
Perform periodic change	Whether or not accounts will be changed periodically according to the password change settings.
Allow manual change	Whether or not the password in the account can be changed manually (a.k.a. "Change Now").
Head start	The number of days before the password expires (according to the ExpirationPeriod parameter) that the CPM will initiate a password change process.
Access validity period	The number of hours during which this account can be accessed before it will be changed.
Password length	The length of automatically generated passwords according to the platform.
Password complexity policy	Details about the password complexity rules, according to the platform, including the following: <ul style="list-style-type: none"> ▪ Password Length ▪ Number of upper case characters ▪ Number of lower case characters ▪ Number of digits ▪ Number of special characters ▪ Forbidden characters
Password change reason	The specified reason for a planned change. The possible values are: <ul style="list-style-type: none"> ▪ Expiration period change ▪ One time usage ▪ None <p>If there is more than one reason for the planned password change, the reason for the nearest change will be displayed.</p>
Disabled	Whether or not automatic management has been disabled for this account. Possible values are: <ul style="list-style-type: none"> ▪ Yes (by user) ▪ Yes (by CPM) ▪ No
Disabled reason	If automatic account management has been disabled, the reason why.
Change failure	Whether or not the last password change process failed.
Failure reason	If the last password change process failed, the reason why.
Password age (days)	The number of days that have passed since the last password change.
Expiration date	The date and time when the password is due to expire. Note: This date might be different from the planned change date.
Creation date	The date when the account was created.

Column	Description
Deleted	Whether or not the account was deleted.
Last modified by	The name of the last user who modified the privileged account.
Last verification date	The date and time when the password was last verified.
Folder	The name of the folder where the account is stored.
Name	The name of the privileged account.
Device type	The type of device on which the account is used.
<account property>	Any account property defined by the report configuration (any account property can be added to the configuration).

- **Entitlement Report** – Users' entitlement rights in the Privileged Account Security solution. Users who have the following authorizations in the Vault can generate this report:

- Either Manage Users or Audit Users

This report includes the following output:

Column	Description
User	The name of user or group whose entitlement rights are detailed in this report.
Full name	Full name of the user or group.
Group	Whether or not the user is a group, If it is a group, 'Yes' is displayed. If not, 'No' is displayed.
Group membership	Whether users' Safe authorizations are defined by their group membership or directly in their user account. If access is through group membership, 'Yes' is displayed. If access is through direct user ownership, 'No' is displayed.
Location	The location of the user/group in the Vault hierarchy.
User type	The user's type as defined in the CyberArk license (EPV, AIM, CPM, etc).
Target platform name	The unique platform name that is associated with the account listed in the next column.
Target system	The hostname or IP address of the target system where the privileged account is used.
Target account	The name of the privileged account.
Safe	The name of the Safe where the privileged account is stored.
Object Name	The name of the account.
Folder Name	The name of the folder where the account is stored.
Read	Whether or not the user/group has permission to view the privileged account. If the user/group has permissions, 'Yes' is displayed, if not, 'No' is displayed.

Column	Description
Use	Whether or not the user/group has permission to use the privileged account. If the user/group has permissions, 'Yes' is displayed, if not, 'No' is displayed.
Change	Whether or not the user/group has permissions to change the privileged account. If the user/group has permissions, 'Yes' is displayed, if not, 'No' is displayed.
Other permissions	A list of additional permissions allocated to the user/group.
Command	The name of the command or command group that has been defined.
Command Entitlement	The name of the user or group that is defined with this command or command group.
Allow/Deny	Whether the user or group is allowed or denied use of this command or command group.
Command Restriction	The restriction that is applied to the specified command or command group. Each restriction is displayed in its own column.

- **Activity Log** – Activities performed in the Vault. Users who have the following authorizations can generate this report:

- **User related activities** – Audit Users in the Vault

Note: Users can generate this report for users in the same level or lower in the Vault hierarchy.

and

- **Safe/Account related activities** – View Audit in Safes that will be included in the report

This report includes the following output:

Column	Description
Time	The time when the activity was performed. By default, the date and time are displayed in the timezone of the web server. However, the PVWA can be configured to display the date and time in different time zones.
User	The full name of the user who performed the activity.
Action	The activity that was performed.
Safe	The Safe where the privileged account is stored.
Target	The privileged account that was used in the activity.
Target Platform	The unique ID of the platform that was allocated to the privileged account used in the activity. This is only relevant to Account activities reports.
Target System	The remote system where the privileged account was used. This is only relevant to Account activities reports.
Target Account	The name of the target account where the privileged account was used. This is only relevant to Account activities reports.

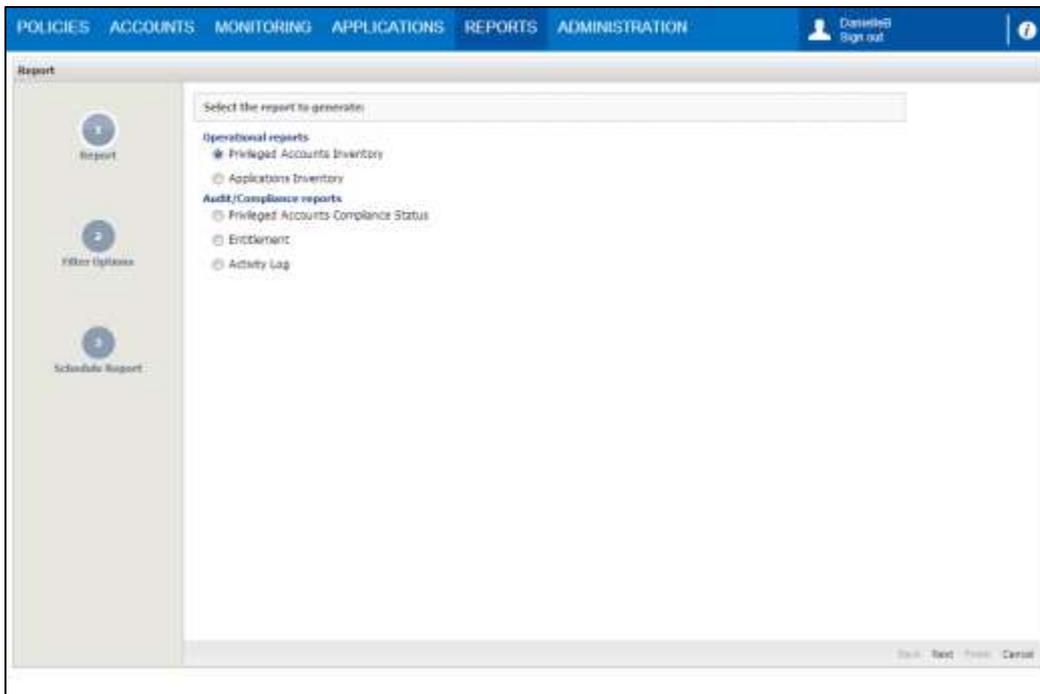
Column	Description
New Target	The new location/name of an account on which the activity was performed and extra details about the activity that was performed. This is only relevant to Account activities reports.
Reason	The reason given by the user for performing the activity.
Alert	An indication that this activity prompted an alert.
Request ID	The unique ID of the request that was created in order to retrieve the privileged account used in the activity.
Client ID	The unique ID of the client used in the activity.

Reports are saved in the Safe that is specified in the **DefaultSafe** parameter in the Reports section of the Web Access Options. By default, this is the PVWAReports Safe which is created automatically when the first report is generated. For more information about the Reports Safe, refer to *The Environment in the Password Vault* in the Privileged Account Security Installation Guide.

Report parameters can be configured in the Reports parameters of the Web Access Options in the System Configuration page. For more information, refer to *Reports*, page 562, in *Configuring the PVWA*.

To Generate Reports in PVWA

1. Click **REPORTS** to display the My Reports page.
2. Click **Generate Report**; the Report wizard appears.



3. Select the report to generate, then click **Next**; the Filter Options page appears.

This page enables you to specify filters for the report. You can specify any of the following filters:

- **Privileged Accounts Inventory**

Filter	Specifies
General filters – These general filters are common to the Privileged Accounts Inventory.	
Free Search	A free text search for contents of this inventory. The properties that are searched are configured in the Reports parameters of the Web Access Options in the System Configuration page. For more details, refer to <i>Configuring the System through PVWA</i> , page 918.
Safe	Part of the name or the whole name of the Safe/s where the PVWA will search for information.
Account name	The full name or part of the name of accounts to include.
Device type	Part of the name or the whole name of the device for which all assigned accounts will be included. If this field is left empty, accounts for all device types will be included.
Platform name	Part of the name or the whole name of the platform name for which assigned accounts will be included. If this field is left empty, accounts that are not assigned to any platform will be included.
Group	All accounts associated with the specified group will be included. If this field is left empty, accounts that are not associated with any group will be included. This field is only enabled if a Safe name or part of one has been specified.
Include account usages	Whether or not service accounts will be included.

Filter	Specifies
Automatic Management – Specifies automatic management filters.	
No filter	No automatic management filters will be used.
Only accounts with failures	Only accounts whose automatic management failed and could not be managed will be included.
Include only disabled accounts	Only accounts that were disabled as a result of repetitive failures will be included.
Activities – Specifies activity filters.	
No filter	No activity filters will be used.
Accessed by users in the last days	Only accounts that were accessed by human users during the last specified number of days will be included.
Changed in the last days	Only accounts that were changed during the last specified number of days will be included.

- **Applications Inventory**

Filter	Specifies
Free Search	A free text search for applications to include in this inventory. The search is performed in the following properties: <ul style="list-style-type: none"> ♦ Application ID ♦ Application Description ♦ Business Owner Name ♦ Business Owner Email ♦ Authentication Details ♦ Machine IP
Location	The name of the Location in the Vault hierarchy where the PVWA will search for application IDs.

- **Privileged Accounts Compliance Status**

Filter	Specifies
General filters – Defines general filters for the Privileged Accounts Compliance Status report.	
Free Search	A free text search for contents of this status report. The properties that are searched are configured in the Reports parameters of the Web Access Options in the System Configuration page. For more details, refer to <i>Configuring the System through PVWA</i> , page 918.
Safe	Part of the name or the whole name of the Safe/s where the PVWA will search for information.
Account name	The full name or part of the name of accounts to include (name of password object in the Vault).
Device type	Part of the name or the whole name of the device for which all assigned accounts will be included. <ul style="list-style-type: none"> ▪ Leave this field empty to include accounts for all device types.

	<ul style="list-style-type: none"> ▪ Select No device associated to include accounts that are not associated to a device.
Platform name	<p>Part of the name or the whole name of the platform name for which assigned accounts will be included.</p> <ul style="list-style-type: none"> ▪ Leave this field empty to include all accounts. ▪ Select No platform associated to include only accounts that are not associated to a platform.
<p>Account Expiration Status – Defines filters applied to detect the accounts expiration status according to the expiration period or one time usage settings in the Master Policy. Note: This filter does not support account groups.</p>	
No filter	No account expiration filters will be used.
Only expired accounts	Only accounts that have already expired will be included.
Include accounts about to expire	Accounts that will expire during the specified period of time will be included. Users can customize the timeframe to search.
<p>Password Change Mode – Filters accounts according to the way the password is changed.</p>	
No filter	No password change filters will be used.
Only accounts changed manually	Only accounts that are changed manually will be included.
Only accounts changed via CPM with manual trigger	Only accounts that are changed automatically by the CPM with a manual trigger will be included.
Only accounts changed via CPM automatically	Only accounts that are managed automatically by the CPM will be included.
<p>Automatic Management Status – Filters accounts according to the CPM management status.</p>	
No filter	No automatic management status filters will be used.
Only accounts with failures	Only accounts that are marked with automatic management failures will be included.
Include only disabled accounts	Only disabled accounts will be included.
<p>Activities – Filters accounts according to the most recent access/change activities.</p>	
No filter	No activity filters will be used.
Accessed by users in the last <number> days	Accounts that were accessed by users during a customized period of time.
Changed in the last <number> days	Accounts that were changed by users during a customized period of time.

- **Entitlement Report**

Filter	Specifies
Location	The name of the Location in the Vault hierarchy where the PVWA will search for users and groups.
Include sublocations	Whether or not the PVWA will search for users and groups in sublocations of the specified Location.
User or group	The name of the user or group in the Vault for which the report will be created. You can specify either a name or a wildcard.
Include groups	Whether or not groups will be included.
Include disabled users	Whether or not disabled users will be included.
User type	The User type that will be included.
Safe name	The name of the Safe that will be included. You can specify either a Safename or a wildcard.
Include OPM command permissions	Whether or not privileged commands and command groups will be included.
Target Systems – Specifies target systems to include in the report.	
Platform name	The platform name to include in the report. You can specify either a name or a wildcard.
System	The address of the target system, specified in the account. You can specify either a name or a wildcard.
Account	The name of the account user, specified in the username property. You can specify either a name or a wildcard.

- **Activity Log**

Filter	Specifies
General filters – These general filters are common to the Activity Log report. Note: The list for each filter only contains the first 500 objects. Specify an object that does not appear in the list by typing it.	
Location	The name of the Location in the Vault hierarchy where the PVWA will search for users and groups.
Include sublocations	Whether or not the PVWA will search for users and groups in sublocations of the specified Location.
User or group	The name of the user or group that will be included. You can specify either a name or a wildcard.
User type	The User type that will be included.
Safe	The name of the Safe that will be included. You can specify either a name or a wildcard.
Client ID	The client ID that will identify Safe activities that will be included in the report. Note: Safe activities can be filtered according to the Client ID, but user activities cannot. Therefore, this report will contain information about user activities for all Client IDs.
Request ID	The unique ID of the request related to activities included in this report.

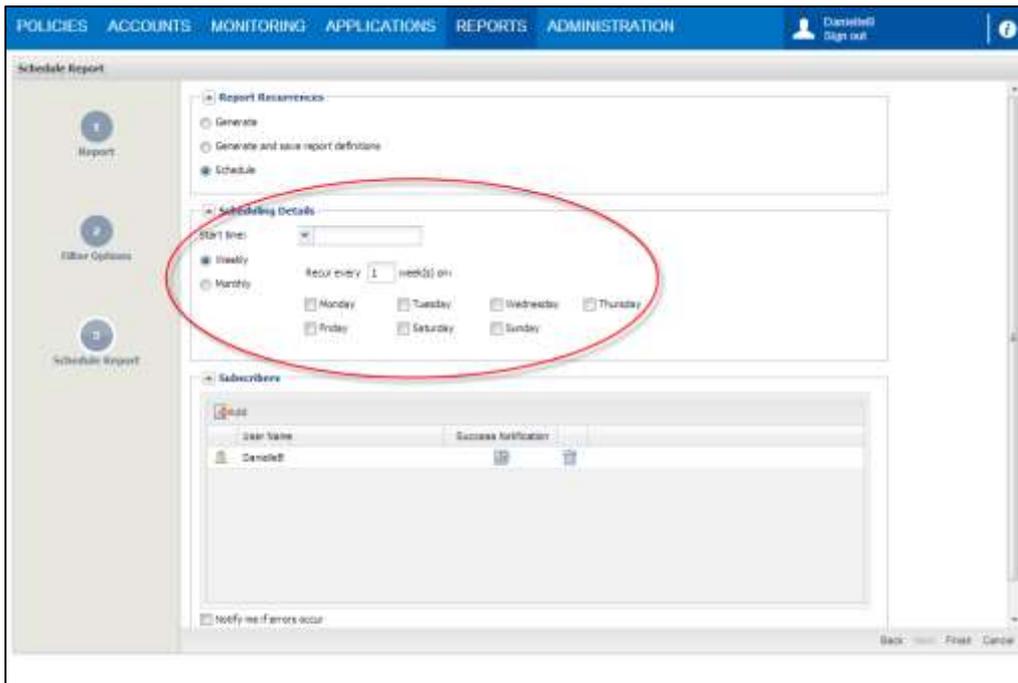
Filter	Specifies
Display only alerts	Whether or not only alerts will be displayed.
Target Systems – Specifies target systems to include in the report.	
Platform name	The platform name to include. You can specify either a name or a wildcard.
System	The address of the target system, specified in the account. You can specify either a name or a wildcard.
Account name	The name of the account user, specified in the username property. Wildcards are not supported.
Activities – Specifies the activities to include in the report. Users can select one or more types of activity from the following predefined list. If none of the activities are selected, all activities will be included in this report.	
<ul style="list-style-type: none"> ◆ Privileged Account Access Activities ◆ Approval Workflow Activities ◆ Privileged Account Management Activities ◆ Privileged Account Auto-detection Activities ◆ Safe and Member Management Activities ◆ User Login Activities ◆ User, Group and Location Management Activities ◆ Vault System Administration Activities ◆ Reports Management Activities ◆ File Access Activities ◆ File Management Activities 	
History – Specifies the history to include in the report.	
All actions	All actions in the Vault activities log will be included.
All actions during the previous days	All actions that occurred during the previous number of days will be included.
All actions between	All actions that occurred between the specified dates will be included in the report.

4. Specify the General report filters and any other filters that will specify the exact report you want, then click **Next**; the Schedule Report page appears.

This page enables you to schedule reports for automatic and manual generation, and specify which users can access them.

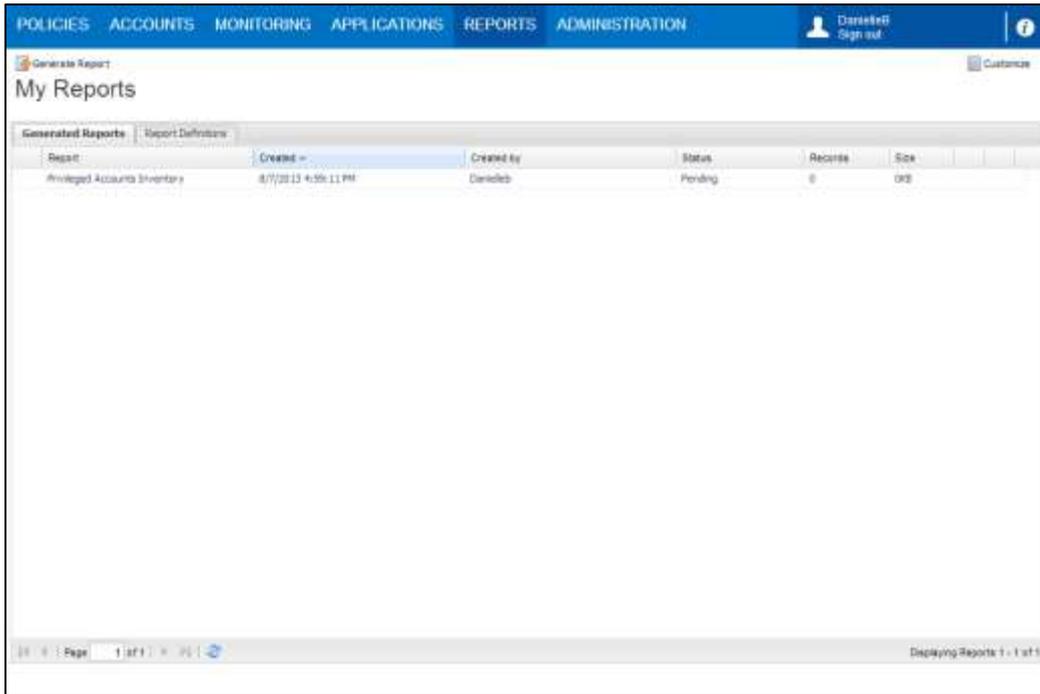
5. In the **Report Recurrences** section, specify the filters that determine how frequently this report will be generated:

Filter	Description
Generate	Generate the report. This is the default value.
Generate and save report definitions	Generate the report and save the report definitions to use again.
Schedule	Schedule the report to run weekly or monthly. When you select this option, a drop-down section appears and enables you to specify how frequently the report will be generated. Specify the following information:
	Start time The time when the report generation will begin.
Weekly	Specify the number of weeks between report generations, and the days of the week when the report will be generated.
Monthly	Specify the frequency and the day of the week when the report will be generated.



6. In the **Subscribers** section, add the users who will be able to access the generated report. The name of the user who is currently defining the report is already listed in the Subscribers list.
 - i. Click **Add**; the Add Subscriber window appears.
 - ii. In the Search field, specify the name or part of the name of a user or group to add as a subscriber to this report.
 - iii. From the User Type drop-down box, select Users & Groups, Users, or Groups to indicate the type of subscriber to look for during the search.
 - iv. In the Search In drop-down box, select either Vault or the External Directory where the user that you will add as a subscriber is defined.
 - v. Click **Search**; a list of users and/or groups in the Vault or the specified external directory that match the name specified in the Search field is displayed.
 - vi. Select the user or group to add as a subscriber.
 - vii. To specify that the user or group will receive an automatic notification whenever this report has been generated, select **Notify user/group when report is ready**.
 - viii. Click **Select**; the user is added as a subscriber to this report.
7. Select **Notify me if errors occur** to send a notification to the user generating the report if an error occurs and it cannot be generated.

Whether or not this option is selected, a notification can be sent to one or more groups if a report is not generated successfully. The name of these groups is specified in the Reports parameters in the System Configuration page.
8. Click **Finish**; the report is now generated and is displayed in the Generated Reports tab in the My Reports page.



Reports only contain the information that the user who generated the report is authorized to access. Any other information will not be included in the report, regardless of the specified properties in the Reports parameters. For more information, refer to *Configuring the System through PVWA*, page 918.

The status column displays the current status of the report. This could be any of the following:

- **Pending** – The report has not yet been generated.
- **Running** – The report is currently being generated.
- **Failed** – The report could not be generated. A tooltip describes the reason why the report generation failed.
- **Done** – The report was generated successfully, and can be viewed and saved in either CSV or Excel format.

By default, all reports are saved in the **PVWAReports** Safe which can be accessed by members of the **PVWAMonitor** group. Both of these are configurable in the Reports parameters.

Each user can view the reports they have generated, and which appear in their My Reports page. Only users who have specifically been given access authorizations in the Reports Safe will be able to see all the reports. Otherwise, users will be able to view their own reports or reports that they were subscribed to when the report was generated.

Managing Reports

After reports have been generated successfully, they can be managed in the My Reports page in the following tabs:

- **Generated Reports** – This tab lists all the reports that have been generated successfully and whose status is 'Done'.
- **Reports Definitions** – This tab lists the reports whose definition was saved as well as scheduled reports.

Generated Reports

The Generated Reports tab displays reports that have been configured, regardless of their status. By default, this list displays the name of the report, when it was created, the user who created it, its status, and the number of records in the report. These columns can be modified by authorized users in the System Configuration page.

Reports whose status is 'Done' can be managed with the following options:

- **Save** – Users can either open or save reports either in Excel or in CSV format.
- **Protect** – Users can protect reports whose status is either 'Done' or 'Failed' so that they will not be deleted automatically after the retention period expires. Users can also 'Unprotect' reports whenever they wish.
- **Delete** – Users can delete reports that are no longer needed.
- **Hide** – Users can hide a report so that it is no longer displayed in the Reports List. If the user is one of several subscribers to this report, the report will be hidden and not be deleted, and will be deleted according to the retention period in the Safe. However, if this user is the only subscriber to this report, hiding the report will delete it.
- **Send** – Users can send a notification that contains a link to a report to other subscribers.

To Manage Reports

- In the Generated Reports list, click the action icon on the report to manage; the action menu appears and displays the actions that can be performed.



Saving and Opening Reports

Reports can either be saved as Excel files or in CSV format which can be imported into third party applications.

To Save and Open a Report in Excel Format

1. In the Generated Reports list, click the **Save report in Excel format** icon to save or open the report.



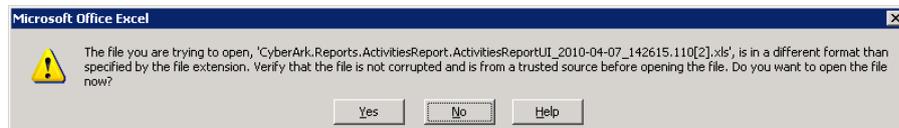
The File Download window appears.

2. Click **Open** to save the report as an Excel file and open it immediately,

or,

Click **Save** to save the report and open it another time; the Save As window appears enabling you to specify a location to save the report.

Note: In Office 2007, due to a new security feature, the following warning message is displayed:

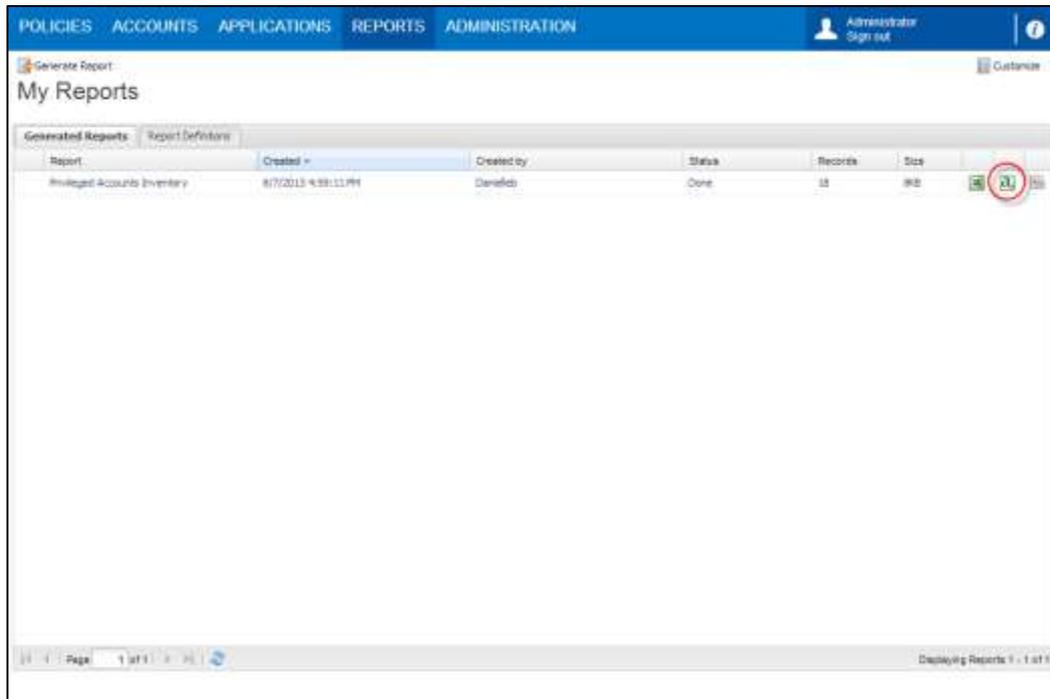


Click **Yes** to open the report in Excel.

3. The Excel files display generated reports using preconfigured settings. For more information about configuring reports in Excel, refer to *Reports*, page 562, in *Configuring the PVWA*.

To Save and Open a Report in CSV Format

1. In the Generated Reports list, click the **Save report in CSV format** icon to save or open the report.



The File Download window appears.

2. Click **Open** to save the report in CSV format and open it immediately,
or,
Click **Save** to save the report as a CSV file and open it another time; the Save As window appears enabling you to specify a location to save the report.

Protecting Reports

Reports are deleted automatically after the Safe retention period expires. If you have reports that you do not want deleted automatically, you can protect it so that it will not be deleted automatically. While protected, the report cannot be deleted manually either.

In order to protect and unprotect reports, users must belong to the group that is authorized to manage reports, by default called **PVWAMonitor**,

To Protect a Report

In order to protect reports, users must have the **Retrieve files/accounts** authorization in the Safe where the reports are stored, by default **PVWAReports**.

- In the Generated Reports list, click the action icon for the report to protect, then select **Protect**; the PVWA protects the report immediately, and it cannot be deleted.

To Unprotect a Report

In order to unprotect reports, users must either have created the report or must have the **Unlock files/accounts** authorization in the Safe where the reports are stored, by default **PVWAReports**.

- In the Generated Reports list, click the action icon for the report to unprotect, then select **Unprotect**; the PVWA removes protection from the report immediately. This report will be deleted automatically after the Safe retention period expires, and can also be deleted manually as described below.

Deleting Reports

When reports are no longer needed, they can be deleted manually.

In order to delete reports, users must belong to the group that is authorized to manage reports, by default called **PVWAMonitor** and must have the **Delete files/accounts** authorization in the Safe where the reports are stored, by default **PVWAReports**.

To Delete a Report

1. In My Reports, click the action icon for the report to delete, then select **Delete**; a message appears prompting you for confirmation.
2. Click **Yes** to delete the report,
or,
Click **No** to leave the report in the Safe and return to the Reports list.

Report Definitions

The Report Definitions tab displays a list of reports that have been saved and/or scheduled. By default, this list displays the name of the report and how frequently the report will be generated, if the report is scheduled. These columns can be modified by authorized users in the System Configuration page. Reports can be managed with the following options:

- **Generate Now** – Preconfigured reports can be generated at any time.
- **Edit** – Report scheduling and subscribers can be edited after the report has been configured.
- **Delete** – Users can delete report configurations that are no longer needed.

To Manage Report Definitions

- In the Report Definitions list, click the action icon on the report configuration to manage; the action menu appears and displays the actions that can be performed.

To Edit a Report

Authorized users can edit reports. Unauthorized users will see this page in View Only mode.

1. In the Report Definitions list, click the report definition to edit,
or,

Click the action icon for the report definitions to edit, then select **Edit**.

The Report wizard for editing a report appears.

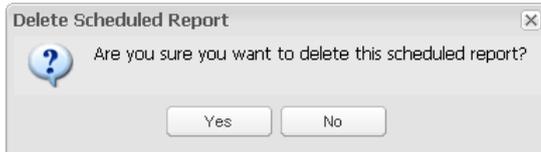
2. Edit the following pages as needed:
 - **Filter Options** – This page contains the filters specified for the report.
 - **Schedule Report** – This page contains the scheduled reports for automatic and manual generation, and defines which users can access them.For more information, refer to *To Generate Reports in PVWA*, page 348.
3. Click **Next** to proceed to the next page in the wizard,
or,
Click **Finish** to save your changes and display the Reports page.

Deleting Report Definitions

When report definitions are no longer needed, they can be deleted manually.

To Delete a Report Definition

1. In the Report Definitions list, click the action icon for the report definitions to delete, then select **Delete**; the following message appears prompting you for confirmation.



2. Click **Yes** to delete the report definitions,
or,
Click **No** to leave the report definitions.

Running Preconfigured Reports Manually

Preconfigured reports can be run manually at any time by authorized users.

To Generate a Preconfigured Report Manually

- In the Report Definitions list, click the action icon for the preconfigured report to run, then select **Generate Now**; the report is generated according to the report configurations, and can be viewed in the Generated Reports list.

Generating Reports in the PrivateArk Administrative Client

The following reports can be generated in the PrivateArk Administrative Client:

- **Safes List** – A list of Safes and their properties according to location.
- **Owners List** – A list of Owners of the specified Safe(s) and their permissions.
- **Safe Activities** – All the activities that have taken place in a particular Safe(s) during the specified period.
- **Active/Non-active Safes** – A list of active or non-active Safes for activities over a specified period of time. The report includes a list of active or non-active Safes and some of their properties.

The active or non-active status of the Safe is determined by the administrative or data-related tasks that were carried out in it, and not by whether it was opened or closed.

Note: The 'Last Used' date is updated for both administrative and data-related tasks and also for Safe open and close operations.

- **Entitlement Report** – Users' entitlement rights in the EPV regarding user, Safe, active platform, target machine, target account, etc. This report includes each user's effective access control and authorization level on each account that the user has access to in the EPV.
- **License Capacity Report** – The licensed user types and objects in the Vault, the maximum number of licenses for each type or object, and the number of used licenses for each one.
- **Users List** – A list of Users and disabled Users according to their location, including the location's quota and the User's own quota.
- **User Activities** – Users' activities in the Vault, including those who have been disabled. The activities do not include data-related activities. These reports can be generated by User Managers and by the Auditor User.
- **Active/Non-active Users** – Active or non-active Users, including disabled Users, in the specified Vault over the specified period of time. Active Users are defined as those who have logged on to the Vault for whatever purpose.

To Generate Reports

1. From the **Tools** menu, select **Report**, then choose the type of Report to generate; the Report window appears.

Note: You can select any report, but it will only include the information that you are authorized to view according to your permissions.

The Report Parameters tab displays the parameters that are appropriate to the report that you are generating.

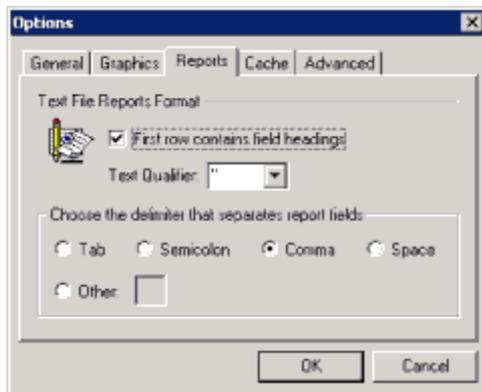
2. In the Report Parameters window, specify the information to include in the report. If you leave any parameters empty, the report will include all the information for that parameter that you are authorized to access.
3. In the History section, specify the period of time that the report will cover.
4. Select Alerts to generate a report of alerts that occurred during the period of time that you have specified.
5. Select the Report Output tab to specify the output of the report.

6. To generate the report and display it immediately, in the Show report in section, select **Microsoft Excel**.

7. Select the parameter by which to sort the information, and specify ascending or descending to determine the order in which the information will appear.
8. To generate the report and save it as a text file, select **Save Report to folder** then specify the location in which to save the report.
9. Click **OK**.
 - Reports in Excel are displayed immediately. Alerts appear in red for easy identification.
 - When a text report has been generated, a message will appear to inform you that the report generation is complete.

To Define the Text Report Format

1. From the **Tools** menu, select **Options**, then select the Reports tab; the Reports window appears.



2. To include column titles in the report, select **First row contains field names**.
3. Select the Text Qualifier to separate the column titles.
4. Choose the delimiter to separate the report fields, then click **OK**.

Adding and Managing Files and Documents

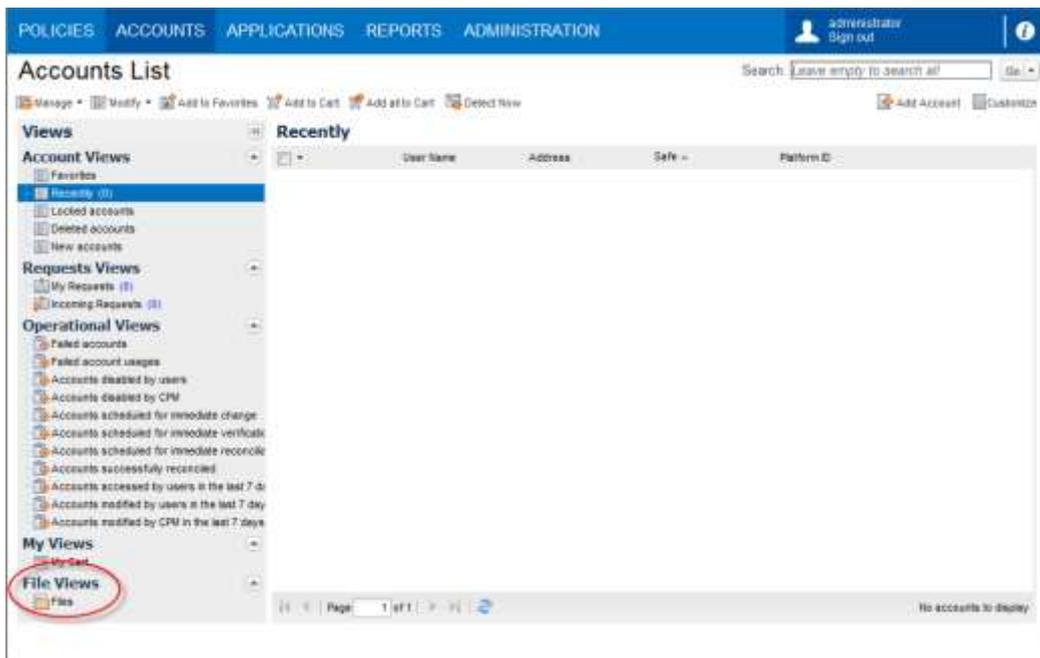
Various types of files are frequently required to support password management, as follows:

- Files that enable passwords to be used, such as certificates or key files.
- Files that are related to one or more passwords, such as organizational policies, recovery documents, and lists of devices that are managed by the Enterprise Password Vault.
- Other files that are related to the IT environment and the Password Vault implementation, such as network diagrams.

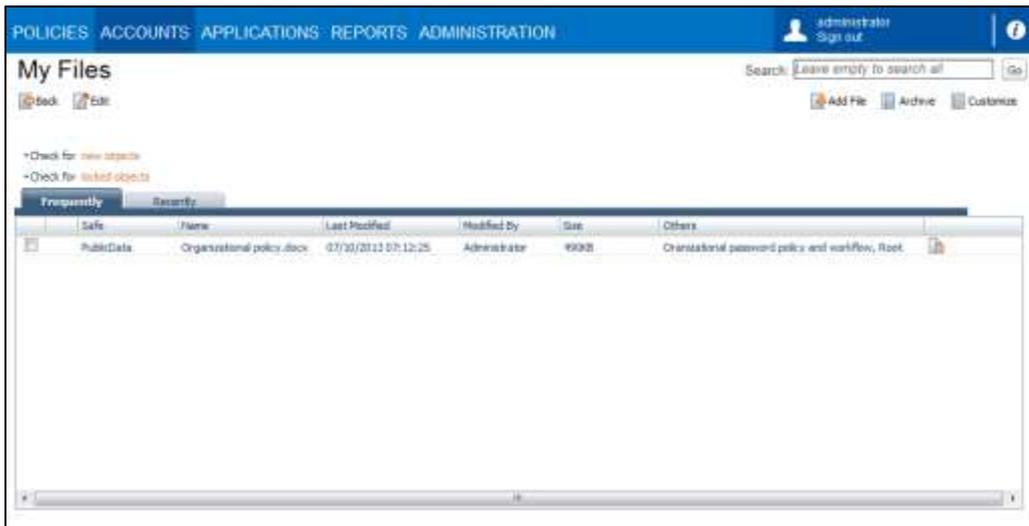
Through the Password Vault Web Access, you can store these files in the Privileged Account Security and access them whenever required. You can also inspect the activities that have taken place in the file and view different file versions. In addition, you can send a link to a specific file to any user who is authorized to access it.

To Access Files

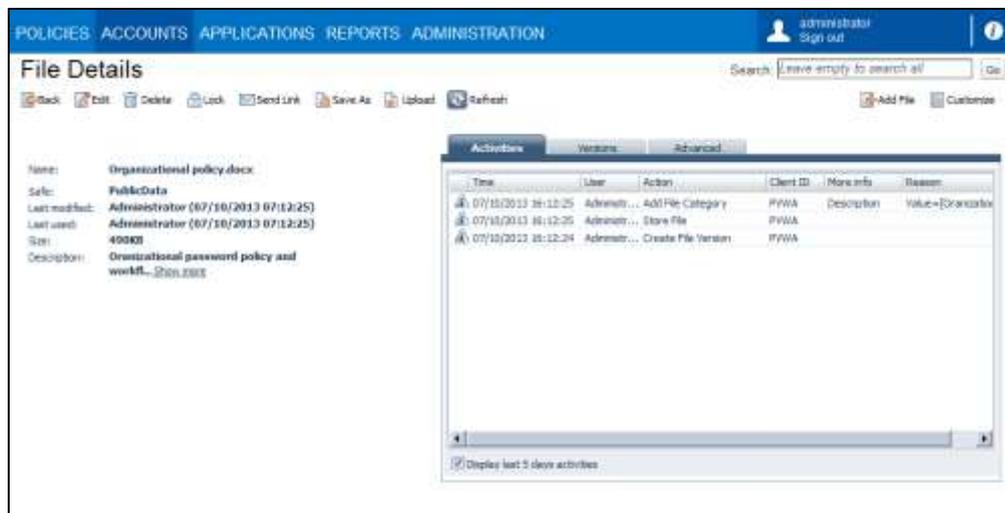
1. In the Accounts List, click **Files**.



The My Files page appears and displays a list of files that you are authorized to access.



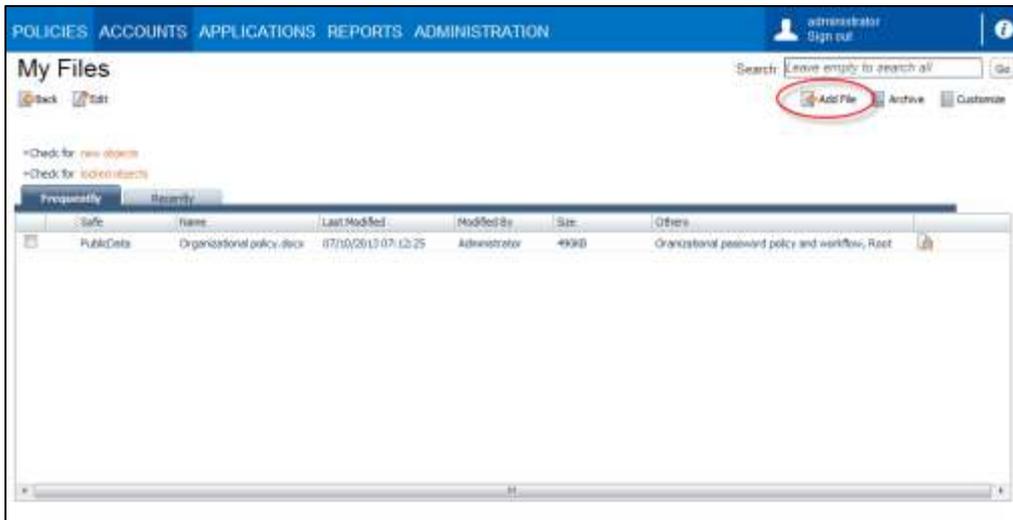
2. Download the file or open it to view.
 - i. Click the file to view; the File Details page appears.



- ii. On the toolbar, click **Save As** to save the file in a different location, or click any other button for other actions.
- or,
- i. In the line of the file to view, click **Save File**; the File Download window appears.
 - ii. Click **Open** to open the file,
 - or,
 - Click **Save** to save the file in a different location.

To Add a New File

1. In the Accounts List, click **Files**; the My Files page appears.
2. Click **Add File**.



The Add File page appears.

The screenshot shows the 'Add File' form. At the top, there are 'Save' and 'Cancel' buttons. The form has the following fields:

- 'Store in Safe': A dropdown menu with '(select)' as the current value.
- 'Description': A text input field.
- 'Keywords': A text input field with a note below it that says 'Keywords separated by semicolon'.
- 'Local File': A text input field with a 'Browse...' button next to it.

 At the bottom of the form, there are 'Save' and 'Cancel' buttons.

3. Select a Safe where the file will be stored.
4. If necessary, specify a description of the file and one or more keywords, separated by commas.
5. Click **Browse** and select the file to store in the Safe, then click **Save**; the file is uploaded to the Safe and a message appears confirming that the file was uploaded successfully.

The Central Policy Manager

This chapter guides you through management processes for the Central Policy Manager (CPM).

This chapter includes the following sections:

- Administrating the Central Policy Manager
- Configuring Accounts for Automatic Management
- Configuring Service Accounts

Administering the Central Policy Manager

Managing the Central Policy Manager

The CPM is installed on a Windows system as an automatic system service called CyberArk Password Manager.

It can be stopped and started through the standard Windows service management tools.

To Stop the CyberArk Password Manager Service

1. From the **Start** menu, select **Settings**, then **Control Panel**.
2. From the list of Control Panel options, select **Administrative Tools**, then **Services**; the Services window appears.
3. Right-click **CyberArk Password Manager**, and select **Stop**.

To Start the CyberArk Password Manager Service

1. From the **Start** menu, select **Settings**, then **Control Panel**.
2. From the list of Control Panel options, select **Administrative Tools**, then **Services**; the Services window appears.
3. Right-click **CyberArk Password Manager**, and select **Start**.

Replicating Vault Server Passwords

As the CPM requires credential files to authenticate to the Vault, it is essential that the credential files in the DR Vault are always identical to those on the Production Vault. In order for the CPM to access the Vault and continue working seamlessly in a Disaster Recovery situation, its new credentials must be replicated to the DR Vault whenever they are changed.

This is configured by the following parameter in the CreateCredFile utility:

- **DisableSyncPasswordToDR** – Whether or not passwords in user credential files will be replicated to all DR sites before they are replaced. The default value of this parameter is 'No', which makes sure that user credential files on all DR sites (if they exist) are synchronized with the Production Vault and that users will be able to continue working with the Vault seamlessly after a failover. If this parameter is changed to 'Yes', passwords will be replaced in credential files regardless of whether or not they have been replicated to all DR sites.

Configuring the Central Policy Manager

In addition to platforms, the CPM has its own configuration settings. This includes general parameters for the CPM, and extra parameters related to log files and email **notifications**. The configuration file containing the setting (cpm.ini) is created automatically during setup and stored in the Root folder of the <username> Safe, by default called the 'PasswordManager' Safe. Users can configure it through the ADMINISTRATION page.

To Configure the CPM in the PVWA

1. Log onto the PVWA with an Administrator user.
2. Click ADMINISTRATION to display the **System Configuration** page, then in **Central Policy Manager**, select **CPM Settings** for the CPM to manage; the CPM settings page is displayed.
3. In the **General** parameters, specify the following parameters:
 - **Interval** – Specify the number of minutes after which the CPM re-reads the list of platforms, in order to handle new platforms or remove deleted ones.
 - **Email parameters** – Specify the following email parameters so that the CPM can send error notifications to defined recipients. For more information, refer to *CPM Log Errors Email Notifications*, page 376.
 - **NotifyPeriod** – The minimal interval in hours between email notifications.
 - **NotifyOnlyOnError** – Whether or not to send only error notifications.
 - **AdminEmailAddress** – The email address where email notifications will be sent.
 - **SmtptServer** – The IP address of the SMTP server.
 - **SenderAddress** – The email address where the email is sent from.
 - **Subject** – The subject title of the email.
 - **Log parameters** – Specify the following log parameters so that the CPM can save log files and upload them into the Vault. For more information, refer to *CPM Activity Logs*, page 372.
 - **LogCheckPeriod** – The interval in hours after which the log files will be uploaded to the Vault. After the log files are uploaded to the Vault, they are deleted from the CPM machine. This is relevant to the pm and pm_error log files. ThirdParty logs are not uploaded to the Vault and are copied to the Logs\Old\ThirdParty folder based on this interval.
 - **LogSafeFolderName** – The full name of the folder in the Safe where the log files will be saved.
 - **LogSafeName** – The name of the Safe where the log files will be saved.
 - **Events parameters** – Specify the following Events parameters so that the Password Vault Web Access will be able to display information about the CPM.
 - **WriteStartCycleEvent** – Whether or not the CPM will write an 'I'm alive' event each time it reads platforms from the CPM Safe. These events are written to the PasswordManager_Info Safe.
 - **LogPasswordEvents** – Whether or not the CPM will write a corresponding event each time it changes, verifies, or reconciles a password.
 - **CopyPoliciesToCPMInfoSafe** – Whether or not the CPM will copy platform files from the CPM Safe to the CPM information Safe each time it reads these files, so that they can be viewed by users in the PVWA.

- **DisableExceptionHandling** – How the CPM will function when the system stops suddenly.
 - When this parameter is set to Yes, the CPM will pass control of exception handling to the operating system, resulting in crash dumps. This is the default value.
 - When this parameter is set to No, the CPM will log a system crash, but will not pass control to the operating system.
 - **Auto-detection parameters** – Specify the following auto-detection parameters to determine how the CPM will manage auto-detection processes.
 - **ADPoolSize** - The size of concurrent automatic detection processes being executed. Restart the CPM to apply this parameter.
 - **AllowManualRequests** –Whether or not CPM will search for auto-detection processes initiated manually by users.
 - **ManualRequestsInterval** – The time interval in minutes between searches for auto-detection processes initiated manually by users.
 - **ManualRequestsRecoveryStartTime** – The number of retroactive hours to search for auto-detection processes initiated manually by users.
4. Click **Apply** to apply the new configurations.

CPM Activity Logs

All activities that are carried out by the CPM are written in a log file and stored in the Log subfolder of the Password Manager installation folder from where they can be copied into a Safe. The frequency of the upload and the files' location in the Safe is specified in the CPM parameters file.

The following two log files contain the activities of the CPM:

- **pm.log** – This file contains all the log messages, including general and informative messages, errors, and warnings.
- **pm_error.log** – This file contains only warning and error messages.

Storing Log Files in the Vault

All the CPM log files can be automatically uploaded to a Safe in the Vault on a regular basis, according to a predefined period of time in the CPM parameters file. Each time a log file is uploaded to the Vault, it is copied to the History subfolder of the Log folder, and the CPM begins writing to a new log file.

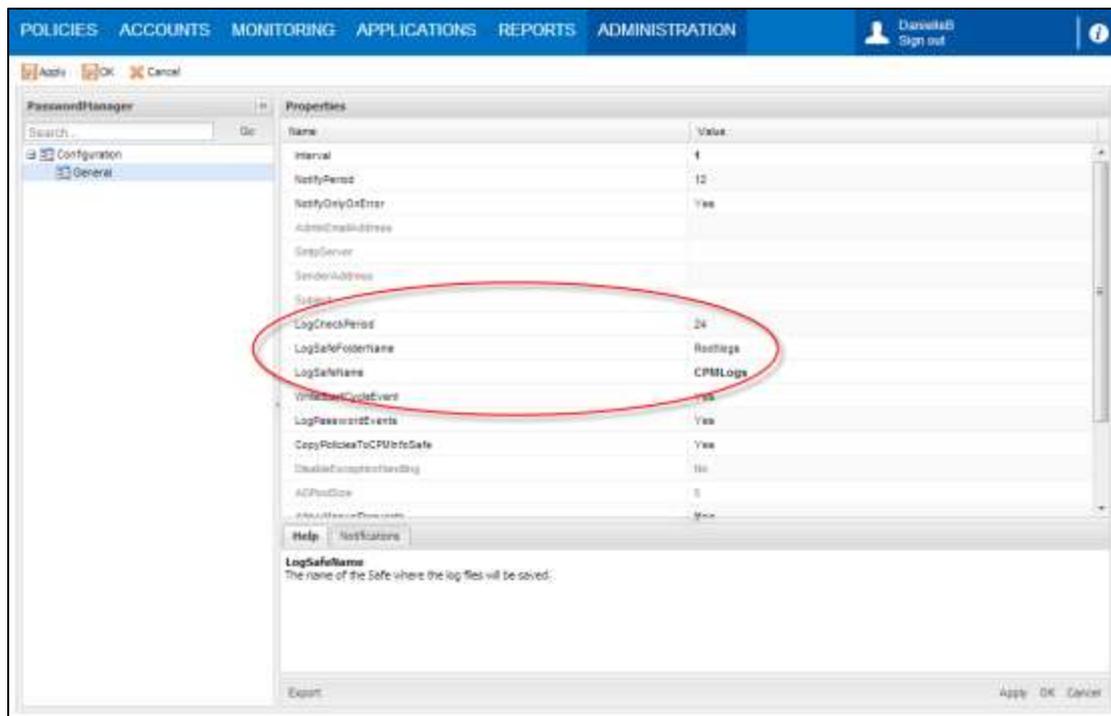
The log files can be stored in any Safe by the CPM user, who requires the **Create Passwords/Files** and **Delete Passwords/Files** authorizations in the specified Safe.

1. Click **ADMINISTRATION** to display the System Configuration page, then in **Central Policy Manager**, select **CPM Settings** for the CPM to manage; the CPM settings page is displayed.

2. Specify the following Log file properties so that the log files can be copied into the Safe.

- LogSafeName
- LogSafeFolderName
- LogCheckPeriod

For example, you could create a Log folder in the 'CPMLogs' Safe, and upload the log files into this folder every 24 hours. In this case, the CPM log properties file would look like this:



Accessing Old Log Files in the Vault

By storing log files in the Vault, you can access old log files at any time using the Vault's version feature. For this reason, it is recommended to save the log files in a Safe that keeps file versions for a long period of time.

1. Open the Safe where the log files are stored, and display the Safe Properties window.
2. In the History tab, specify that versions in the Safe will be kept for at least 14 days, or longer.

ThirdParty Log Files

Errors that occur during the password change process on the remote machine are saved in a log file and stored in the Log folder. This file specifies the date and time when the error occurred, and a detailed error message that specifies why the error occurred. This file is generated by the CPM built-in password generation plug-ins.

The name of the log file comprises the following details, which enable you to identify which account generated an error.

`<type of password>-<Safe>-<folder>-<name of account>.log`

For example, the following ThirdParty log file name would contain details about an error that occurred on an account for a **Unix** machine, stored in the **R&D Accounts Safe**, under the **Root** folder, where the account is called **Operating System-UnixSSH-1.1.1.250-Root**:

UnixSSH-R&D Accounts-Root-Operating System-UnixSSH-1.1.1.250-Root.log

History Log Files

After the time period specified in the **LogCheckPeriod** parameter, pm.log and pm_error.log files are moved into the Old subfolder of the Logs folder, and ThirdParty log files are moved into the ThirdParty subfolder of the Old subfolder.

The file is marked with a time stamp and renamed as follows:

`<filename> (<date>-<time>).log`

For example, the log files that were copied to the Safe on January 1st, 2013, at 6.00am, will be renamed as follows:

- pm.log will be renamed to **pm (20130101-060000).log**
- pm_error.log will be renamed to **pm_error (20130101-060000).log**

Deleting Log Files

In order to keep the log files on the local drive to a minimum, the log files that have already been copied to the Safe can be deleted regularly. The DeleteFiles utility, **deletefiles.exe**, is included in the CPM installation folder for this purpose.

Note: This is relevant only to the pm*.log and pm_error*.log files in the Old subfolder of the Logs folder. ThirdParty log files in the ThirdParty subfolder of Logs\Old are deleted automatically, based on the value configured in the **OldLogRetention** parameter described in *Enhanced CPM Logging*, page 375.

The DeleteFiles utility uses the following syntax:

```
DeleteFiles <NumOfDays> <FolderName1> [FolderName2] [/? | /h]
```

Parameter	Specifies
NumOfDays	The number of days after creation that files cannot be deleted.
FolderName1	The name of a folder whose contents will be deleted.
FolderName2	The name of a second folder whose contents will be deleted. This parameter is optional.
/?	Lists the available options.

It is recommended to run this utility on the two subfolders of the Logs folder under the Password Manager installation folder.

This utility can be scheduled to run automatically. However, as this utility deletes files, and they cannot be used afterwards, make sure that the parameters specified in the utility are correct. All activities that are carried out by this utility are written to a log file, `deletefiles.log`, in the folder where the utility is run.

Enhanced CPM Logging

The CPM creates the following log files to monitor CPM activity and status in addition to the log files described above.

- **PMConsole.log** – This file contains informational messages about the CPM, such as ‘Server is starting’ and ‘Server is shutting down’. This log is meant for system administrators who monitor the status of the CPM. Errors that refer to CPM function and user authentication are included in this log.
- **PMTrace.log** – This file contains errors and trace messages. The types of messages that are included depend on the debug levels specified in the main configuration file in the following parameters:
 - **CPMDebugLevels** – Sets the debug level of the CPM. You can set several values, separated by commas.

Debug level	Indicates
0	No messages will be written to the trace log.
1	CPM exceptions will be written to the trace log. This is the default debug level.
2	CPM trace messages will be written to the trace log.
3	CPM CASOS activities will be written to the trace log.
4	CPM CASOS debug activities will be written to the trace log.
5	CPM CASOS errors will be written to the trace log.
6	All CPM CASOS activities and errors will be written to the trace log.

When these log files reach 25 MB, they are moved to the ‘old’ subfolder of the Logs folder. New log files are started automatically.

The following parameter determines when old log files will be deleted.

- **OldLogRetention** – The number of days that trace, console and ThirdParty log files will be saved, after which they will be deleted. By default, log files are saved for **seven** days. To prevent old files from being deleted, specify **0** (zero). For more information, refer to the Log File parameters in *CPM Settings*, in the Privileged Account Security Reference Guide.

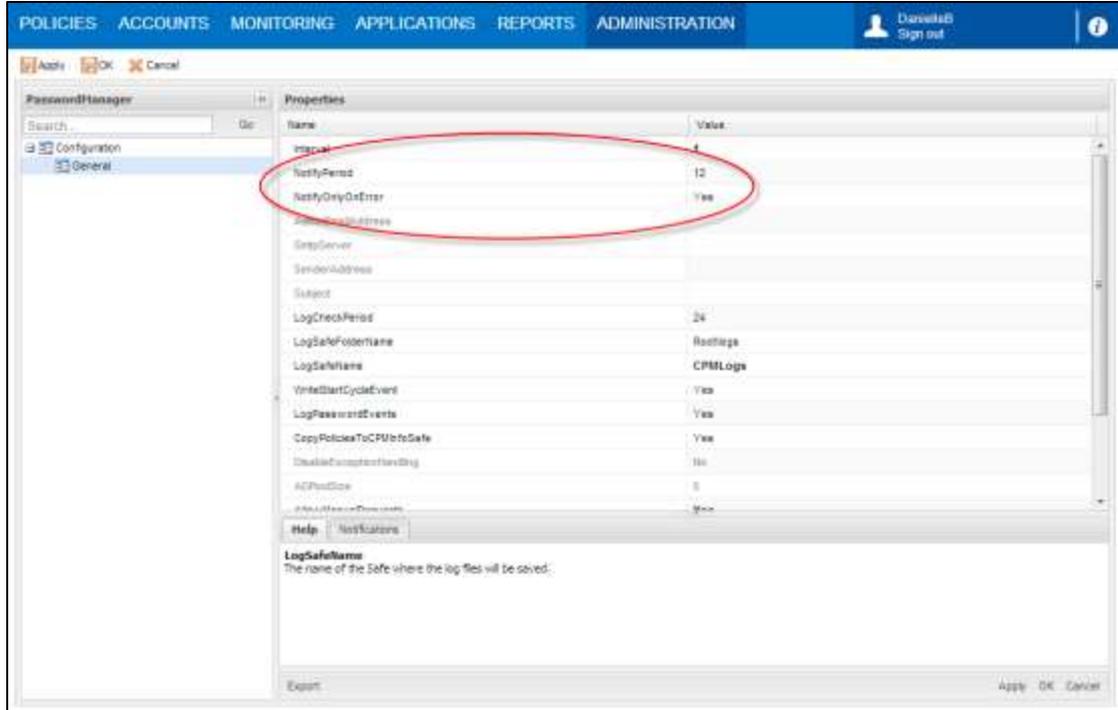
CPM Log Errors Email Notifications

Regular notifications of activity in the CPM can be sent to a specified email address, according to a predefined period of time. These notifications can either be general or a collection of warnings and errors that are saved in the log files.

These configurations are specified in the Central Policy Manager Settings in the System Configuration page by the following parameters:

- SMTP Server parameters:
 - **SmtplibServer** – The IP address of the SMTP server.
 - **SenderAddress** – The email address where the email is sent from.
 - **AdminEmailAddress** – The email address where email notifications will be sent.
 - **Subject** – The subject title of the email.
- **NotifyPeriod** – The number of hours between notification messages.
- **NotifyOnlyOnError** – Specifies the type of notification that will be sent:
 - A general message will be sent to indicate that the service is working successfully. This message will contain the same content as the record that is written in the pm.log.
 - Notifications of errors and warnings that occur during account management procedures will be sent. This message will contain the same content as the record that is written in the pm_error.log. These notifications are only sent once.

For example, to receive notifications of errors and warnings every hour, specify the following parameters:



To receive both types of notification hourly, change **NotifyOnlyOnError** to **No**.

Configuring Accounts for Automatic Management

You can configure accounts for automatic management by specifying parameters either in the platform or as an account property. These parameters can be specified by authorized users in the ADMINISTRATION page of the PWWA. For more information about configuring platforms, refer to *Managing Target/Service Account Platforms*, page 108.

Applying Platforms

- **Limiting Platforms to Specific Safes** – Platforms can be restricted to specific Safes, according to the **AllowedSafes** parameter in the **General** section of the platform:

This feature is especially relevant if you implement the new reconciliation functionality to prevent automatic reconciliation being performed on every Safe and giving unauthorized users access to passwords.

In large-scale environments, it is very important to enable the CPM to focus its search operations on specific Safes, instead of scanning all Safes it is allowed to see in the Vault.

For example,

- To limit this platform to Safes called 'LinuxPasswords' and 'AIXPasswords', specify the following: **AllowedSafes=(LinuxPasswords)|(AIXPasswords)**
- To apply a platform on all Safes, specify **AllowedSafes=.***. This is the default value.

Managing Password Change Processes

Activating a Password Change Process

The CPM changes passwords according to the following **Password Change** parameters. For a full list of platform parameters, refer to the Privileged Account Security Reference Guide.

- **Expiration period** – Passwords in accounts marked with the **ExpirationPeriod** parameter are changed automatically after a predetermined period of time. This expiration period search process is based on the **Interval** parameter in the applied platform.
- **Starting the change process before the expiration period elapses** – The CPM can initiate a password change process before the scheduled time that is specified in a platform. The **HeadStartInterval** parameter determines the number of days before the account's expiration that the CPM will initiate a password change process. If, for any reason, a password cannot be changed, the policy is not violated, and there is time to resolve any potential problems.

- **Manual or Automatic check-in (exclusive and one time account mode) –** When the Master Policy enforces check-in/check-out exclusive access, passwords are changed when the user clicks the Release button and releases the account. This is based on the **ImmediateInterval** parameter in the applied platform. If the user forgets to release the account, it is automatically released and changed by the CPM after a predetermined number of minutes, defined in the **MinValidityPeriod** parameter specified in the platform.
- **One-time account (without exclusive mode) –** When the Master Policy enforces one-time password access, passwords are used once, and then changed by the CPM after a predetermined number of minutes, defined in the **MinValidityPeriod** parameter specified in the platform.
 - **Changing passwords after the request timeframe –** When the Master Policy enforces check-in/check-out exclusive access or one-time password access, passwords that can only be accessed after a dual control request that contains a specified timeframe has been confirmed can be changed automatically by the CPM after the timeframe has expired, according to the **PasswordLevelRequestTimeframe** parameter. This parameter overrides the **MinValidityPeriod** parameter and is not relevant if the platform is for a group. This parameter is only relevant when the Master Policy enforces dual control password access approval.
- **Immediate password change –** You can change one or more passwords immediately in the PVWA, regardless of whether they have been used or reached their expiration period. This is based on the **ImmediateInterval** parameter in the applied platform. For more information about changing password values in the Password Vault Web Access, refer to *Changing Passwords that are Managed Automatically by the CPM*, page 198.
- **Changing passwords on specific days –** Password change processes can be restricted to specific days. This means that the CPM will only change passwords on the days of the week specified in the **ExecutionDays** parameter. The days of the week are represented by the first 3 letters of the name of the day. Sunday is represented by Sun, Monday by Mon, etc.
- **Enforcing password platforms for passwords that are specified manually –** You can enforce a predefined platform and ensure that only characters that meet the password complexity requirements are specified. The **EnforcePasswordPolicyOnManualChange** parameter determines whether or not platform rules will be enforced for manual password changes so that end-users will not be able to set non-compliant passwords. Specify either **Yes** or **No**. The default value is **Yes**.
- **Preventing users from specifying previous password values –** You can control the number of previous password values that users cannot specify when they change a password value manually. The **EnforcePasswordVersionsHistory** determines the number of previous password values that are stored in the Vault and cannot be specified. Valid values are between **1** and **50**, and the default value is **7**. Specify **-1** to disable this feature.

Verifying Passwords

The CPM verifies passwords according to the following **Password Verification** parameters:

- **Initiating password verification manually** – Password verification processes can be initiated manually by users in the PVWA on passwords marked with the **VFAllowManualVerification** parameter.
- **Verifying passwords automatically** – The CPM will automatically start a password verification process on passwords marked with the **VFPerformPeriodic Verification** parameter, according to the number of days specified in the **VFVerification Period** parameter.
- **Predefined Time Period** – When the Master Policy enforces password changes at predefined intervals, you can specify the timeframe during which passwords will be verified using the **VFFromHour** and **VFToHour** parameters. Before or after this time period, the CPM can either verify passwords automatically or in response to a user's action after the specified number of days have passed.
- **Verifying Passwords on Specific Days** – Password verification can be restricted to specific days. This means that the CPM will only verify passwords on the days of the week specified in the **VFExecutionDays** parameter. The days of the week are represented by the first 3 letters of the name of the day. Sunday is represented by Sun, Monday by Mon, etc.

Reconciling Passwords

The CPM reconciles passwords according to the following **Password Reconciliation** parameters:

- **Initiating password reconciliation manually** – Password reconciliation processes can be initiated manually by users in the PVWA on passwords marked with the **RCAAllowManualReconciliation** parameter.
- **Synchronizing passwords automatically** – The CPM will automatically reconcile passwords marked with the **RCAutomatic ReconcileWhenUnsynced** parameter after it detects a password on a remote machine that is not synchronized with its corresponding password in the Vault.
- **Automatic reconciliation process** – A reconciliation process will be launched automatically in response to the CPM plug-in error codes that are represented in the **RCReconcileReasons** parameter.
- **Predefined Time Period** – You can specify a timeframe during which passwords will be reconciled. Passwords marked with the **RCFromHour** and **RCToHour** parameters will be reconciled during the specified period. After this hour, the CPM can either reconcile passwords automatically or in response to a user's action after the specified number of days have passed.
- **Reconciling Passwords on Specific Days** – Password reconciliation can be restricted to specific days. This means that the CPM will only reconcile passwords on the days of the week specified in the **RCExecutionDays** parameter. The days of the week are represented by the first 3 letters of the name of the day. Sunday is represented by Sun, Monday by Mon, etc.

- **Skipping Reconciliation Processes** – The CPM can skip the reconciliation process and disable accounts if they are not linked to a reconciliation account at either platform level or account level, in platforms marked with the **IgnoreReconcileOn MissingAccount** parameter. This avoids errors being generated by the failed retries operations until the account is disabled by the CPM. An informative message is written to both the PM.LOG and PM_ERROR.LOG to record the reason why the reconciliation was not performed. This is relevant for reconciliation processes that are initiated manually in the PVWA, as well as processes that occur automatically.
- **Reconciliation Account Password** – A specific reconciliation account password is used to reset passwords that are reconciled. The reconciliation account password can either be defined specifically or dynamically, as described below:

- **Specific account** – You can specify an account that contains the password that will be used to reset passwords when they are reconciled, with the following parameters:

- ReconcileAccountSafe
- ReconcileAccountFolder
- ReconcileAccountName

Typically, reconcile account passwords are defined this way for Windows accounts to specify a domain account that has the appropriate privileges to reconcile local or domain passwords on multiple machines.

- **Dynamic rule** – Instead of specifying a particular reconciliation account password, you can define a dynamic rule that uses password property values to identify a relevant reconciliation account password. A dynamic rule can be specified either at platform level or at account level.

Typically, passwords in Unix environments are defined dynamically. The rule defines a naming convention that the CPM identifies and uses to match the relevant reconcile account for each system. This eliminates the need to link each Unix password to a relevant reconcile account on the same machine.

The following table lists the parameters that can be used to define a dynamic rule:

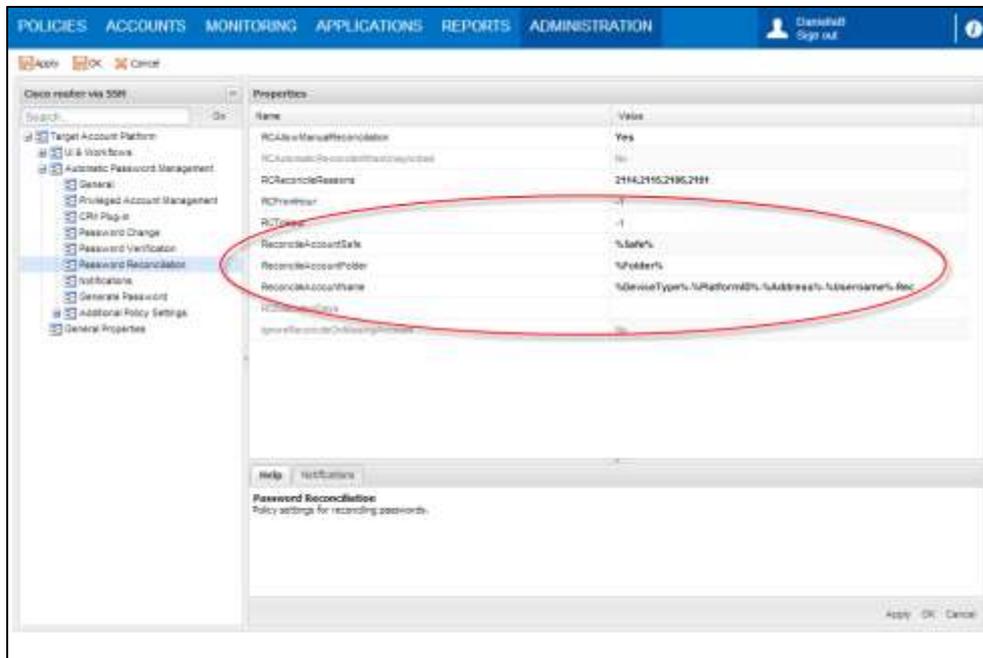
Parameter	Specifies
%Safe%	The name of the Safe where the reconcile account password is stored.
%Folder%	The name of the folder where the reconcile account password is stored.
%Name%	The name of the password object that will be used as the reconcile account password.
%<Password Property>%	The name of any password property that is defined in the reconcile account password.

The following example shows a password that will be reconciled and the dynamic rule that would define a reconciliation account password for it:

Password to reconcile:

Network Device-CiscoSSH-1.1.1.250-CiscoUser

Dynamic rule:



In this example, the platform would use a reconciliation account password in the same Safe and folder as the main password with the following name:

Network Device-CiscoSSH-1.1.1.250-CiscoUserReconcile

Alternatively, if you have configured a password name pattern in the Web Access Options settings page, you can specify '%Name%' instead of specifying all the properties individually, as follows:

ReconcileAccountSafe	
ReconcileAccountFolder	
ReconcileAccountName	%Name%Reconcile

Managing Reconciliation on Locked Accounts

In addition, the following parameter in the Additional Policy Settings prevents reconciliation processes from failing due to accounts that are locked as a result of invalid logon attempts, and ensures continuous account management:

- **UnlockUserOnReconcile** – Determines whether or not the CPM will unlock a locked target account during the reconcile operation. The default value is **No**.

Resetting Passwords using the Reconciliation Account

The following parameter in the Additional Policy Settings performs periodic password reset operations using associated reconciliation accounts, instead of a password change operation. This is configured at platform level. This feature is useful when a one-time password is used with a Directory Services minimum password-age restriction, or when the password policy prevents the user from changing their own password.

- **ChangePasswordInResetMode** – Defines whether or not password changes will be performed via reset mode using the reconciliation account. The default value is **No**.

Associating Logon Accounts

The CPM associates logon accounts to enable users to log onto remote machines where they can perform identity management tasks. Logon accounts can be configured in either of the following ways:

- **At platform level** – All accounts attached to a specific platform will use the logon account specified in the platform.
- **At account level** – A logon account can be initiated manually in the Account Details page. For more information, refer to *Linked Accounts*, page 220.

The following parameters in the **Privileged Account Management** parameters specify the default logon account that will be associated with each new account.

- **LogonAccountSafe** – The name of the Safe, or a dynamic rule that specifies it, where the default logon account that will be used for accounts associated with this platform is stored.

Note: PSM cannot access logon accounts if the Master Policy is configured to enforce dual control password access approval.

- **LogonAccountFolder** – The name of the folder, or a dynamic rule that specifies it, where the default logon account that will be used for accounts associated with this platform is stored.
- **LogonAccountName** – The name of the default logon account that will be used for accounts associated with this platform.

Setting Up Supported Platforms

The CPM supports remote password management on the following platforms, all of which are available out of the box, as well as many more:

- Applications
- Databases
- Directories
- Network Devices
- Operating Systems
- Remote Access
- Security Appliances
- Websites
- Service Accounts

The CPM can also dynamically support password management on additional remote entities. For more information, contact your CyberArk support representative.

For each of the platforms that the CPM supports, different account properties are required. Therefore, a sample platform is supplied for each of the supported platforms in the Central Policy Manager of the System Configuration page. For more information about platforms, refer to *Adding New Platforms*, page 110.

Certain platform properties and account properties are essential for the CPM to manage and use the passwords.

For information about required and optional properties for each platform, refer to the Privileged Account Security Reference Guide.

For a complete list of account properties, refer to *Appendix A: Account Properties*, page 924.

Managing Passwords in a Terminal Environment

The CPM enables organizations to manage passwords automatically on remote machines and store the new passwords in the CyberArk Password Vault, with no human intervention, according to the organizational policy.

The Terminal plug-in enables the CPM to change passwords on any environment or platform that can be accessed by a terminal connection through standard protocols such as SSH, Telnet, FTP, etc. As a result, this plug-in enables passwords to be changed on Unix machines, Cisco routers, Cisco PIX, OS/390 (Z/OS) machines, Netscreen machines, and many other platforms. Any environment that can be accessed in terminal mode is supported by this plug-in. In addition, this plug-in enables the CPM to verify passwords on remote machines through a terminal connection, and reconcile them when necessary. For more information about environments and platforms that support automatic password verification and reconciliation, refer to the relevant section below.

The Terminal plug-in responds to the output it receives from the remote machine on which passwords are changed according to a prompts file. During CPM installation, sample prompts files that contain standard prompts are supplied for every supported platform. However, some platforms are configured differently to the standard, in which case, you need to specify the exact prompts.

Operating Systems

Windows Domain Accounts

Automatic password management is supported on Windows Domain Accounts on IPv4 and IPv6.

Supported Platforms

The CPM supports remote password management for Windows Domain users on the following platforms:

- Windows 2000/2003/2008/2008R2 with Service Pack 1 Active Directory server/2012/2012R2

Configuring Automatic Management for Windows 2008

1. On the remote device, enable the following:
 - Client for Microsoft Networks
 - File and Printer Sharing for Microsoft Networks
2. Configure the user who will perform the reconciliation process:
 - Make sure that the reconciliation user is a domain user and is a member of the 'Administrators group' on the remote machine.or,
 - i. Make sure that the reconciliation user is a member of the 'Administrator's group'.
 - ii. Disable the UAC for the 'Administrator's group':
 - a. In the Local Security Policy, select **Local Policies**.
 - b. In **Security Options**, disable **User Account Control: Run all administrators in Admin Approval Mode**.

Platform

In the Platform Management page, make sure that the following target account platform is displayed:

- Windows Domain Account

Password Management Features

The CPM can change and verify Windows Domain passwords on remote machines. If a password is invalid, the CPM can generate a new password and replace the invalid password on the remote machine and its corresponding password in the Password Vault. The parameters that define these tasks are in the platform. A reconciliation account password can be specified either at platform level or at account level.

You can also configure the CPM to perform periodic password reset operations (using reconciliation accounts) instead of a password change operation. This is configured at the platform level. This feature is useful when a one-time password is used with a Directory Services minimum password-age restriction, or when the password policy prevents the user from changing their own password.

For more information, refer to *Configuring Accounts for Automatic Management*, page 377.

Note: After a predefined number of invalid attempts to manage Windows domain accounts, the CPM disables automatic management before the accounts are disabled in the remote device. Therefore, it is highly recommended to configure the main account for automatic reconciliation. If the account in the remote device is locked due to invalid logon attempts, the reconciliation process will unlock it. This is configurable in the platform's Additional Policy Settings.

Resolving Remote Machine Addresses

This plug-in can be configured to skip the step during which it resolves the specified 'address' property of the account to a domain name. In some cases, this value may contain a DNS name that is not necessarily a resolvable address (e.g. fully qualified domain name). This resolving is critical when working with an 'address' property that contains NETBIOS Domain Accounts.

To disable the address resolving in Windows domain accounts, in the `ExtraInfo` section of the relevant Windows platform, set the value of the **DisableAddressResolving** parameter to **Yes**.

For more information about this parameter, refer to the Privileged Account Security Reference Guide.

Windows Domain Service Accounts

The CPM can synchronize multiple copies of Windows domain accounts that have been changed and are used in different resources in the following services:

- Windows Services Accounts
- Windows Scheduled Tasks
- Windows IIS Application Pools Passwords
- Windows COM+ Applications
- Windows IIS Directory Security (Anonymous Access) Passwords

For more information, refer to *Managing Service Accounts*, page 151.

Additional Logon Password

If an extra password is required to log onto the machine where the Windows Domain Account is, you can add a link to the extra password that will be used to log onto the remote machine. The extra password can be a domain or a local Windows account password. If an extra password is not defined, the Windows Account password will be used to log onto the remote machine.

The user that logs onto the remote machine requires the following permissions:

- Remote access
- Reset passwords

The additional logon user's password may or may not be managed by the CPM.

For more information, refer to *Linked Accounts*, page 220.

Windows Local Accounts

Automatic password management is supported on Windows local accounts on IPv4 and IPv6.

Supported Platforms

The CPM supports remote password management for local users on Windows desktops/servers on the following platforms:

- Windows 2000./2003/2008/2008R2 with Service Pack 1/2012/2012R2
- Windows XP/Vista/Windows 7 with Service Pack 1/Windows 8

Configuring Automatic Management for Windows 2008/Vista

1. On the remote device, enable the following:
 - Client for Microsoft Networks
 - File and Printer Sharing for Microsoft Networks
2. Configure the user who will perform the reconciliation process:
 - Make sure that the reconciliation user is a domain user and is a member of the 'Administrators group' on the remote machine.or,
 - i. Make sure that the reconciliation user is a member of the 'Administrator's group'.
 - ii. Disable the UAC for the 'Administrator's group':
 - a. In the Local Security Policy, select **Local Policies**.
 - b. In **Security Options**, disable **User Account Control: Run all administrators in Admin Approval Mode**.

Platform

In the Platform Management page, make sure that both the following target account platforms are displayed:

- Windows Server Local Accounts
- Windows Desktop Local Accounts

Password Management Features

The CPM can change and verify Windows desktop/server passwords on remote machines. If a password is invalid, the CPM can generate a new password and replace the invalid password on the remote machine and its corresponding password in the Password Vault. The parameters that define these tasks are in the platform. A reconciliation account password can be specified either at platform level or at account level.

For more information, refer to *Configuring Accounts for Automatic Management*, page 377.

Note: After a predefined number of invalid attempts to manage Windows local accounts, the CPM disables automatic management before the accounts are disabled in the remote device. Therefore, it is highly recommended to configure the main account for automatic reconciliation. If the account in the remote device is locked due to invalid logon attempts, the reconciliation process will unlock it. This is configurable in the platforms' Additional Policy Settings.

DNS Target Verification

In Windows environments where DNS servers constantly change and DNS records are not always up-to-date, the CPM can verify the target machine name before running platform activity to make sure that it is consistent with the address configured in the Vault. This ensures that the CPM manages the correct target machines, preventing CPM errors and facilitating continuous account management.

This is specifically applicable to the initial management cycle for machines sourced by auto-detection or by the Password Upload Utility.

For more information about configuring platforms to resolve addresses, refer to the *VerifyMachineNameBeforeAction* parameter in the Privileged Account Security Reference Guide.

Windows Local Service Accounts

The CPM can synchronize multiple copies of Windows local accounts that have been changed and are used in different resources in the following services:

- Windows Services Accounts
- Windows Scheduled Tasks
- Windows IIS Application Pools Passwords
- Windows COM+ Applications
- Windows IIS Directory Security (Anonymous Access) Passwords

For more information, refer to *Managing Service Accounts*, page 151.

Password Auto-Detection

The CPM can automatically detect Windows machines in the enterprise directory and create and manage passwords for those machines in the Password Vault. The platform can specify an unlimited number of OUs, also known as Containers, so that the CPM search in the enterprise directory is extremely flexible and streamlined.

For more details, refer to *Configuring Automatic Provisioning*, page 390.

Notes:

If the passwords that are changed belong to local users in a Windows XP environment that is not part of the Windows domain, the security settings might be different from the machine that belongs to the Windows domain, and the password will not be changed.

The following instructions enable you to alter the security settings so that passwords can be changed successfully.

1. From the Control Panel, select **Administrative Tools**, then **Local Security Policies**; the Local Security Settings window appears.
2. Expand the **Local Policies**, then display the **Security Options**.
3. Change the value of **Network Access: Sharing and security model of local accounts** to **Classic local users authenticate as themselves**.

Note: This does not change the Windows XP security level.

4. Run the following command to check that the change has taken effect.

```
net use \\machine-name
```

5. Run the following command to check that you have the **\\machine-name\IPC\$** share on your machine.

```
net use
```

Windows Local Accounts with WMI

The CPM enables organizations to automatically reset Windows Local Account passwords on remote machines using WMI without human intervention. This plug-in enables the CPM to manage accounts on remote machines when the firewall does not permit access with file and printer sharing services or when these services are disabled.

This plug-in can also be used in cases where the password policy needs to be bypassed (using minimum password age enforcement) due to the fact that a password reset is performed (instead of a password change). In addition, this plug-in is designed in a generic manner, allowing users to execute a predefined command on remote machines (requires development by Professional Services).

Automatic password management is supported on Windows local accounts with WMI on IPv4 and IPv6.

Supported Platforms

The CPM supports remote password management for Windows Local Accounts with WMI on the following platforms:

- Windows 2000/2003/2008/2008R2/2012/2012R2 server
- Windows XP/Vista/7/8

WMI Platform Prerequisites

1. Open the firewall to enable the following:
 - Windows Management Instrumentation (WMI)
2. Make sure that the following is enabled on the remote device:
 - Client for Microsoft Networks

In order for the WMI plugin to work, if NETBIOS is disabled on the target machine, a static port for WMI must be set on the target machine. For more information, refer to Windows documentation.

Configuration

1. In the PVWA, configure the user who will perform the reconciliation process:
 - Make sure that the reconciliation user is a domain user and is a member of the 'Administrators group' on the remote machine.or,
 1. Make sure that the reconciliation user is a member of the 'Administrator's group'.
 2. Disable the UAC for the 'Administrator's group':
 - i. In the Local Security Policy, select **Local Policies**.
 - ii. In **Security Options**, disable **User Account Control: Run all administrators in Admin Approval Mode**.

Platform

In the Platform Management page, make sure that the following target account platforms is displayed:

- Windows Local Accounts WMI

Password Management Features

The CPM can change Windows desktop/server passwords on remote machines. If a password is invalid, the CPM can generate a new password and replace the invalid password on the remote machine and its corresponding password in the Password Vault. The parameters that define these tasks are in the platform. A reconciliation account password is specified either at platform level or at account level.

For more information, refer to *Configuring Accounts for Automatic Management*, page 377.

Windows Local Service Accounts

The CPM can synchronize multiple copies of Windows local accounts with WMI that have been changed and are used in different resources in the following services:

- Windows Services Accounts
- Windows Scheduled Tasks
- Windows IIS Application Pools Passwords
- Windows COM+ Applications
- Windows IIS Directory Security (Anonymous Access) Passwords

For more information, refer to *Managing Service Accounts*, page 151.

Password Auto-Detection

The CPM can automatically detect Windows machines in the enterprise directory and create and manage passwords for those machines in the Password Vault. The platform can specify an unlimited number of OUs, also known as Containers, so that the CPM search in the enterprise directory is extremely flexible and streamlined.

For more details, refer to *Configuring Automatic Provisioning*, page 390.

Configuring Automatic Provisioning

The CPM can be configured to provide automatic full life-cycle management for Unix/Linux accounts, ESX host accounts and Windows Local and domain accounts and their service accounts (such as Windows Services, Scheduled Tasks, etc). This ranges from provisioning to removal or archiving, and includes all management tasks in between, ensuring complete control and secure management.

The CPM can provision accounts automatically in the following ways:

- **Accounts Feed** – A new workflow that provides a structured process that aligns with business processes to meet your organizational standards and policies. The CPM scans an organizational network and retrieves a list of accounts that have access to its computers and their dependencies. This is the recommended workflow. For more information, refer to *Accounts Feed*, page 168.
- **Auto-Detection Processes** – The CPM retrieves a list of machines and creates a local administrator account for each machine, based on a predefined template. For more information, continue below.

Note: The Accounts Feed, which is the next generation workflow for discovering and provisioning privileged accounts, replaces auto-detection functionality. For more information, refer to *Accounts Feed*, page 168.

The CPM can automatically provision accounts from the following target machine sources:

- **Active directory or platform** – This source is used to provision any computer in directory services, such as servers, desktops, test environments, etc. Usually this source is only Windows.
- **VMWare vCenter** – This source is used specifically to provision entire server environments, such as all Unix/Linux flavors and Windows.



Accounts must not be provisioned from both sources. Make sure you specify the exact source to prevent duplicated accounts and management.

Use the following information to decide which of the three available target machine sources, best suits your needs:

- **Policy** - Use this machine source to perform a Windows service account scan on machines you have already created accounts for, and have linked them to specific platforms.
- **Active directory** - Use this machine source to provision Windows desktops and/or server machine local accounts (or service account scan) and your active directory is relatively organized by specific object names or OU structure. If you are using an AD bridge product, you can also provision local accounts for Unix/Linux machines.
- **VMWare vCenter** – Use this machine source to provision entire or certain parts of your virtual server environment or if your Active Directory is not organized. This machine source will provision local accounts for ESX/Linux/Unix/Windows and scan for Windows service accounts.

This capability provides the following features:

- **Automatic provisioning for Windows local accounts** – Your enterprise's external directory can be integrated with the Password Vault to create, update,

and remove privileged accounts automatically in the Vault for Windows machines in Windows domains.

Automatic provisioning for VMware Unix/Linux guest machines Root accounts

– Your enterprise's vCenter directory can be integrated with the Password Vault to create, update, and remove privileged accounts automatically in the Vault for Root or local accounts in VMware Unix/Linux guest machines.

- **Automatic provisioning for VMware ESX host Root accounts** – Your enterprise's vCenter directory can be integrated with the Password Vault to create, update, and remove privileged accounts automatically in the Vault for Root accounts in VMware ESX host machines. This enables you to maintain the organizational password policy across your vCenter environment.
- **Automatic provisioning for service accounts** – All local and domain service accounts (also identified as usages) can be detected and provisioned automatically in the Password Vault, where they benefit from all of CyberArk's standard account life-cycle management features. This greatly reduces administration overhead required by the IT personnel when machines are added or updated, or when existing machines are removed from the system.
- **Flag local or domain accounts used in Windows Services, Schedules tasks, etc, and are not currently managed by the Privileged Account Security solution** – Accounts that have not been used for a while and/or are not currently managed in the Password Vault can be automatically identified and flagged, prompting the Privileged Account Security solution to notify and/or automatically start managing any potential shared/privileged domain/local account that is used in a Windows Service or other Windows service account, and is not currently managed by the Privileged Account Security solution.
- **On demand automatic detection and reporting** – Users can initiate specific automatic detection processes for local and domain service accounts and generate a report of all the detected service accounts, with or without provisioning them in the Vault.
- **Auditing Automatic Detection Activities** – A record of all automatic detection activities is maintained in the Vault, and a report can be generated at any time with all these details, providing a full audit of every account and service account that is detected and/or provisioned in the Vault.

Configuring Windows Machine Auto-Detection Processes

The Privileged Account Security solution provides a streamlined and flexible way to provision new machines in Active Directory/vCenter environments, and automatically manage their accounts, as well as scan Windows machines and provision any service accounts of these accounts such as Windows services or Scheduled Tasks, as well as flag any accounts that are being used within these service accounts and are not managed in the Privileged Account Security solution.

The Privileged Account Security solution detects machines from different types of data sources: LDAP, platforms, and VMware vCenter. When a platform source is used, the CPM retrieves a list of all machine addresses linked to a certain platform. When an LDAP or vCenter source is used, Directory locations are specified in a set of rules. These rules, also known as Machine Sets, define containers in the data source that will be searched. They also specify where the accounts for detected machines will be stored in the Password Vault. The detected machines can then be scanned for specific Windows predefined service accounts. Any related detected service accounts is then provisioned in the vault and associated with the relevant

Windows account, so it will later be synchronized whenever the password of this account is changed.

The automatic detection process will automatically reflect any change in the domain environment such as new added machines or removed machines, eliminating any overhead of manual updates.

There are several types of high level auto-detection functions available:

- **Windows local administrator account provisioning** – Based on an LDAP/vCenter query, the CPM will retrieve a list of machines and will create a local administrator account for each machine, based on a predefined template.
- **Windows local administrator group report (applicable only in simulate mode)** – Based on an LDAP query, the CPM will retrieve a list of machines and query the local administrator group members (Local users, Domain users and Domain groups) for each machine that it identifies, then list the results in an Excel report.
- **Windows Service Account scan** – Based on the machine source configured (LDAP/Platform/vCenter), the CPM will retrieve a machine list, and will scan for a list of platform configured service accounts (Windows Services, COM+ applications, IIS anonymous user, Windows scheduled tasks, and Windows IIS Application Pools Accounts) for each machine. Newly detected service accounts will be updated in the master account or new master accounts will be generated based on the predefined template.
- **VMware Based ESX Root Host Account Provisioning** – Based on the defined machine set, the CPM will query the vCenter directory and create a root user account for each ESX host machine, based on a predefined template.
- **VMware Based Unix/Linux Root Guest Account Provisioning** – Based on the defined machine set, the CPM will query the vCenter directory and create a root user account for each Unix/Linux guest machine, based on a predefined template.

Each automatic detection process comprises the following high level sections:

- Configuration for machine detection, based on an existing platform (policy), Active Directory, or vCenter scanning, including relevant directory details, such as: connection details, directory based context, search filters, containers, etc.
- Configuration for the types of accounts to provision, including predefined templates for local accounts, and definition for archiving accounts
- Configuration for scanning machines for Windows service accounts, including predefined templates for provisioning those service accounts in the Vault.
- Configuration for receiving different types of notifications related to the automatic detection process

If an error occurs during an auto-detection process that could damage existing accounts, the auto-detection mechanism will stop the current cycle and write an error to the log file. This enables the system administrator to check the specified parameters and connection and find out why no information was received from the Active Directory.

You can configure the PWA to onboard newly discovered usages as disabled for automatic CPM Management.

- The **ADUsageDisabled** parameter defines whether the auto-detection process will upload dependencies as disabled for automatic CPM Management.
 - **Yes** – The auto-detection process will upload dependencies as disabled for automatic CPM Management. You can enable the dependencies manually after they have been onboarded. For further information about enabling and disabling Automatic Account Management, refer to *Disabling Automatic Account Management*, page 212.
 - **No** – All dependencies will be enabled after they have been onboarded. Default is **No**.

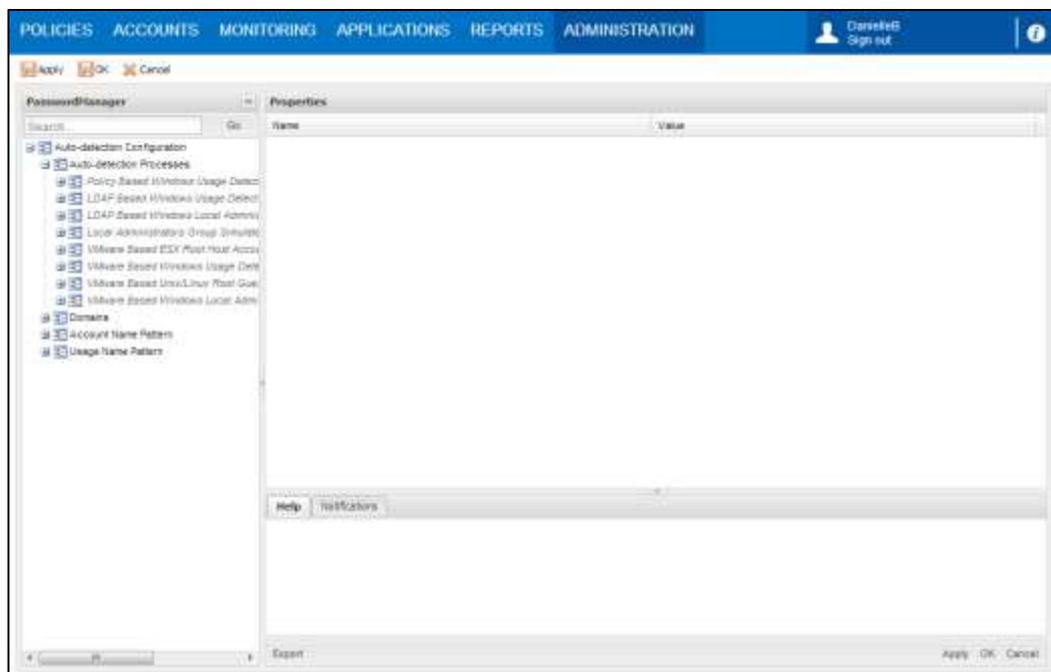
For clean installation the default is Yes and for upgrades the default is No.

Using the built-in auto-detection sample file to simplify initial configuration and usage

The built-in auto-detection sample files simplify the initial configuration and guide you through the auto-detection configuration process in a simple and intuitive manner. The sample processes should be used in conjunction with a Password Upload Utility CSV file to help pre-create the Safe account and template environment needed for each auto-detection sample process. For more information about the Password Upload Utility, refer to *Implementing the Password Upload Utility*, page 793.

For more detailed information of all auto-detection process settings, refer to *To Configure an Auto-Detection Process to Detect Machines*, page 403.

1. Click **ADMINISTRATION** to display the System Configuration page, then click **Auto-Detection**; the Auto-detection settings page for the CPM is displayed.



2. Expand Auto-detection Processes to view the defined auto-detection processes.

By default, the following sample auto-detection processes are included in the installation:

- **Policy-based Windows service account detection** – Defines a platform-based auto-detection process that automatically provisions Windows Domain accounts and service accounts.
- **LDAP-based Windows usage detection** – Defines an LDAP-based auto-detection process that automatically provisions Windows Domain accounts and service accounts.
- **LDAP-based Windows Local administrator account provisioning** – Defines an auto-detection process that automatically provisions Windows Local accounts and service accounts.
- **Local administrators group simulate report** – Defines an auto-detection process that generates a report that lists all local administrators group members on the target machines. This auto-detection process can be only run in Detect Now simulate mode.
- **VMware Based ESX Root Host Account Provisioning** – Defines an auto-detection process that automatically provisions VMWare-based ESX root host accounts.
- **VMware Based Windows Usage Detection** – Defines an auto-detection process that automatically provisions VMWare-based Windows service accounts.
- **VMware Based Unix/Linux Root Guest Account Provisioning** – Defines an auto-detection process that automatically provisions VMWare-based Unix/Linux root guest accounts.
- **VMware Based Windows Local Administrator Guest Account Provisioning** – Defines an auto-detection process that automatically provisions VMWare-based Windows Local accounts.

Use one of these sample processes as a template for your own auto-detection process. Use copy/paste to create a new process, rename it, and then customize it to meet your specific needs.

3. Select the auto-detection process that best fits your needs. These instructions show how to configure the new process using the built-in sample processes.
4. Right-click the auto-detection process and select **Copy**.
5. Right-click **Auto-detection Processes** and select **Paste**.
6. In the **ADProcessName** property, specify a unique name that describes the purpose of this process.
7. Right-click the value of the **ADProcessID** property and select **Revert to default**. This will clear the property value and a new value will be automatically generated after the changes are saved.

Note: After the changes have been saved and the process ID has been generated, **do not** edit the ID again. This will prevent the auto-detection process from running as configured.

8. By default, the **ADProcessActive** property is set to **No**. Do not change this value now. **After** you have finished defining the new auto-detection process, you will change this value to **Yes**.

9. Click **Apply** to save the new parameter values and stay in the Auto-detection Configuration page,
or,
Click **OK** to save them and return to the System Configuration page.

Example of how to use the Auto-detection sample (VMware vCenter):

The auto-detection process samples are designed to include all possible default parameters for each process. To configure a specific parameter related to the enterprise environment, replace the parameter displayed in {} curly brackets (both in the CSV file and the process). For example, change {Enter address here} to **1.1.1.112**.

VMware Based Unix/Linux Root Guest Account Provisioning

Note: In order to enable the CPM to successfully provision accounts from VMWare vCenter environments, Before initiating an auto-detection process to provision VMWare-based accounts, make sure that the guest virtual machine is running and that the required VMWare tools are installed.

This auto-detection process will provision Root accounts for all guest machines for Linux Guest and Solaris Guest types.

1. In the process configuration, expand the **Machine Detection** parameters.
2. Expand VMWarevCenter Detection, then display the Machine detection connection details properties.
3. Specify the details of the bind account that will be used to detect machines, using the following parameters:
 - **ADConnectionAccountSafe** – The name of the Safe where the account that contains credentials and information that is required to perform machine detection is stored.
 - **ADConnectionAccountFolder** – The folder where the account that contains credentials and information that is required to perform machine detection is stored.
 - **ADConnectionAccountObject** – The name of the account that contains credentials and information that is required to perform machine detection.

Note: Use one of the following account types:

- A VMWare vCenter local account with the required permissions in the VMWare vCenter environment,
 - or,
 - A domain account of the VMWare vCenter domain with the required permissions in the VMWare vCenter environment.
4. In the **ADMachineDetectionTargetAddress** parameter, specify the DNS address of the VMWare vCenter Server.
 5. This step is optional: Define the provisioning of more specific accounts based on the vCenter Inventory Browser.
To access the browser:
 - i. In the **VMWarevCenter Detection** section, expand the **Machine Sets** parameters.

- ii. Specify the browser's base path in one of the following ways:
 - In **Machine Sets**, in the **ADDefaultBasePath**, specify the base path of the vCenter inventory browser,
 - or,
 - Expand the **Machine Sets**, then in the relevant Machine Set, in the **ADBasePath**, specify the base path of the vCenter inventory browser.

For information about applying additional filtering for Machine Sets, refer to *To define a vCenter detection method*, page 404.

6. In the process configuration, select **Notifications**; the Notification parameters are displayed.
7. In the **NFNotifyOnErrorsRecipients** parameter, specify the email recipients that will be notified about auto-detection errors that occur during this process.
8. Click **Apply** to save the new parameter values and stay in the Auto-detection Configuration page,

or,

 Click **OK** to save them and return to the System Configuration page.

These changes will be applied the next time the CPM refreshes the configuration, according to the value of the **ADReloadInterval** property under in the auto-detection configuration.

9. Open the CSV file that you will use in the Password Upload Utility, and specify the following parameters for the **ADVMConnectionAccount** account which will be used to connect to the VMWare vCenter environment and retrieve relevant information for the auto-detection provisioning process:
 - Password parameter
 - Address parameter
 - UserName parameter

This user must have the Administrator role on the VMWare vCenter or be a member of a group with the Administrator role.

10. Run the Password Upload Utility with the modified CSV file to create the Safe, accounts and template environment. For more information, refer to *Running the Password Upload Utility*, page 801.

VMware Based ESX Root Host Account Provisioning

This auto-detection process will provision Root accounts for all ESX hosts linked to the VMware vCenter Server.

1. In the process configuration, expand the **Machine Detection** parameters.
2. Expand VMWarevCenter Detection, then display the **Machine detection connection details** properties.
3. Specify the details of the bind account that will be used to detect machines, using the following parameters:
 - **ADConnectionAccountSafe** – The name of the Safe where the account that contains credentials and information that is required to perform machine detection is stored.
 - **ADConnectionAccountFolder** – The folder where the account that contains credentials and information that is required to perform machine detection is stored.

- **ADConnectionAccountObject** – The name of the account that contains credentials and information that is required to perform machine detection.

Note: Use one of the following account types:

- A VMWare vCenter local account with the required permissions in the VMWare vCenter environment,
or,
 - A domain account of the VMWare vCenter domain with the required permissions in the VMWare vCenter environment.
4. In the **ADMachineDetectionTargetAddress** parameter, specify the DNS address of the VMWare vCenter Server.
 5. This step is optional: Define the provisioning of more specific accounts based on the vCenter Inventory Browser.

To access the browser:

- i. In the **VMWarevCenter Detection** section, expand the Machine Sets parameters.
- ii. Specify the browser's base path in one of the following ways:
 - In **Machine Sets**, in the **ADDefaultBasePath**, specify the base path of the vCenter inventory browser,
or,
 - Expand the **Machine Sets**, then in the relevant Machine Set, in the **ADBasePath**, specify the base path of the vCenter inventory browser.

For information about applying additional filtering for Machine Sets, refer to *To define a vCenter detection method*, page 404.

6. In the process configuration, select **Notifications**; the Notification parameters are displayed.
7. In the **NFNotifyOnErrorsRecipients** parameter, specify the email recipients that will be notified about auto-detection errors that occur during this process.
8. Open the CSV file that you will use in the Password Upload Utility, and specify the following parameters for the **ADVMConnectionAccount** account which will be used to connect to the VMWare vCenter environment and retrieve relevant information for the auto-detection provisioning process:
 - Password parameter
 - Address parameter
 - UserName parameter

This user must have the Administrator role on the VMWare vCenter or be a member of a group with the Administrator role.

9. Run the Password Upload Utility with the modified CSV file to create the Safe, accounts and template environment. For more information, refer to *Running the Password Upload Utility*, page 801.

VMware Based Windows Service Accounts Detection

This auto-detection process will provision accounts based on service accounts detection of all Windows guest machines running on the VMware vCenter environment. The following service accounts will be scanned for Windows services, Scheduled Task, COM Plus, IIS Anonymous and IIS Application Pools.

1. In the process configuration, expand the **Machine Detection** parameters.
2. Expand VMWarevCenter Detection, then display the Machine detection connection details properties.
3. Specify the details of the bind account that will be used to detect machines, using the following parameters:
 - **ADConnectionAccountSafe** – The name of the Safe where the account that contains credentials and information that is required to perform machine detection is stored.
 - **ADConnectionAccountFolder** – The folder where the account that contains credentials and information that is required to perform machine detection is stored.
 - **ADConnectionAccountObject** – The name of the account that contains credentials and information that is required to perform machine detection.

Note: Use one of the following account types:

 - A VMWare vCenter local account with the required permissions in the VMWare vCenter environment,
or,
 - A domain account of the VMWare vCenter domain with the required permissions in the VMWare vCenter environment.
4. In the **ADMachineDetectionTargetAddress** parameter, specify the DNS address of the VMWare vCenter Server.
5. This step is optional: Define the provisioning of more specific accounts based on the vCenter Inventory Browser.
To access the browser:
 - i. In the **VMWarevCenter Detection** section, expand the **Machine Sets** parameters.
 - ii. Specify the browser's base path in one of the following ways:
 - In **Machine Sets**, in the **ADDefaultBasePath**, specify the base path of the vCenter inventory browser,
or,
 - Expand the **Machine Sets**, then in the relevant Machine Set, in the **ADBasePath**, specify the base path of the vCenter inventory browser.

For information about applying additional filtering for Machine Sets, refer to *To define a vCenter detection method*, page 404.
6. In the process configuration, select **Notifications**; the Notification parameters are displayed.
7. In the **NFNotifyOnErrorsRecipients** parameter, specify the email recipients that will be notified about auto-detection errors that occur during this process.

8. Open the CSV file that you will use in the Password Upload Utility, and specify the following parameters for the **ADVMConnectionAccount** account which will be used to connect to the Windows guest machines for the auto-detection service accounts scan:
 - Password parameter
 - Address parameter
 - UserName parameter

This user must have the Administrator role on the VMWare vCenter or be a member of a group with the Administrator role.
9. Run the Password Upload Utility with the modified CSV file to create the Safe, accounts and template environment. For more information, refer to *Running the Password Upload Utility*, page 801.

VMware Based Windows Local Administrator Guest Account Provisioning

This auto-detection process will provision Windows Administrator accounts for all Windows guest machines in the VMWare vCenter Inventory.

1. In the process configuration, expand the **Machine Detection** parameters.
2. Expand VMWarevCenter Detection, then display the Machine detection connection details properties.
3. In the **ADMachineDetectionTargetAddress** parameter, specify the DNS address of the VMWare vCenter Server.
4. This step is optional: Define the provisioning of more specific accounts based on the vCenter Inventory Browser.

To access the browser:

- i. In the **VMWarevCenter Detection** section, expand the **Machine Sets** parameters.
- ii. Specify the browser's base path in one of the following ways:
 - In **Machine Sets**, in the **ADDefaultBasePath**, specify the base path of the vCenter inventory browser,or,
 - Expand the **Machine Sets**, then in the relevant Machine Set, in the **ADBasePath**, specify the base path of the vCenter inventory browser.

For information about applying additional filtering for Machine Sets, refer to *To define a vCenter detection method*, page 404.

5. In the process configuration, select **Notifications**; the Notification parameters are displayed.
6. In the **NFNotifyOnErrorsRecipients** parameter, specify the email recipients that will be notified about auto-detection errors that occur during this process.
7. Open the CSV file that you will use in the Password Upload Utility, and specify the following parameters for the **ADVMConnectionAccount** account which will be used to connect to the VMWare vCenter environment and retrieve relevant information for the auto-detection provisioning process:
 - Password parameter
 - Address parameter
 - UserName parameter

This user must have the Administrator role on the VMWare vCenter or be a member of a group with the Administrator role.

8. Run the Password Upload Utility with the modified CSV file to create the Safe, accounts and template environment. For more information, refer to *Running the Password Upload Utility*, page 801.

Before Configuring Auto-Detection Processes

Steps 1-7 are relevant for LDAP and VMWare vCenter based Auto-Detection processes.

Steps 2 and 5-7 are relevant for Platform based Auto-Detection processes (step 4 is not mandatory and should be only used when a single account has administrator privileges on all target machines).

1. In the PVWA, add the Safe(s) where accounts for detected machines will be stored. These accounts are created according to predefined templates for local accounts. In the Add Safe page, specify the CPM that will manage the accounts in the Safe.

The name of this Safe will be specified in the **ADWorkspaceSafe** parameter when you configure auto-detection processes described below.

Note: These Safes are specifically for auto-detected accounts which will be updated each time the CPM runs in the Active Directory. Any accounts that are created manually in these Safes will be deleted and will not be accessible.

For more information about adding Safes, refer to *Adding Safes in the PVWA*, page 65.

2. Add the Safe(s) where newly detected accounts and service accounts will be stored. These accounts and service accounts are created as a result of a service accounts scanning process. In the Add Safe page, specify the CPM that will manage the accounts in the Safe.

The name of this Safe will be specified in the **ADNewDetectedAccountsSafe** parameter when you configure auto-detection processes described below.

3. Create an account that contains the logon credentials and information required to connect to the appropriate data source. Save this password in a Safe that is owned and managed by the CPM user, **not** in the Safe where new accounts will be created for the automatically detected machines.

- **For LDAP based processes:**

In this account, specify the following properties:

- Address
- Port
- User DN
- User Name (optional)

If this user has already been created and is already managed by the CPM, specify the User DN and Port parameters.

The User Name property is only required if you define a machine scan operation as part of the auto-detection process.

Note: The default port for LDAP with plain protocol is port 389.
The default port for LDAP with SSL protocol is port 636.

This user should only have 'read' permissions in the Active Directory and should only have access to directory locations where the organization's computers are located. This user is called the LDAP Connection Account.

This account will be specified in the **LDAP Connection Details** when you configure auto-detection processes described below.

- **For VMWare vCenter based processes:**

In this account, specify the following properties:

- Address
- User Name

This user should have the administrator role on the VMWare vCenter or be a member of a group with the administrator role.

This account will be specified in the **Machine Detection Connection Details** when you configure auto-detection processes described below.

Note: This account must be a local vCenter user or domain user in the same domain that the vCenter belongs to.

4. Create an account that contains the logon credentials and information required to connect to each machine that will be scanned for service accounts. As in the previous step, save this password in a Safe that is owned and managed by the CPM user, **not** in the Safe where new accounts will be created for automatically detected machines.

In this account, specify the following properties:

- Address
- User Name

This user requires 'administrator' permissions on each remote machine that it will scan. This user is called the AD Machine Connection Account.

This account will be specified in the **Machine Connection Details** when you configure auto-detection processes to scan for service accounts described below.

5. Create a new account that will be used as the local account template for each machine that is detected. Save this account in a Safe owned by the CPM user, **not** in the Safe where new accounts will be created for the automatically detected machines.

i. In the new account, specify the following properties:

- **Device type** – The type of device where the account will be used.
- **Platform Name** – The unique ID of the platform that will be applied to this account.
- **User name** – The name of the local user on the detected machine. This username can either be specified statically or dynamically for LDAP processes as follows:

Specify a pattern that includes a directory attribute that exists in the directory for the local machine. For example, %cn%_Admin would specify a user whose name is comprised of the directory 'cn' attribute and '_Admin'.

If a specified attribute does not exist in the directory for a local machine, the username in the accounts appears as specified in this property and a message is written in the ldap_<policy_id>.log.

- **Address** (this is a required Windows platform property) – Write a dummy value, e.g. "address".

Any other properties that are required in the new accounts created for the detected machines.

- ii. Specify a password that will be used as the default password in each account that will be created for the local account on the detected machine. If there is no default password, a dummy value can be entered and the password will be updated by the reconciliation process.
 - **CPM Disable** – Disable automatic management for this account, so that the CPM will not manage it and replace the password as it is used only as template account. In the reason edit box, specify **ADTemplate**.

This account will be specified in the **Local Account Template** parameters when you configure auto-detection processes to scan for service accounts described below.

6. If the process will scan machines to detect unmanaged domain accounts, create a new account that will be used as the domain account template. Save this account in a Safe owned by the CPM user, **not** in the Safe where new accounts will be created for the automatically detected machines.
 - i. In the new account, specify the following properties:
 - **Device type** – The type of device where the account will be used.
 - **Platform Name** – The unique ID of the platform that will be applied to this account.
 - **UserName** and **Address** (these are required Windows platform properties) – Write dummy values, e.g. “MyUser”.
 - **Logon to** – The name of an Active Directory property that contains the short domain name for the user, in the following format: *%domain name field%*. This property must exist in the Active Directory for the detected machine. If the Active Directory does not have a corresponding property, you can use a fixed value. This domain will be used as the default domain the first time the CPM connects to the remote machine specified in this accounts. This parameter is optional.
 - Any other properties that are required in the new accounts created for the detected machines.
 - ii. Specify a password that will be used as the default password in each account that will be created for the detected domain account. If there is no default password, a dummy value can be entered and the password will be updated by the reconciliation process.
 - iii. Select **Disable automatic management for this account** so that the CPM will not manage it and replace the password as it is used only as template account. In the reason edit box, specify **ADTemplate**.

This account will be specified in the **Domain Account Template** parameters when you configure auto-detection processes to scan for service accounts described below.

7. Create the service accounts template account. This step is optional as the platform name can be specified directly in the process configuration:
 - i. In the PWWA, create a new account for each service accounts type to scan. This account will be used as the account template for each new service account that will be discovered during the machines scan.
 - ii. Save this account in a Safe owned by the CPM user, **not** in the Safe where new accounts will be created for the automatically detected machines.

- iii. In the new account, specify the following properties:
 - **Platform Name** – The unique ID of the platform that will be applied to this service account.
 - **CPM Disable** – Disable automatic management for this account, so that the CPM will not manage it and replace the password as it is used only as template account. In the reason edit box, specify **ADTemplate**.
 - All other required properties are added automatically by the CPM.

This account will be specified in the **Usage Template Account** parameters when you configure auto-detection processes to scan for service accounts described below.

To Configure an Auto-Detection Process to Detect Machines

The following instructions describe how to configure auto-detection processes to detect machines automatically in platforms or external data sources, and create accounts automatically in the Vault for these machines.

1. Expand the new auto-detection process and select **Machine Detection**. You can define three types of different detection methods:
 - Policy (Platform)
 - LDAP
 - VMware

Although more than one machine detection method can be defined under a single auto-detection process, only one is active at any time. This is determined by the value of the **ADMachinedetectionMethod** property in the Machine Detection section.

- **To define an LDAP detection method:**
 - i. In the **Machine Detection** parameters, set **ADMachinedetectionMethod** to **LDAP**.
 - ii. In the Machine Detection section, expand **LDAP detection**.
If this section is not displayed, right-click the Machine detection section, and select **Add LDAP Detection**.
 - iii. In the **LDAP Connection Details** parameters, specify the full pathname of the account that contains credentials and information required to connect to the LDAP directory, as described in step 3 in *Before Configuring Auto-Detection Processes*, page 400, above in the following parameters.
 - **ADLDAPConnectionAccountSafe** – The name of the Safe where the account that contains credentials and information required to connect to the LDAP directory is stored.
 - **ADLDAPConnectionAccountFolder** – The folder where the account that contains credentials and information required to connect to the LDAP directory is stored.
 - **ADLDAPConnectionAccountObject** – The name of the account that contains credentials and information required to connect to the LDAP directory.

- iv. Define the **Machine Sets** to detect. Each machine set represents an OU for machine detection. By default, the auto-detection sample processes contain two separate machine sets, one for Servers detection and one for Workstations detection.

A set of default parameter values is defined for all the machine sets. These parameters will be used as default values for each machine set when a corresponding parameter does not exist in a specific machine set. It can also be overridden by specifying the same parameter for a specific machine set.

In the **Machine Sets** section, for each machine set, specify the following parameters:

- **ADWorkspaceSafe** – The name of the Safe where new accounts that are created according to predefined templates for local accounts will be stored, as described in step 1 of *Before Configuring Auto-Detection Processes*, page 400. If more than one machine set is defined in this auto-detection process, this parameter must be specified for each machine set and default value cannot be used.
 - **ADBaseContext** – The starting point of the search in the directory.
 - **ADQueryFilter** – The LDAP query that defines the computers search inside each container.
 - **ADContainerQueryFilter** – The LDAP query that defines the containers search.
 - **ADIsQueryFilterRecursive** – Specifies whether the search scope is one level beneath the container or in every level beneath it.
- v. After the **Machine Sets** are configured, define the **Directory Properties**. This section defines properties in the external directory that will be used in this auto-detection process. For more information, refer to the *Privileged Account Security Reference Guide*.
- vi. Finally, configure the **Machine Detection Interval** section which defines the intervals between occurrences of detection in this auto-detection process.
- By default the LDAP detection process is configured to run every 7 days at any time or on any day.

- To define a vCenter detection method:
 - i. In the **Machine Detection** parameters, set **ADMachinedetectionMethod** to **VMWare**.
 - ii. In the **Machine Detection** section, expand **VMWare vCenter Detection**. If this section is not displayed, right-click the **Machine detection** section, and select **Add VMWare vCenter detection**.
 - iii. In the **Machine detection connection details** parameters, specify the full pathname of the account that contains credentials and information required to connect to the vCenter, as described in step 3 in *Before Configuring Auto-Detection Processes*, page 400 above in the following parameters.
 - **ADMachinedetectionTargetAddress** - The network address of the VMware vCenter that will be used for machine detection
 - **ADConnectionAccountSafe** – The name of the Safe where the account that contains credentials and information required to connect to the vCenter is stored.

- **ADConnectionAccountFolder** – The folder where the account that contains credentials and information required to connect to the vCenter is stored.
 - **ADConnectionAccountObject** – The name of the account that contains credentials and information required to connect to the vCenter.
- iv. Define the Machine Sets to detect. Each machine set represents a filter for machine detection.
- A set of default parameter values is defined for all the machine sets. These parameters will be used as default values for each machine set when a corresponding parameter does not exist in a specific machine set. It can also be overridden by specifying the same parameter for a specific machine set.
- In the **Machine Sets** section, for each machine set, specify the following parameters:
- **ADVMWareDetectionType** – Defines the machine types that will be detected in the VMWare environment. Options are ESX / ESXi hosts or virtual guest machines.
 - **ADDefaultBasePath** – The starting point of the search in the directory.
 - **ADOSTypeFilter** – Defines a filter of OS family types.
 - **ADVirtualMachineNameFilter** – Defines the filter of virtual machines based on their name in the VMWare environment. This parameter only applies to Machine Sets if the **ADDefaultVMWareDetectionType** parameter is set to **VirtualMachine**.
 - **ADVirtualMachineNetworkNameFilter** – Defines the filter of virtual machines based on their DNS address. This parameter only applies to Machine Sets if the **ADDefaultVMWareDetectionType** parameter is set to **VirtualMachine**.
 - **ADIsQueryFilterRecursive** – Specifies whether the search scope is one level beneath the container or in every level beneath it.
 - **ADDefaultIncludeIncompleteObjects** – Whether or not the CPM will create accounts that cannot be managed by the CPM, such as accounts that do not have a network address. If this parameter is set to Yes, the CPM will create disabled accounts in the workspace Safe. Disabled accounts will be marked with (CPM)ADIncompleteObject.
- v. After the Machine Sets are configured, define the **Machine Detection Properties**. This section defines properties associated to the vCenter directory that will be used in this auto-detection process.
- **ADSSL** – Whether or not an SSL connection will be used to connect to the remote target.
- Note:** If the auto-detection process will connect to VMWare vCenter using its default certificates, make sure that the ADProcess is configured to use **ADSSL=IgnoreUntrustedCertificate**. This type of configuration will enable the use of SSL during connection but will not require the certificates on the VMWare vCenter to be signed. VMware recommends that you replace default certificates with those signed by a commercial certificate authority. It is recommended that VMWare vCenter is installed with signed certificates and that the ADSSL

parameter in the ADProcess configuration is set to ADSSL=Yes.

- **ADPlatformAddressField**- The name of the field in the external directory that contains the machine's address.
 - Note:** If the vCenter environment and guest machines are not configured to work with a DNS address, set this parameter to IPAddress. Using this parameter will force accounts to be created based on IP addresses and not DNS addresses. This configuration is not supported in Windows service account scan processes.

vi. Finally, configure the **Machine Detection Interval** section which defines the intervals between occurrences of detection in this auto-detection process.

By default, the vCenter detection process is configured to run every seven days at any time or on any day.

- **To define a platform detection method:**
 - i. In the **Machine Detection** parameters, set **ADMachineDetectionMethod** to **Policy**.
 - ii. In the **Policy Detection** parameters, in the **ADPolicyName** property, specify the name of the platform that will be used to detect machines.
 - iii. In the main Auto-detection Configuration, right-click **Domains** then select **Add ADDomain**; a new set of domain parameters is created to enable you to specify the domain definitions that translate the domain's NETBIOS name to a DNS name whenever automatic translation of the NETBIOS name cannot be performed. This translation is necessary when the auto-detection finds an account or a service account which is defined on the remote machine with its NETBIOS name.

Add as many ADDomains as you require to define all the domains where machines will be scanned for service accounts during the auto-detection process

2. In the **Account Management** section, define how the auto-detection process will detect and create new accounts. In particular, specify the following properties in this section:

- **ADProvisionMachineLocalAccount** – Set this value to Yes if you wish the auto-detection process to create a local privileged account for each machine that is discovered in the LDAP machine detection.
- **ADProvisionNewDetectedUnmanagedAccounts** – Set this value to Yes if you wish the auto-detection process to create a account object for unmanaged accounts that are detect in the machine service account scan.
- **ADManagedAccountsSafes** – A Safes pattern that indicates the Safes where master accounts of service accounts are already defined and new service accounts for these accounts will be created. Bear in mind that the more Safes you define, the longer the auto-detection process will take, as it will search each Safe in order to find master accounts. On the other hand, if you omit a relevant Safe from this list, the CPM will not find the required master accounts.

Processes that are based on LDAP will search for master accounts to correlate the results of the service accounts scan, whereas processes based on platforms (policies) will search for accounts linked to platforms in order to retrieve IP addresses to create a machine source list and then to correlate the results of the service accounts scan.

Note: The master accounts' address file category must be in Fully Qualified Domain Name (FQDN) format and not IP/NETBIOS/ machine short-name.

- **ADNewDetectedAccountsSafe** – The name of the Safe where detected master accounts and their service accounts will be created as a result of a scanning process. The name of the Safe specified in this parameter must be listed in the `ADManagedAccountsSafes` parameter, as described in step 2 of *Before Configuring Auto-Detection Processes*, page 400.
3. Define other properties under **Account Management** section according to your needs. These properties can also be left with default values:

- **ADDelay** – The delay in minutes between the detection of an account and the time that the CPM will start managing it. A negative number indicates that there will be no delay, and the CPM will start managing the account immediately. If this parameter specifies a delay, detected accounts will only be managed after the specified time. However, if this parameter is changed back to -1 (no delay), accounts that have been detected but are not yet managed by the CPM will only be managed after the next automatic detection cycle.

Define the account archive procedure. To archive accounts that are no longer detected on the Active Directory in a different folder, specify the following parameter:

- **ADArchiveFolder** – The name of the folder where archived accounts will be stored.

To delete accounts that are no longer detected on the Active Directory, specify the following parameter:

- **ADDeleteOnArchive** – Whether or not these accounts will be deleted. The default value is **Yes**, which indicates that accounts will be deleted.

If archived accounts are not deleted, they will appear in the Accounts list with no indication of their location. Therefore, add the **ADArchiveFolder** folder to the **Displayed Columns** parameters in **Accounts UI Preferences for Accounts** in the **Web Access Options** so that users will be able to see the folder where each account is stored, and identify the archived accounts.

4. Expand the **Account Management** parameters and select **New Accounts**. Specify the name and location of the Local Account Template that will be used to create detected local accounts, using the following properties:

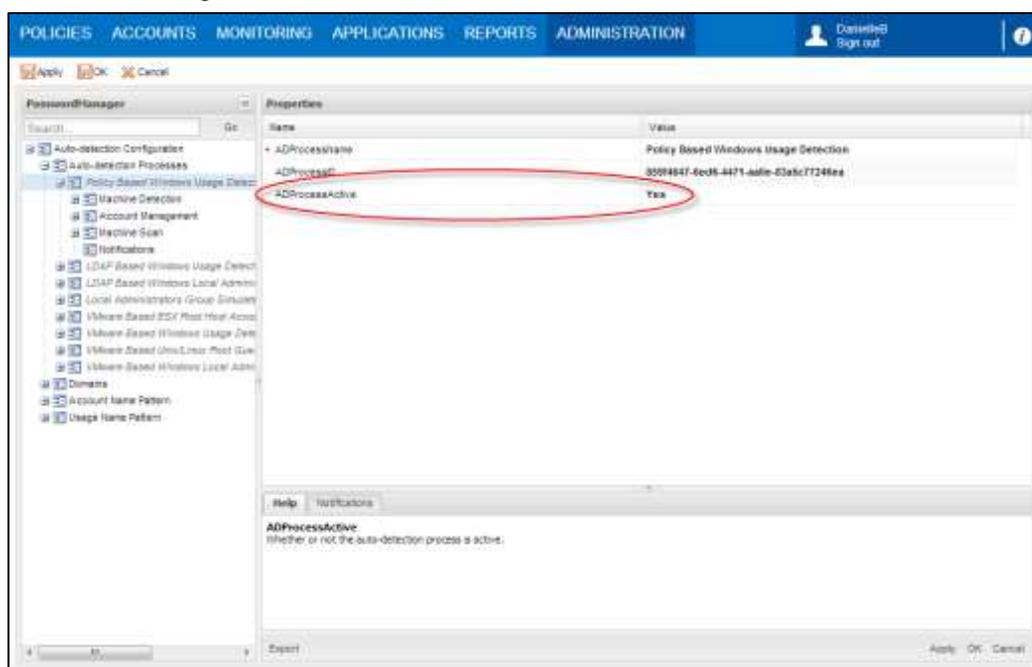
- **ADLocalAccountTemplateSafe** – The name of the Safe where the template account for local accounts is stored.

Note: To enable password reconciliation, after you have created this Safe, specify the Safe name in the `AllowedSafes` parameter.

- **ADLocalAccountTemplateFolder** – The name of the folder where the template account for local accounts is stored.
- **ADLocalAccountTemplateObject** – The name of the template account that will be used to create detected local accounts.

This is described in step 5 of *Before Configuring Auto-Detection Processes*, page 400.

- In each specific process, in the general process parameters, change the **ADProcessActive** property to **Yes**. This activates the auto-detection process that has been configured.



To Configure an Auto-Detection Process to Scan Machines

The following instructions describe how to configure the auto-detection process to scan machines, identify unmanaged accounts and their service accounts, and automatically create corresponding accounts in the Vault for them.

- In the **Machine Scan Interval** parameters, define the intervals between occurrences of machine scanning in this auto-detection process.
 - ADMachineScanInterval** – This parameter specifies how frequently, in minutes, the CPM will run this auto-detection process. The default value is **14** days.

Note: If time limitations are defined (`ADMachineScanFromHour`, `ADMachineScanToHour`, and `ADMachineScanExecutionDays` parameters) the interval must be less than these limitations. For example, if the timeframe when the auto-detection process will run is between 02:00 and 04:00, the interval must be less than 120 minutes (two hours).
 - ADMachineScanFromHour** – This parameter specifies the time when the CPM will start running this auto-detection process. The default value is **-1**, indicating that the auto-detection process is not limited to a specific timeframe.
 - ADMachineScanToHour** – This parameter specifies the time until when the CPM will start running this auto-detection process. The default value is **-1**, indicating that the auto-detection process is not limited to a specific timeframe.
 - ADMachineScanExecutionDays** – This parameter specifies the days of the week when the CPM will start running this auto-detection process. By default, these processes will be run every day of the week.

- **ADImmediatelyScanMachineUponDetection** – This parameter indicates whether or not a machine will be scanned for service accounts as soon as it is detected. The default value is **Yes**, which indicates that the CPM will execute the machine scan on newly detected machines immediately after they are detected. These machines will also be scanned in the regular machine scan, if the process defines it.
2. In the **Machine Scan** section, define the **Machine Connection Details**. These parameters specify the account that is used to log onto the machine to scan.
- **ADMachineConnectionAccountSafe** – The name of the Safe where the account that contains credentials and information required to connect to the remote machine to scan is stored.
 - **ADMachineConnectionAccountFolder** – The folder where the account that contains credentials and information required to connect to the remote machine to scan is stored.
 - **ADMachineConnectionAccountObject** – The name of the account that contains credentials and information required to connect to the remote machine to scan.

Note: The machine scan connection account address file category must be in Fully Qualified Domain Name (FQDN) format and not IP/NETBIOS/machine short-name.

This account was created in step 4 of *Before Configuring Auto-Detection Processes*, page 400.

If this account is not defined, the detected machine account will be used to log onto the machine to scan. This parameter is mandatory if LDAP detection was defined without the provision of machine local accounts, i.e., if the **ADProvisionMachineLocalAccount** parameter was set to **No**.

3. In the **Usage Types** parameters, define the type of service accounts that will be scanned by this auto-detection process on remote machines.
- i. To add a service account to the list of service accounts to be scanned, right click on **Usage Types** and select the required type of service accounts; a new section is created to configure this service account.
 - ii. In the service account's **ADUsagePolicyID** property, specify the ID of the platform that will be applied to service accounts that will be scanned by this process. By default, the platform that is allocated to each type of account is specified in this property when the service account is created, but you can specify the ID of any platform you wish to apply to new service accounts.
If you require more properties than those specified in a platform, you can specify a template account to apply to detected service accounts. If a template account is created, the platform name will be taken from the template.

To create a service account template:

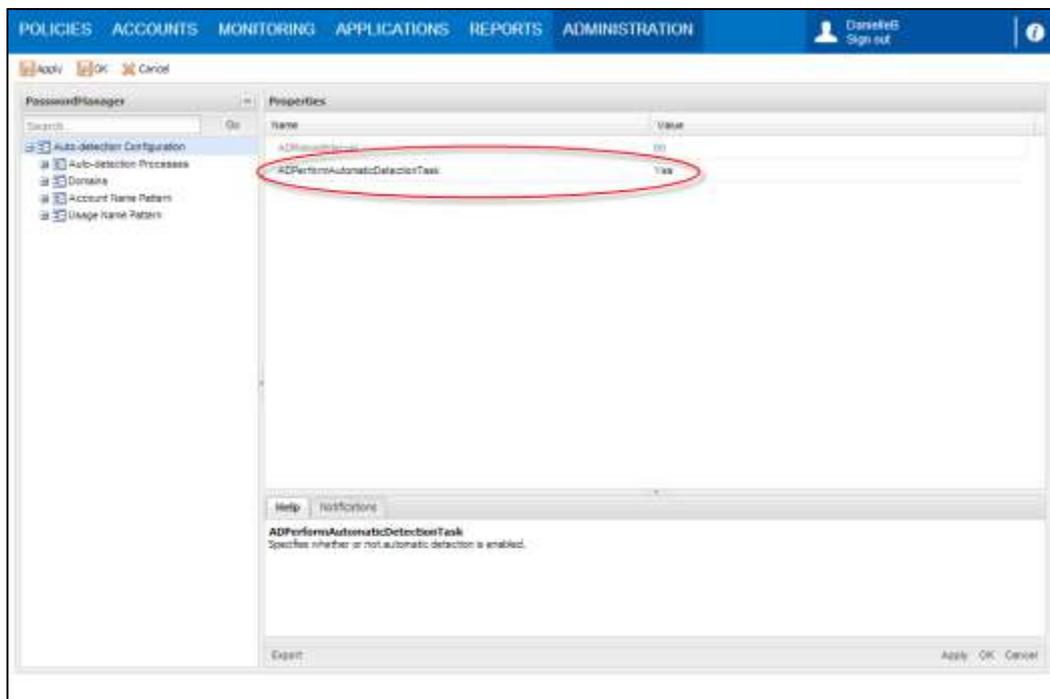
- a. Right-click the name of the service account, and select **Add Usage Template**.
- b. In the **Usage Template** section, specify the **Usage Template Account** in the following properties:
 - **ADUsageTemplateSafe** – The name of the Safe where the template account for the service account is stored.
 - **ADUsageTemplateFolder** – The name of the folder where the template account for the service account is stored.
 - **ADUsageTemplateObject** – The name of the template account to use for the detected usage.

This is described in step 7 of *Before Configuring Auto-Detection Processes*, page 400.

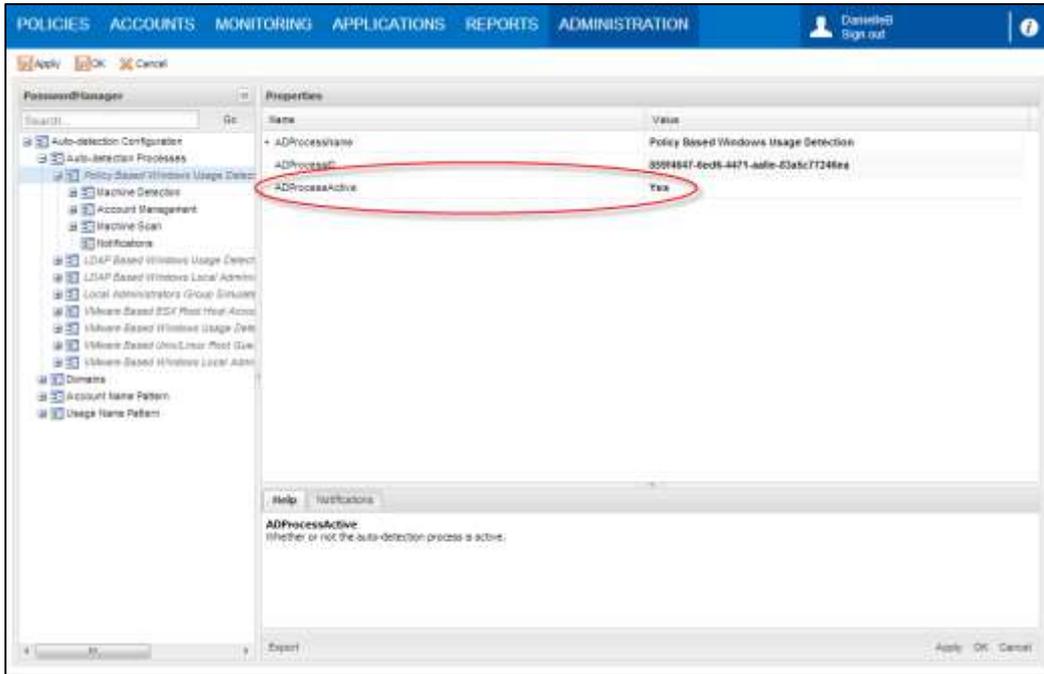
- iii. Set the **ADScanUsage** property that appears in the properties for each service account that is defined. By default, when you add a new service account this property is set to **No**, which indicates that the specific service account type will not be scanned. To enable this scan, set this property to **Yes**.

For information about the other parameters and properties that can be set for Machine Scanning, refer to the Privileged Account Security Reference Guide.

4. In the main Auto-detection Configuration parameters, make sure that the **ADPerformAutomaticDetectionTask** parameter is set to **Yes**. This activates the auto-detection functionality and enables the CPM to execute auto-detection processes.



- In each specific process, in the general process parameters, make sure that the **ADProcessActive** property to **Yes**. This activates the auto-detection process that has been configured.

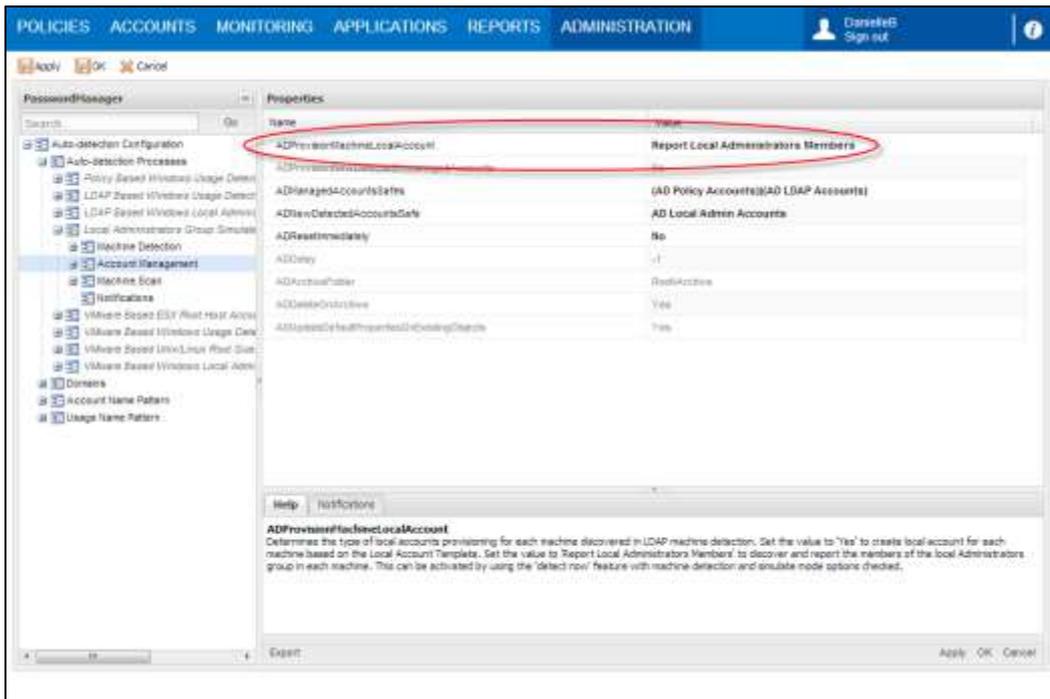


To Configure an Auto-Detection Process to Generate a Local Administrators Group Membership Report

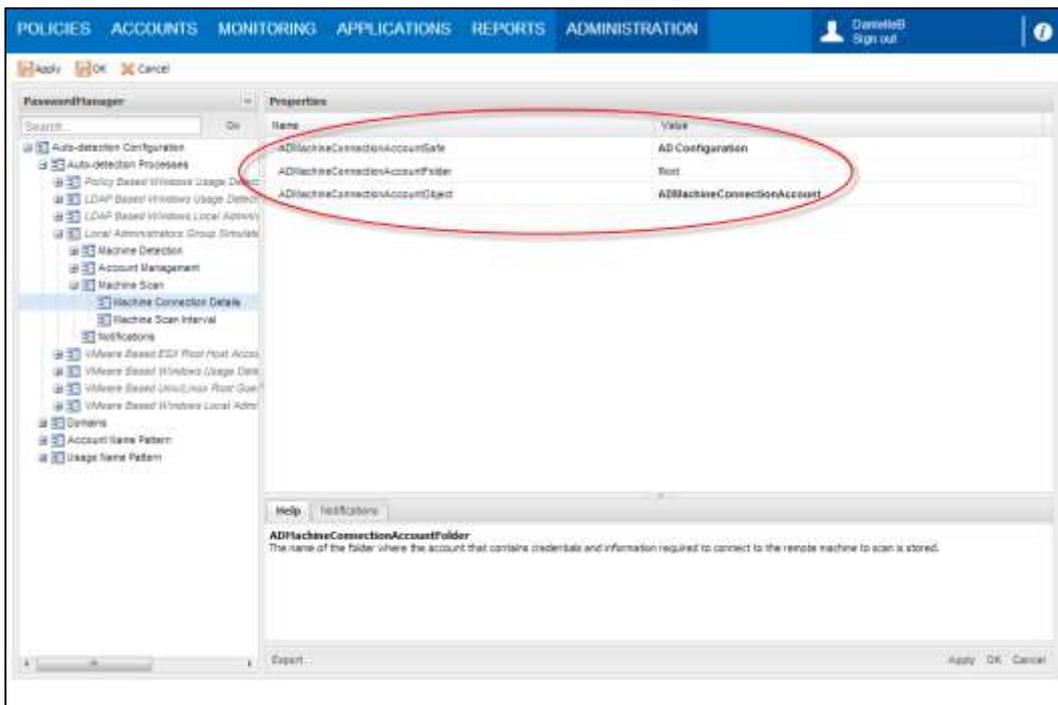
The following instructions describe how to configure auto-detection processes to generate a report that lists all local administrators group members on the target machines. This auto-detection process can be only run in Detect Now simulate mode.

Note: This process is specifically to detect Administrators Group Membership and cannot be used to scan for service accounts as well.

- In the **Auto-detection** parameters, expand the process that will generate the report.
- In the Account Management parameters, set **ADProvisionMachineLocalAccount** to **Report Local Administrators Members**.



3. In the **Machine Scan** parameters, select **Machine Connection Details**, and specify the machine connection detail parameters.

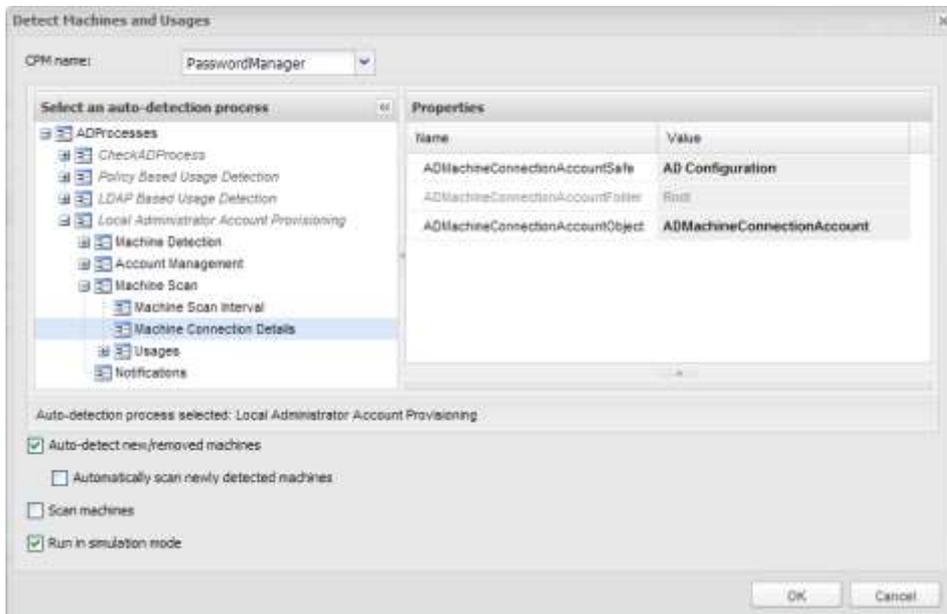


4. Click **Apply** to save the new parameter values and stay in the Web Access Options page,
or,
Click **OK** to save them and return to the System Configuration page.

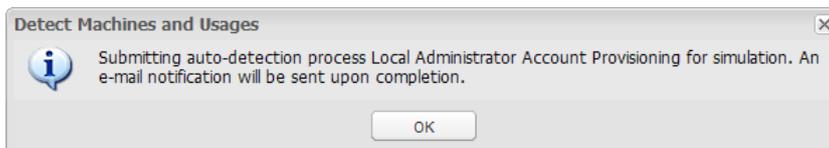
These changes will be applied the next time the CPM refreshes the configuration, according to the value of the **ADReloadInterval** property under in the auto-detection configuration.

To Generate a Local Administrators Group Membership Report

1. In the ACCOUNTS page, click **Detect Now**; the Detect Machines and Usages window appears.
2. Select the auto-detection process that has been configured to detect local administrators group membership on remote machines.
3. Expand the process and check that all the parameters are configured as described above.
4. Select the options according to the whether the process is based on a policy or LDAP data source:
 - Platform:
 - Auto-detect new/removed machines
 - Run in simulation mode
 - LDAP:
 - Scan machines
 - Run in simulation mode



5. Click **OK**; the following message appears.



The CPM runs the auto-detection process and generates a report of local administrators group membership on the remote machines defined in the process. When the process has finished generating the report, an email notification will be sent to your email account to enable you to access the report.

6. Click **OK** to close this message.

To Configure an Auto-Detection Process to Activate Notifications

The following instructions describe how to configure the auto-detection process to activate notifications automatically at different stages of the process. The notifications are activated individually, giving you the flexibility to activate any of the notifications. You can also create your own recipients list that will override the predefined recipients list.

1. In the **Notifications** section, specify any of the following notifications:
 - **Failed process notifications** – When activated, this notification is sent when the auto-detection process fails.
 - **NFNotifyOnErrors** – Indicates whether or not a notification will be sent if a process fails. The default value is **No**.
 - **NFNotifyOnErrorsRecipients** – Lists one or more email addresses for recipients. Multiple addresses are separated by a comma. The addresses specified here replace the default recipients list specified in the ENE.
 - **Successfully completed process notifications** – When activated, this notification is sent when the auto-detection process has run successfully.
 - The **NFNotifyOnSuccess** – Indicates whether or not notifications will be sent after a process has run successfully. The default value is **No**.
 - The **NFNotifyOnSuccessRecipients** – Lists one or more email addresses for recipients. Multiple addresses are separated by a comma. The addresses specified here replace the default recipients list specified in the ENE.
 - **New accounts notifications** – When activated, this notification is sent when new accounts are discovered that are used in service accounts, but are not currently managed by the Privileged Account Security solution.
 - The **NFNotifyOnNewDiscoveredAccounts** – Indicates whether or not notifications will be sent when new accounts are created after being detected in an auto-detection process. The default value is **No**.
 - The **NFOnNewDiscoveredAccountsRecipients** – Lists one or more email addresses for recipients. Multiple addresses are separated by a comma. The addresses specified here replace the default recipients list specified in the ENE.

To Save Auto-Detection Processes

- Click **Apply** to save the new parameter values and stay in the Web Access Options page,

or,

Click **OK** to save them and return to the System Configuration page.

These changes will be applied the next time the CPM refreshes the configuration, according to the value of the **ADReloadInterval** property under in the auto-detection configuration.

To Configure Auto-Detection Processes for SSL Connections

If the CPM will use an SSL connection to connect to the Active Directory, the directory must have its own certificate. Import the CA Certificate that signed the certificate used by the Active Directory into the Windows certificate store on the CPM machine to facilitate this SSL connection (recommended).

1. Display the Microsoft Management Console.
2. From the **File** menu, select **Add/Remove Snap-in**; the Add/Remove Snap-in window appears.
3. Click **Add**; the Add Standalone Snap-in window appears.
4. Select **Certificates**, then click **Add**; the Certificates snap-in window appears.
5. Select **Computer Account**, then click **Next**; the Select Computer window appears.
6. Select **Local Computer**, then click **Finish**; the Add Standalone Snap-in window appears.
7. Click **Close**; the Add/Remove Snap-in window appears and displays Certificates (Local Computer).
8. Click **OK**; the main Console window appears.
9. Expand **Certificates (Local Computer)**, then expand **Trusted Root Certification Authorities**; the Certificates folder appears.
10. Select **Certificates**, then from the **Action** menu, select **All Tasks**, then **Import ...**; the Certificates Import Wizard appears.
11. Click **Next**; the File to Import window appears.
12. Select the certificate file to import, then click **Next**; the Certificate Store window appears.
13. Select **Place all certificates in the following store**, then click **Next**; the Completing the Certificate Import Wizard window appears and displays the details of the selected certificate.
14. Click **Finish**; the selected certificate is imported to the computer account and can now be used to authenticate external users to the CyberArk Vault.

Configuring Active Directory High-Availability Implementations

- In the Directory Properties section, specify the second active directory that the CPM will recognize if the first directory does not respond by the time of the specified timeout in the following parameter:
 - **ADAlternativeAddress** – An additional host for the Active Directory. Values of the main host, the user DN and the password will be taken from the properties specified in the ADPlatformsSourceDetailsObject parameter.

Managing Auto-Detected Accounts with Groups

Accounts that are auto-detected by a specific auto-detection process can be configured so that they are assigned to a single group.

To Configure Auto-Detected Accounts for Group Management

1. Create the following two platforms:
 - One platform to manage the group of auto-detected accounts – Make sure that the **SearchForUsages** parameter is set to **Yes**.
Note: For more details about creating account groups and a complete list of parameters that are relevant to group platforms, refer to *Account Groups*, page 215.
 - One platform to manage the members of the group – This platform must be either for Windows Desktop Local Accounts or Windows Server Local Accounts.
Note: If a platform for auto-detected Windows machines has already been created, you do not need to create another platform.
2. In the PVWA, add a new account using the following guidelines:
 - Add the account in the Safe where the CPM is configured to store auto-detected accounts. By default, this is the AD Workspace Safe.
 - The details of this account will be replaced by information retrieved from auto-detected accounts and, as such, you can specify any value in the address, user name and password content fields.
 - Assign this account to either a Windows Desktop Local Account or a Windows Server Local Account platform.
 - This account will be deleted after the initial auto-detection process finishes.**Note:** The group account must be created in the same Safe as its member accounts. Since the member accounts were created in the AD Workspace Safe, the group account must also be created in that Safe.

3. In the CPM tab of the Account Details page, associate this account with the group that will manage the auto-detected accounts
 - If a group already exists for auto-detected accounts, in the Account Group section, click **Modify**, and then select the group to assign this member to.
 - If a group does not yet exist for auto-detected accounts:
 - i. In the Account Group section, click **Create New**; the Account Group details edit boxes appear.
 - ii. In the Group edit box, specify the name of the new group, for example 'ADAccounts', then click **Save**; the group is created and the Group members list appears.
 - iii. Select the platform name of the group platform you created to manage the group of auto-detected accounts in step 1 above.
4. Configure the template account:
 - i. Make sure that the template account is stored in a Safe owned by the CPM user, and not in the Safe where new accounts will be created for the automatically detected machines.
 - ii. In the Platform Management page, in the platform applied to the template account, specify the **GroupName** property. This defines the name of the group that the automatically detected passwords will belong to.
 - iii. In the template account, add a new account property called **GroupName**, and specify the value defined in the GroupName property in the platform applied to the template account, created in the previous section.
 - iv. In the Account Details of the template account, the following error message will be displayed:

CASTM004E Object [<GroupObjectName>] was not found on Safe [CyberArk Service Accounts]
This message can be ignored.

For more information about creating account properties in the PrivateArk Client, refer to *Defining Custom Account Properties*, page 161.

For more information about adding account properties in platforms, refer to *Modifying Target Account Platforms*, page 109.

To Test Group Management Configuration for Auto-Detected Accounts

1. Create a demo auto-detection process that will detect only two machines on the Active Directory.
2. In the PVWA, display the Account Details page of one of the newly detected passwords and click **Verify**.
3. In the CPM tab, check that the password in the account was verified successfully.
4. Display the Account Details page of the other password that was created automatically, and in the CPM tab, check that the password in the account was also verified successfully.

Unix Accounts

Automatic password and key management is supported on Unix accounts on IPv4 and IPv6.

Note: On HP-UX 11.x, automatic password management is only supported on IPv4.

Supported Platforms

The CPM supports remote password management on Unix machines on the following platforms:

- Sun Solaris 5.8
- HP-UX 11.x
- IBM AIX 5
- RHEL 3

Connection Methods

This plug-in supports the following connection methods to connect to the remote machine:

- SSH
- Telnet

Platform

In the Platform Management page, make sure that one of the following target account platforms is displayed, according to the connection method you have chosen:

- Unix via SSH (for password authentication)
- Unix via SSH Keys (for SSH key authentication)
- Unix via Telnet (for password authentication only)

Password Management Features

The CPM can change and verify Unix passwords and SSH keys on remote machines. If a password or key content is invalid, the CPM can generate new content and replace it on the remote machine as well as its corresponding content in the Password Vault. The parameters that define these tasks are in the platform. A reconciliation account can be specified either at platform level or at account level.

For both the Unix via SSH and Unix via SSH Keys platforms, reconciliation accounts can authenticate to the target machine with either a password or an SSH Key.

Notes:

- The reconcile account must use a root user or a power user with root permissions.
- If the reconcile account user authenticates to the target server with a password, on the target machine, in `sshd_config`, set the **PasswordAuthentication** parameter to **yes**.

For more information, refer to *Configuring Accounts for Automatic Management*, page 377.

Additional Logon Passwords

For the Unix via SSH platform, additional logon accounts can authenticate to the target machine with either a password or an SSH Key. For the Unix via SSH Keys platform, logon accounts are not supported.

For more information, refer to *Linked Accounts*, page 220.

AS400 (iSeries) Accounts

Automatic password management is supported on AS400 accounts on IPv4.

Supported Platforms

The CPM supports remote password management on IBM's AS400 (iSeries) on the following platforms:

- AS400 (iSeries) computers using OS/400 V5R2 or higher

Installing IBM iSeries Access for Windows

- On the machine that runs the CPM, install the IBM iSeries Access for Windows for the supported OS version.

For more information about supported OS versions, refer to <http://www-03.ibm.com/systems/i/software/access/windows/supportedos.html>.

Port

This plug-in supports communication between the CPM and the AS400 machine on the following ports:

- Non-SSL ports:
 - 449
 - 8476 - This is the default port.
 - 8475 – For reconcile operations
- SSL:
 - 9475 – For reconcile operations
 - 9476

Platform

In the Platform Management page, make sure that the following target account platform is displayed:

- AS400

To configure an SSL connection, in the **Additional Policy Settings** section of the platform, set the following parameter:

Parameter	Description	Acceptable Values	Default Value
UseSSL	Whether or not an SSL connection will be used to connect to the remote device.	Yes/No	No

Password Management Features

The CPM can change, verify, and reconcile AS400 (iSeries) passwords on remote machines and store a corresponding password in the Password Vault. The parameters that define the password change process are in the platform.

By default, password generation and reconciliation for accounts managed by this platform are restricted by the following parameters:

- **QPWDPOSIDF (same character in the same position rule)** - Whether or not characters (alphabetic or numeric) can be used in the same positions as in the previous password. This is not case-sensitive, meaning that characters cannot be used in the same position, regardless of whether they are upper or lower case. For example, '123a' cannot be replaced by '456a' or '456A'. This is defined by the **PreventSameCharPerPrevPassPosition** parameter in the platform. By default, this parameter is set to **Yes** in the AS400 platform.
- **QPWDLMTREP (repeated characters rule)** – Whether or not characters can be used more than once in a password. This feature does not allow repeated characters anywhere in the password, consecutive or not. For example, the password cannot be set to '123aa456', '123a456a', or '123A456a'. In addition, it is not case-sensitive, meaning that characters cannot be repeated, regardless of whether they are upper or lower case. This is defined by the **PreventRepeatingCharacters** parameter in the platform. By default, this parameter is set to **Yes** in the AS400 platform.

Supported Users

The CPM can manage AS400 (iSeries) passwords on remote machines for the following users:

- Regular operating system users
- Dedicated Service Tools (DST) users
- System Service Tools (SST) users

Supporting Regular Operating System Users

By default, AS400 (iSeries) accounts for regular OS users are defined as **RegularUserProfile** type accounts. This means that they do not need additional logon or reconciliation accounts. Logon accounts that are associated with RegularUserProfile accounts will be ignored. Reconciliation accounts that are associated with RegularUserProfile accounts must also be defined as RegularUserProfile accounts.

Supporting Dedicated Service Tools (DST) and System Service Tools (SST) Users

AS400 (iSeries) accounts for Service Tools users must be defined as **ServiceToolUser** type accounts. These accounts require the following additional accounts:

- **Logon account** – An additional logon account is required for all operations on Service tool accounts. This account must be defined as a regular OS account (**RegularUserProfile** type account) and can be defined at either platform or account level.

- The logon account requires the following permissions:
 - *SERVICE
 - *SECADM
 - *ALLJOB
- **Reconcile account** – An additional reconcile account is mandatory for reconcile operations on Service tool accounts. This account must be defined as a Service tool account (**ServiceToolUser** type account).

The reconcile account needs the Service tool security privilege.

Note: After two invalid attempts to manage service tool accounts, the CPM disables automatic management before they are disabled in the remote device. Therefore, it is highly recommended to configure the main account for automatic reconciliation. If the account in the remote device is locked due to invalid attempts, the reconciliation process will unlock it.

Notes:

- The CPM supports DES encryption for password policies used by service tool users. This encryption uses the following characteristics:
 - User IDs specify ten uppercase characters. The reconcile user ID is limited to eight characters.
 - Passwords must contain eight alphanumeric, case-sensitive characters.

For more information about Service tool password policies, refer to:

<http://publib.boulder.ibm.com/infocenter/iseres/v5r4/index.jsp?topic=/rzamh/rzamhpwpolicies.htm>

- It is recommended to configure this platform to allow default and expired passwords to be changed.
 1. In the Dedicated Service Tool (DST) or System Service Tools (SST), select Work with System Security.
 2. In the Work with System Security display, change Allow a service tools user ID with a default and expired password to change its own password field from No (the default) to Yes.

For more information, refer to *Configuring Accounts for Automatic Management*, page 377.

OS/390 (Z/OS) Accounts

Automatic password management is supported on OS/390 accounts on IPv4.

Supported Platforms

The CPM supports remote password management on OS/390 (Z/OS) machines for RACF users' passwords.

Connection Methods

This plug-in supports the following connection methods to connect to the remote machine:

- SSH
- Telnet
- FTP

Platform

In the Platform Management page, make sure that one of the following target account platforms is displayed, according to the connection method you have chosen.

- OS390 via SSH
- OS390 via Telnet
- OS390 via FTP

Requirements

SSH/Telnet must be enabled through the USS (Unix Services) on the remote machine.

Password Management Features

The CPM can change OS/390 (Z/OS) passwords on remote machines and store a corresponding password in the Password Vault. The parameters that define the password change process are in the platform. For more information, refer to *Configuring Accounts for Automatic Management*, page 377.

Additional Logon Passwords (relevant for SSH/Telnet only)

If the CPM changes a password that belongs to a user who requires an additional password to log onto the remote machine, create an additional password object to contain the extra user's logon details, and link it to the original password object. For more information, refer to *Linked Accounts*, page 220.

Note:

If you chose FTP as the protocol to use to log onto the remote machine, if the password is incorrect the logon phase will succeed and the error will be given in the change password phase. This is done because the FTP protocol doesn't return different errors when the password is expired and when it is invalid. As the CPM supports password management even when the passwords have expired, this error is treated as a success and it is checked again in the next phase.

It is important to understand that the process will not succeed under any circumstances if it should fail due to a problem.

ESX/i Accounts

Supported Platforms

The CPM supports remote password management on ESX/i machines for server accounts on the following platforms:

- ESXi 5.0, ESX/ESXi 4.1, ESX/ESXi 4.0, ESX 3.5

Connection Methods

This platform supports the following connection method to connect to the remote machine:

- HTTP/HTTPS

Supported Users

This platform supports the following connection methods to connect to the remote machine:

- Root
- Local users on ESX/i, excluding DCUI, Vpxuser, and Vimuser users

Note: This plug-in only manages accounts on ESX/ESXi machines. vCenter accounts are managed as regular Windows accounts by the Windows plug-in.

Platform

In the Platform Management page, make sure that the following target account platform is displayed:

- VMware ESX Account API

To configure and SSL connection type, in the **Additional Policy Settings** section of the platform, set the following parameter:

Parameter	Description	Acceptable Values	Default Value
UseSSL	Whether or not an SSL connection will be used to connect to the remote device.	Yes/No/ IgnoreUntrusted Certificate	IgnoreUntrusted Certificate

Password Management Features

The CPM can change, verify, and reconcile ESX/i Accounts passwords on remote machines and store a corresponding password in the Password Vault. The parameters that define the password change process are in the platform. For more information, refer to *Configuring Accounts for Automatic Management*, page 377.

The user that will change and reconcile the remote account requires the following role on the remote machine:

- Administrator

The user that will verify the remote account requires the following role on the remote machine:

- Read only

Additional Logon Passwords

If the CPM changes a password that belongs to a user who requires an additional password to log onto the remote machine, create an additional password object to contain the extra user's logon details, and link it to the original password object. For more information, refer to *Linked Accounts*, page 220.

Databases

Databases that Support ODBC Connections

Automatic password management is supported on databases that support ODBC connections on IPv4 and IPv6.

Supported Platforms

The CPM supports remote password management on all databases that support ODBC connections.

Support for ODBC

The machine that runs the CPM must support ODBC, version 2.7 and higher. If the machine does not support this version of ODBC, download the latest **MDAC_typ.exe** from the Microsoft downloads site.

Platform

In the Platform Management page, make sure that the following target account platform is displayed:

- ODBC

Connection Methods

This platform supports the following connection methods to connect to remote databases:

- DSN
- Connection string (DSN-less)

To Connect to the Remote Database with DSN

1. Create a System DSN for each database in the CPM machine.
2. Use the testing option in the DSN to test the connection between the CPM machine and the database server.

Note: Make sure that the DSN is a system DSN and not a User DSN.

A DSN may be required to perform management tasks on database accounts. Creating a DSN for a remote database on the CPM server may require users to work with external ODBC drivers (MSSQL, Oracle, etc...). In order to work with 32-bit ODBC drivers on 64-bit platforms (such as the Oracle instant client that is part of the CPM installation), use the ODBC tool which can be found in the following path:
c:\windows\sysWOW64\odbcad32.exe.

To Connect to the Remote Database with a Connection String (DSN-less)

1. Display the ODBC platform, and in the **Additional Policy Settings** section, check the values of the following **required** parameter:

Parameters	Description
ConnectionCommand	The SQL statement that will be used as a connection string. This parameter is required if the DSN parameter is not supplied either at platform level or password level, indicating that the connection method is DSN-Less.

For more details about platform parameters, refer to the Privileged Account Security Reference Guide.

The Connection String template may contain any number of the following variables enclosed with '%' (percentage) sign. These variables will be replaced during run time with the appropriate values from standard or customized account properties. Any of these variables may be specified in the connection string as optional parameters, which will only be replaced if the account property exists.

Variable	Replaced by ...	Taken from
USER	The user name.	Password object properties
OLD PASSWORD	The current password.	Password object properties
LOGON PASSWORD	The password used for logon. The platform automatically detects the password to use for logon. This is either the current password or the password of the reconciliation account.	Password object properties or Reconcile account password object properties
ADDRESS	The address of the remote machine where the password is used.	Password Object properties or, if not defined there, from the Extrainfo section of the platform.
INSTANCE	The instance name of the database environment being managed.	Password object properties or, if not defined there, from the Extrainfo section of the platform.
PORT	The port used by the CPM to access the remote machine.	Password Object properties or, if not defined there, from the Extrainfo section of the platform.
DATABASE	The database name.	Password Object properties or, if not defined there, from the Extrainfo section of the platform.
RECONCILE USER	The name of the user who will replace the invalid password with the new password.	Reconcile account password object properties

Variable	Replaced by ...	Taken from
RECONCILE PASSWORD	The password of the user who will replace the invalid password with the new password.	Reconcile account password object properties

The following example displays a valid connection string for a Microsoft SQL server:

```
Driver={SQL Server};Server=%ADDRESS% [\%INSTANCE%] [, %PORT%];Database=%DATABASE%;Uid=%USER%; Pwd=%LOGONPASSWORD%;
```

In the above example, the instance and the port used to log onto the remote machine are optional and will only be replaced if they exist in the properties of the account being used.

You can also set entire parts of the connection string as optional, as shown in the following example:

```
Driver={SQL Server};Server=%ADDRESS% [\%INSTANCE%] [, %PORT%] ; [Database=%DATABASE%]; Uid=%USER%; Pwd=%LOGONPASSWORD%;
```

In the above example, the name of the database that the account will be used to log onto is optional and will only be specified in the connection string if it is specified in the account properties.

2. Create a temporary DSN and use the testing options in the DSN to test the connection between the CPM machine and the database server.

Configuring the Password Change SQL Statement for ODBC Passwords

During installation a default SQL statement template is configured for each database vendor whose users will be managed by the CPM during a password change task.

- In the **Additional Policy Settings** section of the platform, check the value of the following **required** parameter:

Parameters	Description
ChangeCommand	The legal SQL statement template that will be used to change the password on the required database.

For more details about platform parameters, refer to the Privileged Account Security Reference Guide.

The statement template can contain any number of the following variables enclosed with '%' (percentage) sign. These variables will be replaced during run time with the appropriate values:

Variable	Replaced by ...	Taken from
USER	The user name	Password Object properties
OLDPASSWORD	The current password	Password Object properties
NEWPASSWORD	The new generated password	Password Object properties
DATABASE	The database name	Password Object properties or, if not defined there, from the ExtralInfo section of the platform.

Variable	Replaced by ...	Taken from
RECONCILE USER	The name of the user who will replace the invalid password with the new password.	Reconcile account password object properties
RECONCILE PASSWORD	The password of the user who will replace the invalid password with the new password.	Reconcile account password object properties
LOGON PASSWORD	The password that the plug-in uses for logon. The plug-in automatically detects the password to use for logon. This is either the current password or the password of the reconciliation account.	Password object properties or Reconcile account password object properties

To ensure that these variables will be used as values and not as part of the command, when possible enclose them with quotation marks, as shown in the example below.

The following example displays the command used to change a user password on a Microsoft SQL server:

```
sp_password "%OLDPASSWORD%", "%NEWPASSWORD%"
```

- To reduce the risk of a security hazard, in the **Additional Policy Settings** section of the platform, specify the following parameters:

Parameters	Description
CommandForbiddenCharacters	Characters that cannot be used in the parameters of the change command, listed in the above table. Default values are <code>"V@".'{}() -!*>~!^#</code> .
CommandBlackList	Words that cannot be used in the change command, listed in the above table. Default values are delete, drop, exec .

Note: After upgrading to v9.5, add these parameters manually and specify the default values.

Configuring the Password Reconciliation SQL Statement for ODBC Passwords

During installation a default SQL statement template is configured for each database vendor whose users will be managed by the CPM during a password reconciliation task.

- In the **Additional Policy Settings** section of the platform, check the value of the following **required** parameter:

Parameters	Description
ReconcileCommand	The legal SQL statement template that will be used to reconcile the password on the required database.

For more details about platform parameters, refer to the Privileged Account Security Reference Guide.

The statement template can contain any number of the following variables enclosed with '%' (percentage) sign. These variables will be replaced during run time with the appropriate values:

Variable	Replaced by ...	Taken from
USER	The user name	Password Object properties
OLDPASSWORD	The current password	Password Object properties
NEWPASSWORD	The new generated password	Password Object properties
DATABASE	The database name	Password Object properties or, if not defined there, from the ExtralInfo section of the platform.
RECONCILE USER	The name of the user who will replace the invalid password with the new password.	Reconcile account password object properties
RECONCILE PASSWORD	The password of the user who will replace the invalid password with the new password.	Reconcile account password object properties
LOGON PASSWORD	The password that the plug-in uses for logon. The plug-in automatically detects the password to use for logon. This is either the current password or the password of the reconciliation account.	Reconcile account password object properties

To ensure that these variables will be used as values and not as part of the command, when possible enclose them with quotation marks, as shown in the example below.

The following example displays the command used to reconcile a user password on a Microsoft SQL server:

```
sp_password @new="%NEWPASSWORD%", @loginame="%USER%"
```

- To reduce the risk of a security hazard, in the **Additional Policy Settings** section of the platform, specify the following parameters:

Parameters	Description
CommandForbiddenCharacters	Characters that cannot be used in the parameters of the reconcile command, listed in the above table. Default values are "V@".'{}() - *>~!^#.
CommandBlackList	Words that cannot be used in the reconcile command, listed in the above table. Default values are delete, drop, exec .

Note: After upgrading to v9.5, add these parameters manually and specify the default values.

Adding the Password

When you add the password or edit an existing one to be used on an ODBC database, specify the following properties:

- For a DSN connection:

Property	Description
DSN	A legal DSN name that will be used to access the database through ODBC. This parameter is required if the ConnectionStringFile parameter was not supplied in the platform, indicating that the connection method is DSN, or if the DSN parameter in the platform was not specified.

- For a Connection String connection:

Property	Description
Address	The address of the database server that is used as the value of the ADDRESS variable in the Connection String. This parameter is required.
Port	The port number of the database server. This parameter is optional.
Database	The database name that will be used as the value of the DATABASE variable in the Connection String. This parameter is optional.

Securing the connection between the CPM and the database server

It is recommended to use IPsec (Internet Protocol Security) to secure the connection between the CPM and the Database Server. IPsec is a transport layer mechanism through which you can ensure confidentiality and integrity of TCP/IP-based communications between computers.

IPsec can be configured to secure the connection between the machine that hosts the CPM and the machine that hosts the Database Server, or to secure the entire network.

As IPsec is transparent to applications, it will not affect the communication with the Database Server and the other clients, while at the same time will secure the connection with the CPM.

For more information refer to the Microsoft documentation.

Password Management Features

The CPM can change passwords on databases that support ODBC connections on remote machines and store a corresponding password in the Password Vault. The parameters that define the password change process are in the platform. For more information, refer to *Configuring Accounts for Automatic Management*, page 377.

Oracle Database Accounts

Automatic password management is supported on Oracle database accounts on IPv4 and IPv6.

Supported Platforms

The CPM supports remote password management on Oracle databases on the following platforms:

- Oracle Database V8i-12c

Note: If this version of Oracle does not support the version of Oracle that is installed in your enterprise, install a different version of the Oracle Instant Client on the CPM machine.

ODBC Support

The machine that runs the CPM must support ODBC, version 2.7 and higher. If the machine does not support this version of ODBC, download the latest **MDAC_typ.exe** from the Microsoft downloads site.

During CPM installation, the Oracle Instant Client, version 11, is installed. This enables the CPM to use the Oracle ODBC driver, and connect to Oracle databases without the need for TNSNames.

Platform

In the Platform Management page, make sure that the following target account platform is displayed:

- Oracle

Connection Methods

This plug-in supports the following connection methods to connect to remote databases:

- DSN
- Connection string (DSN-less) without TNSNames
- Connection string (DSN-less)

To Connect to the Remote Database with DSN

1. Create a System DSN for each database in the CPM machine.
2. Use the testing option in the DSN to test the connection between the CPM machine and the database server.

Note: Make sure that the DSN is a system DSN and not a User DSN.

To Connect to the Remote Database with a Connection String (DSN-less) and with TNSNames

1. In the Oracle platform, in the **Additional Policy Settings** section, check the value of the following parameter:

Parameters	Description
ConnectionCommand	The SQL statement that will be used as a connection string. This parameter is required if the DSN parameter is not supplied either at platform level or password level, indicating that the connection method is DSN-Less.

For more details about platform parameters, refer to the Privileged Account Security Reference Guide.

The Connection String template may contain any number of the following variables enclosed with '%' (percentage) sign. These variables will be replaced during run time with the appropriate values:

Variable	Replaced by ...	Taken from
USER	The user name.	Password object properties
OLD PASSWORD	The current password.	Password object properties
LOGON PASSWORD	The password that the plug-in uses for logon. The plug-in automatically detects the password to use for logon. This is either the current password or the password of the reconciliation account.	Password object properties or Reconcile account password object properties
ADDRESS	The address of the remote machine where the password is used.	Password Object properties or, if not defined there, from the Extrainfo section of the platform.
PORT	The port used by the CPM to access the remote machine.	Password Object properties or, if not defined there, from the Extrainfo section of the platform.
RECONCILE USER	The name of the user who will replace the invalid password with the new password.	Reconcile account password object properties
RECONCILE PASSWORD	The password of the user who will replace the invalid password with the new password.	Reconcile account password object properties

The following example displays a valid connection string for an Oracle database:

```
Driver={Oracle in instantclient11_1};Dbq=%ADDRESS%;Uid=%USER%;
Pwd=%OLDPASSWORD%;
```

2. Create a temporary DSN and use the testing options in the DSN to test the connection between the CPM machine and the database server.

To Connect to the Remote Database with a Connection String (DSN-less) and without TNSNames

1. In the Oracle platform, in the **Additional Policy Settings** section, check the value of the following parameter:

Parameters	Description
ConnectionCommand	The SQL statement that will be used as a connection string. This parameter is required if the DSN parameter is not supplied either at platform level or password level, indicating that the connection method is DSN-Less.

For more details about platform parameters, refer to the Privileged Account Security Reference Guide.

The Connection String template may contain any of the following variables enclosed by the '%' (percentage) sign. These variables will be replaced during run time with the appropriate values:

Variable	Replaced by ...	Taken from
USER	The user name.	Password object properties
OLD PASSWORD	The current password.	Password object properties
LOGON PASSWORD	The password that the plug-in uses for logon. The plug-in automatically detects the password to use for logon. This is either the current password or the password of the reconciliation account.	Password object properties or Reconcile account password object properties
ADDRESS	The address of the remote machine where the password is used.	Password Object properties or, if not defined there, from the ExtralInfo section of the platform.
PORT	The port used by the CPM to access the remote machine.	Password Object properties or, if not defined there, from the ExtralInfo section of the platform.
DATABASE	The name of the Oracle service.	Password Object properties or, if not defined there, from the ExtralInfo section of the platform.
RECONCILE USER	The name of the user who will replace the invalid password with the new password.	Reconcile account password object properties
RECONCILE PASSWORD	The password of the user who will replace the invalid password with the new password.	Reconcile account password object properties

The following example displays a valid connection string for an Oracle database:

```
Driver={Oracle in
instantclient11_1};Dbq=//%ADDRESS%:%PORT%/%DATABASE%;Uid=%USER%;Pwd=%LOGO
NPASSWORD%;
```

2. Create a temporary DSN and use the testing options in the DSN to test the connection between the CPM machine and the database server.

Configuring the Password Change SQL Statement for Oracle Passwords

During installation a default SQL statement template is configured for each database vendor whose users will be managed by the CPM during a password change task.

- In the Oracle platform, in the **Additional Policy Settings** section, check the value of the following **required** parameter:

Parameters	Description
ChangeCommand	The legal SQL statement template that will be used to change the password on the required database.

For more details about platform parameters, refer to the Privileged Account Security Reference Guide.

The statement template can contain any number of the following variables enclosed with '%' (percentage) sign. These variables will be replaced during run time with the appropriate values:

Variable	Replaced by ...	Taken from
USER	The user name	Password Object properties
OLDPASSWORD	The current password	Password Object properties
NEWPASSWORD	The new generated password	Password Object properties
DATABASE	The database name	Password Object properties or, if not defined there, from the ExtralInfo section of the platform.
RECONCILE USER	The name of the user who will replace the invalid password with the new password.	Reconcile account password object properties
RECONCILE PASSWORD	The password of the user who will replace the invalid password with the new password.	Reconcile account password object properties
LOGON PASSWORD	The password that the plug-in uses for logon. The plug-in automatically detects the password to use for logon. This is either the current password or the password of the reconciliation account.	Password object properties or Reconcile account password object properties

To ensure that these variables will be used as values and not as part of the command, when possible enclose them with quotation marks, as shown in the example below.

The following example displays the command used to change a user password on an Oracle database:

```
alter user %USER% identified by "%NEWPASSWORD%";
```

If the user does not have the ALTER_USER privilege, specify the following:

```
alter user %USER% identified by "%NEWPASSWORD%" replace "%OLDPASSWORD%";
```

- To reduce the risk of a security hazard, in the **Additional Policy Settings** section of the platform, specify the following parameters:

Parameters	Description
CommandForbiddenCharacters	Characters that cannot be used in the parameters of the change command, listed in the above table. Default values are <code>"V@".!{}() -!*>~!^#</code> .
CommandBlackList	Words that cannot be used in the change command, listed in the above table. Default values are delete, drop, exec .

Note: After upgrading to v9.5, add these parameters manually and specify the default values.

Configuring the Password Reconciliation SQL Statement for Oracle Passwords

During installation a default SQL statement template is configured for each database vendor whose users will be managed by the CPM during a password reconciliation task.

- In the Oracle platform, in the **Additional Policy Settings** section, check the value of the following **required** parameter:

Parameters	Description
ReconcileCommand	The legal SQL statement template that will be used to reconcile the password on the required database.

For more details about platform parameters, refer to the Privileged Account Security Reference Guide.

The statement template can contain any number of the following variables enclosed with '%' (percentage) sign. These variables will be replaced during run time with the appropriate values:

Variable	Replaced by ...	Taken from
USER	The user name.	Password Object properties
OLDPASSWORD	The current password.	Password Object properties
NEWPASSWORD	The new generated password.	Password Object properties
DATABASE	The database name.	Password Object properties or, if not defined there, from the ExtralInfo section of the platform.
RECONCILE USER	The name of the user who will replace the invalid password with the new password.	Reconcile account password object properties
RECONCILE PASSWORD	The password of the user who will replace the invalid password with the new password.	Reconcile account password object properties
LOGON PASSWORD	The password that the plug-in uses for logon. The plug-in automatically detects the password to use for logon. This is either the current password or the password of the reconciliation account.	Reconcile account password object properties

To ensure that these variables will be used as values and not as part of the command, when possible enclose them with quotation marks, as shown in the example below.

The following example displays the command used to reconcile a user password on an Oracle database:

```
alter user %USER% identified by "%NEWPASSWORD%";
```

- To reduce the risk of a security hazard, in the **Additional Policy Settings** section of the platform, specify the following parameters:

Parameters	Description
CommandForbiddenCharacters	Characters that cannot be used in the parameters of the reconcile command, listed in the above table. Default values are 'V@".'{}() - ">~!^#.
CommandBlackList	Words that cannot be used in the reconcile command, listed in the above table. Default values are delete, drop, exec .

Note: After upgrading to v9.5, add these parameters manually and specify the default values.

Adding the Password

When you add the password or edit an existing password that will be used on an ODBC database, specify the following properties:

- For a DSN connection:

Property	Description
DSN	A legal DSN name that will be used to access the database through ODBC. This parameter is required if the ConnectionStringFile parameter was not supplied in the platform, indicating that the connection method is DSN, or if the DSN parameter in the platform was not specified.

- For a Connection String connection:

Property	Description
Address	The address of the database server that is used as the value of the ADDRESS variable in the Connection String. This parameter is required.
Port	The port number of the database server. This parameter is optional.
Database	<ul style="list-style-type: none"> ▪ Using TNSNames – The database name that will be used as the value of the DATABASE variable in the Connection String. This parameter is optional. ▪ Without using TNSNames – The name of the Oracle service.

Password Management Features

The CPM can change and verify Oracle database passwords on remote machines. If a password is invalid, the CPM can generate a new password and replace the invalid password on the remote machine and its corresponding password in the Password Vault. The parameters that define these tasks are in the platform. A reconciliation account password can be specified either at platform level or at account level.

For more information, refer to *Configuring Accounts for Automatic Management*, page 377.

Microsoft SQL Server Accounts

Automatic password management is supported on Microsoft SQL Server accounts on IPv4 and IPv6.

Supported Platforms

The CPM supports remote password management on a Microsoft SQL Server on the following platforms:

- Microsoft SQL Server 7 and above

Required Drivers

The following driver is required on the CPM machine:

- Microsoft SQL Server ODBC driver

Make sure that a Microsoft SQL Server ODBC driver is installed on the machine that runs the CPM. If this driver not installed you can download it from Microsoft downloads web site.

ODBC Support

The machine that runs the CPM must support ODBC, version 2.7 and higher. If the machine does not support this version of ODBC, download the latest **MDAC_typ.exe** from the Microsoft downloads site.

Platform

In the Platform Management page, make sure that the following target account platform is displayed:

- MSSql

Connection Methods

This plug-in supports the following connection methods to connect to remote databases:

- DSN
- Connection string (DSN-less)

To Connect to the Remote Database with DSN

1. Create a System DSN for each database in the CPM machine.
2. Use the testing option in the DSN to test the connection between the CPM machine and the database server.

Note: Make sure that the DSN is a system DSN and not a User DSN.

To Connect to the Remote Database with a Connection String (DSN-less)

1. In the **Additional Policy Settings** section of the platform, check the values of the following **required** parameters:

Parameters	Description
ConnectionCommand	The Connection String that will be used to connect to the database through ODBC. This parameter is required if the DSN parameter is not supplied either at policy level or password level, indicating that the connection method is DSN-Less.

For more details about platform parameters, refer to the Privileged Account Security Reference Guide.

The Connection String template may contain any number of the following variables enclosed with '%' (percentage) sign. These variables will be replaced during run time with the appropriate values:

Variable	Replaced by ...	Taken from
USER	The user name.	Password object properties
OLD PASSWORD	The current password.	Password object properties
LOGON PASSWORD	The password that the plug-in uses for logon. The plug-in automatically detects the password to use for logon. This is either the current password or the password of the reconciliation account.	Password object properties or Reconcile account password object properties
ADDRESS	The address of the database server where the password is used. This parameter is required. If there are several instances on the same machine, specify the instance as part of the address as follows: ADDRESS\INSTANCE. The instance name is case-sensitive and must be specified exactly as it appears.	Password Object properties or, if not defined there, from the ExtralInfo section of the platform.
PORT	The port used by the CPM to access the remote machine.	Password Object properties or, if not defined there, from the ExtralInfo section of the platform.
DATABASE	The database name.	Password Object properties or, if not defined there, from the ExtralInfo section of the platform.
RECONCILE USER	The name of the user who will replace the invalid password with the new password.	Reconcile account password object properties
RECONCILE PASSWORD	The password of the user who will replace the invalid password with the new password.	Reconcile account password object properties

The following example displays a valid connection string on a Microsoft SQL Server:

```
Driver={SQL Server};Server=%ADDRESS%;Database=%DATABASE%;Uid=%USER%;
Pwd=%OLDPASSWORD%;
```

2. Create a temporary DSN and use the testing options in the DSN to test the connection between the CPM machine and the database server.

Configuring the Password Change SQL Statement for Microsoft SQL Passwords

During installation a default SQL statement template is configured for each database vendor whose users will be managed by the CPM during a password change task.

- In the **Additional Policy Settings** section of the platform, check the values of the following **required** parameters:

Parameters	Description
ChangeCommand	The legal SQL statement template that will be used to change the password on the required database.

For more details about platform parameters, refer to the Privileged Account Security Reference Guide.

The statement template can contain any number of the following variables enclosed with '%' (percentage) sign. These variables will be replaced during run time with the appropriate values:

Variable	Replaced by ...	Taken from
USER	The user name	Password Object properties
OLDPASSWORD	The current password	Password Object properties
NEWPASSWORD	The new generated password	Password Object properties
DATABASE	The database name	Password Object properties or, if not defined there, from the Extralfo section of the platform.
RECONCILE USER	The name of the user who will replace the invalid password with the new password.	Reconcile account password object properties
RECONCILE PASSWORD	The password of the user who will replace the invalid password with the new password.	Reconcile account password object properties
LOGON PASSWORD	The password that the plug-in uses for logon. The plug-in automatically detects the password to use for logon. This is either the current password or the password of the reconciliation account.	Password object properties or Reconcile account password object properties

To ensure that these variables will be used as values and not as part of the command, when possible enclose them with quotation marks, as shown in the example below.

The following example displays the command used to change a user password on a Microsoft SQL server:

```
sp_password "%OLDPASSWORD%", "%NEWPASSWORD%"
```

- To reduce the risk of a security hazard, in the **Additional Policy Settings** section of the platform, specify the following parameters:

Parameters	Description
CommandForbiddenCharacters	Characters that cannot be used in the parameters of the change command, listed in the above table. Default values are V@".'{}() -!*>~!^# .
CommandBlackList	Words that cannot be used in the change command, listed in the above table. Default values are delete, drop, exec .

Note: After upgrading to v9.5, add these parameters manually and specify the default values.

Configuring the Password Reconciliation SQL Statement for Microsoft SQL Passwords

During installation a default SQL statement template is configured for each database vendor whose users will be managed by the CPM during a password reconciliation task.

- In the **Additional Policy Settings** section of the platform, check the values of the following **required** parameters:

Parameters	Description
ReconcileCommand	The legal SQL statement template that will be used to reconcile the password on the required database.

For more details about platform parameters, refer to the Privileged Account Security Reference Guide.

The statement template can contain any number of the following variables enclosed with '%' (percentage) sign. These variables will be replaced during run time with the appropriate values:

Variable	Replaced by ...	Taken from
USER	The user name	Password Object properties
OLDPASSWORD	The current password	Password Object properties
NEWPASSWORD	The new generated password	Password Object properties
DATABASE	The database name	Password Object properties or, if not defined there, from the ExtralInfo section of the platform.
RECONCILE USER	The name of the user who will replace the invalid password with the new password.	Reconcile account password object properties
RECONCILE PASSWORD	The password of the user who will replace the invalid password with the new password.	Reconcile account password object properties

Variable	Replaced by ...	Taken from
LOGON PASSWORD	The password that the plug-in uses for logon. The plug-in automatically detects the password to use for logon. This is either the current password or the password of the reconciliation account.	Reconcile account password object properties

To ensure that these variables will be used as values and not as part of the command, when possible enclose them with quotation marks, as shown in the example below.

The following example displays the command used to reconcile a user password on a Microsoft SQL server:

```
sp_password @new="%NEWPASSWORD%", @loginame="%USER%"
```

- To reduce the risk of a security hazard, in the **Additional Policy Settings** section of the platform, specify the following parameters:

Parameters	Description
CommandForbiddenCharacters	Characters that cannot be used in the parameters of the reconcile command, listed in the above table. Default values are V@".'{}() -!*>~!^# .
CommandBlackList	Words that cannot be used in the reconcile command, listed in the above table. Default values are delete, drop, exec .

Note: After upgrading to v9.5, add these parameters manually and specify the default values.

Adding the Password

When you add the password or edit an existing one to be used on an ODBC database, specify the following properties:

- For a DSN connection:

Property	Description
DSN	A legal DSN name that will be used to access the database through ODBC. This parameter is required if the ConnectionStringFile parameter was not supplied in the platform, indicating that the connection method is DSN, or if the DSN parameter in the platform was not specified.

- For a Connection String connection:

Property	Description
Address	The address of the database server that is used as the value of the ADDRESS variable in the Connection String. This parameter is required.
Port	The port number of the database server. This parameter is optional.
Database	The database name that will be used as the value of the DATABASE variable in the Connection String. This parameter is optional.

Password Management Features

The CPM can change and verify Microsoft SQL Server passwords on remote machines, and can reconcile these passwords with both local and domain accounts. If a password is invalid, the CPM can generate a new password and replace the invalid password on the remote machine and its corresponding password in the Password Vault. The parameters that define these tasks are in the platform. A reconciliation account password can be specified either at platform level or at account level.

For more information, refer to *Configuring Accounts for Automatic Management*, page 377.

Sybase Database Accounts

Automatic password management is supported on Sybase database accounts on IPv4 and IPv6.

Supported Platforms

The CPM supports remote password management on a Sybase database on the following platforms:

- Sybase Adaptive Server Enterprise 12.5.2

ODBC Support

The machine that runs the CPM must support ODBC, version 2.7 and higher. If the machine does not support this version of ODBC, download the latest **MDAC_typ.exe** from the Microsoft downloads site.

Required Drivers

The following driver is required on the CPM machine:

- Sybase ODBC driver / Sybase ASE ODBC driver

Make sure that a Sybase ODBC driver is installed on the machine that runs the CPM. If this driver not installed you can install the relevant Sybase Client on the CPM machine.

Platform

In the Platform Management page, make sure that the following target account platform is displayed:

- Sybase

Connection Methods

This plug-in supports the following connection method to connect to remote databases:

- DSN

To Connect to the Remote Database with DSN

1. Create a System DSN for each database in the CPM machine.
2. Use the testing option in the DSN to test the connection between the CPM machine and the database server.

Note: Make sure that the DSN is a system DSN and not a User DSN.

To Connect to the Remote Database with a Connection String (DSN-less)

1. In the **Additional Policy Settings** section of the platform, check the value of the following **required** parameter:

Parameters	Description
ConnectionCommand	The Connection String that will be used to connect to the database through ODBC. This parameter is required if the DSN parameter is not supplied either at policy level or password level, indicating that the connection method is DSN-Less.

For more details about platform parameters, refer to the Privileged Account Security Reference Guide.

The Connection String template may contain any number of the following variables enclosed with '%' (percentage) sign. These variables will be replaced during run time with the appropriate values:

Variable	Replaced by ...	Taken from
USER	The user name.	Password object properties
OLD PASSWORD	The current password.	Password object properties
LOGON PASSWORD	The password that the plug-in uses for logon. The plug-in automatically detects the password to use for logon. This is either the current password or the password of the reconciliation account.	Password object properties or Reconcile account password object properties
ADDRESS	The address of the remote machine where the password is used.	Password Object properties or, if not defined there, from the ExtralInfo section of the platform.
PORT	The port used by the CPM to access the remote machine.	Password Object properties or, if not defined there, from the ExtralInfo section of the platform.
DATABASE	The database name.	Password Object properties or, if not defined there, from the ExtralInfo section of the platform.
RECONCILE USER	The name of the user who will replace the invalid password with the new password.	Reconcile account password object properties
RECONCILE PASSWORD	The password of the user who will replace the invalid password with the new password.	Reconcile account password object properties

The following example displays a valid connection string on a Sybase SQL Server:

```
DRIVER={Sybase ASE ODBC Driver};
NA=%ADDRESS%, %PORT%;UID=%USER%;PWD=%OLDPASSWORD%;
```

The SybaseASEConnectionString.txt file, in the Password Manager\Samples folder, contains a sample SQL statement for this platform.

2. Create a temporary DSN and use the testing options in the DSN to test the connection between the CPM machine and the database server.

Configuring the Password Change SQL Statement for Sybase Passwords

During installation a default SQL statement template is configured for each database vendor whose users will be managed by the CPM during a password change task.

- In the **Additional Policy Settings** section of the platform, check the value of the following **required** parameter:

Parameters	Description
ChangeCommand	The legal SQL statement template that will be used to change the password on the required database.

For more details about platform parameters, refer to the Privileged Account Security Reference Guide.

The statement template can contain any number of the following variables enclosed with '%' (percentage) sign. These variables will be replaced during run time with the appropriate values:

Variable	Replaced by ...	Taken from
USER	The user name	Password Object properties
OLDPASSWORD	The current password	Password Object properties
NEWPASSWORD	The new generated password	Password Object properties
DATABASE	The database name	Password Object properties or, if not defined there, from the ExtralInfo section of the platform.
RECONCILE USER	The name of the user who will replace the invalid password with the new password.	Reconcile account password object properties
RECONCILE PASSWORD	The password of the user who will replace the invalid password with the new password.	Reconcile account password object properties
LOGON PASSWORD	The password that the plug-in uses for logon. The plug-in automatically detects the password to use for logon. This is either the current password or the password of the reconciliation account.	Password object properties or Reconcile account password object properties

To ensure that these variables will be used as values and not as part of the command, when possible enclose them with quotation marks, as shown in the example below.

The following example displays the command used to change a user password on a Sybase database:

```
sp_password "%OLDPASSWORD%", "%NEWPASSWORD%"
```

- To reduce the risk of a security hazard, in the **Additional Policy Settings** section of the platform, specify the following parameters:

Parameters	Description
CommandForbiddenCharacters	Characters that cannot be used in the parameters of the change command, listed in the above table. Default values are <code>"V@".'{}() -!*>~!^#</code> .
CommandBlackList	Words that cannot be used in the change command, listed in the above table. Default values are delete, drop, exec .

Note: After upgrading to v9.5, add these parameters manually and specify the default values.

Configuring the Password Reconciliation SQL Statement for Sybase Passwords

During installation a default SQL statement template is configured for each database vendor whose users will be managed by the CPM during a password reconciliation task.

- In the **Additional Policy Settings** section of the platform, check the value of the following **required** parameter:

Parameters	Description
ReconcileCommand	The legal SQL statement template that will be used to reconcile the password on the required database.

For more details about platform parameters, refer to the Privileged Account Security Reference Guide.

The statement template can contain any number of the following variables enclosed with '%' (percentage) sign. These variables will be replaced during run time with the appropriate values:

Variable	Replaced by ...	Taken from
USER	The user name	Password Object properties
OLDPASSWORD	The current password	Password Object properties
NEWPASSWORD	The new generated password	Password Object properties
DATABASE	The database name	Password Object properties or, if not defined there, from the ExtralInfo section of the platform.
RECONCILE USER	The name of the user who will replace the invalid password with the new password.	Reconcile account password object properties
RECONCILE PASSWORD	The password of the user who will replace the invalid password with the new password.	Reconcile account password object properties
LOGON PASSWORD	The password that the plug-in uses for logon. The plug-in automatically detects the password to use for logon. This is either the current password or the password of the reconciliation account.	Reconcile account password object properties

To ensure that these variables will be used as values and not as part of the command, when possible enclose them with quotation marks, as shown in the example below.

The following example displays the command used to reconcile a user password on a Sybase database:

```
sp_password "%RECONCILEPASSWORD%", "%NEWPASSWORD%", "%USER%"
```

- To reduce the risk of a security hazard, in the **Additional Policy Settings** section of the platform, specify the following parameters:

Parameters	Description
CommandForbiddenCharacters	Characters that cannot be used in the parameters of the reconcile command, listed in the above table. Default values are V@".'{}() - *>~!^# .
CommandBlackList	Words that cannot be used in the reconcile command, listed in the above table. Default values are delete, drop, exec .

Note: After upgrading to v9.5, add these parameters manually and specify the default values.

Adding the Password

When you add the password or edit an existing one to be used on an ODBC database, specify the following properties:

- For a DSN connection:

Property	Description
DSN	A legal DSN name that will be used to access the database through ODBC. This parameter is required if the ConnectionStringFile parameter was not supplied in the platform, indicating that the connection method is DSN, or if the DSN parameter in the platform was not specified.

- For a Connection String connection:

Property	Description
Address	The address of the database server that is used as the value of the ADDRESS variable in the Connection String. This parameter is required.
Port	The port number of the database server. This parameter is optional.
Database	The database name that will be used as the value of the DATABASE variable in the Connection String. This parameter is optional.

Password Management Features

The CPM can change and verify Sybase database passwords on remote machines. If a password is invalid, the CPM can generate a new password and replace the invalid password on the remote machine and its corresponding password in the Password Vault. The parameters that define these tasks are in the platform. A reconciliation account password can be specified either at platform level or at account level.

For more information, refer to *Configuring Accounts for Automatic Management*, page 377.

MySQL Server Accounts

Automatic password management is supported on MySQL Server accounts on IPv4 and IPv6.

Supported Platforms

The CPM supports remote password management on a MySQL Server on the following platforms:

- MySQL version 5 and above

Required Drivers

The following driver is required on the CPM machine:

- Connector/ODBC v3.51.28 (32-bit)

Make sure that a MySQL Connector/ODBC driver is installed on the machine that runs the CPM. If this driver not installed you can download it from MySQL downloads web site (see below).

ODBC Support

The machine that runs the CPM must support ODBC, version 3.51.28 (32-bit). If the machine does not support this version of ODBC, download **the mysql-connector-odbc-3.51.28-win32.msi** from the MySQL downloads site:

<http://dev.mysql.com/downloads/connector/odbc/>

Platform

In the Platform Management page, make sure that the following target account platform is displayed.

- MySQL

Connection Methods

This plug-in supports the following connection methods to connect to remote databases:

- DSN
- Connection string (DSN-less)

To Connect to the Remote Database with DSN

1. Create a System DSN for each database in the CPM machine.
2. Use the testing option in the DSN to test the connection between the CPM machine and the database server.

Note: Make sure that the DSN is a system DSN and not a User DSN.

To Connect to the Remote Database with a Connection String (DSN-less)

1. In the **Additional Policy Settings** section of the platform, check the value of the following **required** parameter:

Parameters	Description
ConnectionCommand	The Connection String that will be used to connect to the database through ODBC. This parameter is required if the DSN parameter is not supplied either at policy level or password level, indicating that the connection method is DSN-Less.

For more details about platform parameters, refer to the Privileged Account Security Reference Guide.

The Connection String template may contain any number of the following variables enclosed with '%' (percentage) sign. These variables will be replaced during run time with the appropriate values:

Variable	Replaced by ...	Taken from
USER	The user name.	Password object properties
OLD PASSWORD	The current password.	Password object properties
LOGON PASSWORD	The password that the plug-in uses for logon. The plug-in automatically detects the password to use for logon. This is either the current password or the password of the reconciliation account.	Password object properties or Reconcile account password object properties
ADDRESS	The address of the database server where the password is used. This parameter is required. If there are several instances on the same machine, specify the instance as part of the address as follows: ADDRESS\INSTANCE. The instance name is case-sensitive and must be specified exactly as it appears.	Password Object properties or, if not defined there, from the ExtralInfo section of the platform.
PORT	The port used by the CPM to access the remote machine.	Password Object properties or, if not defined there, from the ExtralInfo section of the platform.
DATABASE	The database name.	Password Object properties or, if not defined there, from the ExtralInfo section of the platform.
RECONCILE USER	The name of the user who will replace the invalid password with the new password.	Reconcile account password object properties
RECONCILE PASSWORD	The password of the user who will replace the invalid password with the new password.	Reconcile account password object properties

The following example displays a valid connection string on a MySQL Server:

```
Driver={MySQL ODBC 3.51 Driver};server=%ADDRESS%;user=%USER%;option=3;
port=%PORT%;Password=%LOGONPASSWORD%
```

2. Create a temporary DSN and use the testing options in the DSN to test the connection between the CPM machine and the database server.

Configuring the Password Change SQL Statement for MySQL Passwords

During installation a default SQL statement template is configured for each database vendor whose users will be managed by the CPM during a password change task.

- In the **Additional Policy Settings** section of the platform, check the value of the following **required** parameter:

Parameters	Description
ChangeCommand	The legal SQL statement template that will be used to change the password on the required database.

For more details about platform parameters, refer to the Privileged Account Security Reference Guide.

The statement template can contain any number of the following variables enclosed with '%' (percentage) sign. These variables will be replaced during run time with the appropriate values:

Variable	Replaced by ...	Taken from
USER	The user name	Password Object properties
OLDPASSWORD	The current password	Password Object properties
NEWPASSWORD	The new generated password	Password Object properties
DATABASE	The database name	Password Object properties or, if not defined there, from the ExtraInfo section of the platform.
RECONCILE USER	The name of the user who will replace the invalid password with the new password.	Reconcile account password object properties
RECONCILE PASSWORD	The password of the user who will replace the invalid password with the new password.	Reconcile account password object properties
LOGON PASSWORD	The password that the plug-in uses for logon. The plug-in automatically detects the password to use for logon. This is either the current password or the password of the reconciliation account.	Password object properties or Reconcile account password object properties

To ensure that these variables will be used as values and not as part of the command, when possible enclose them with quotation marks, as shown in the example below.

The following example displays the command used to change a user password on a MySQL server:

```
Set password = Password("%NEWPASSWORD%")
```

- To reduce the risk of a security hazard, in the **Additional Policy Settings** section of the platform, specify the following parameters:

Parameters	Description
CommandForbiddenCharacters	Characters that cannot be used in the parameters of the change command, listed in the above table. Default values are <code>"V@".'{}() -!*>~!^#</code> .
CommandBlackList	Words that cannot be used in the change command, listed in the above table. Default values are delete, drop, exec .

Note: After upgrading to v9.5, add these parameters manually and specify the default values.

Configuring the Password Reconciliation SQL Statement for MySQL Passwords

During installation a default SQL statement template is configured for each database vendor whose users will be managed by the CPM during a password reconciliation task.

- In the **Additional Policy Settings** section of the platform, check the value of the following **required** parameter:

Parameters	Description
ReconcileCommand	The legal SQL statement that will be used to reconcile the password on the required database.

For more details about platform parameters, refer to the Privileged Account Security Reference Guide.

The statement template can contain any number of the following variables enclosed with '%' (percentage) sign. These variables will be replaced during run time with the appropriate values:

Variable	Replaced by ...	Taken from
USER	The user name	Password Object properties
OLDPASSWORD	The current password	Password Object properties
NEWPASSWORD	The new generated password	Password Object properties
DATABASE	The database name	Password Object properties or, if not defined there, from the ExtralInfo section of the platform.
RECONCILE USER	The name of the user who will replace the invalid password with the new password.	Reconcile account password object properties
RECONCILE PASSWORD	The password of the user who will replace the invalid password with the new password.	Reconcile account password object properties

Variable	Replaced by ...	Taken from
LOGON PASSWORD	The password that the plug-in uses for logon. The plug-in automatically detects the password to use for logon. This is either the current password or the password of the reconciliation account.	Reconcile account password object properties

To ensure that these variables will be used as values and not as part of the command, when possible enclose them with quotation marks, as shown in the example below.

The following example displays the command used to reconcile a user password on a MySQL server:

```
Set password for "%USER%" = Password("%NEWPASSWORD%")
```

- To reduce the risk of a security hazard, in the **Additional Policy Settings** section of the platform, specify the following parameters:

Parameters	Description
CommandForbiddenCharacters	Characters that cannot be used in the parameters of the reconcile command, listed in the above table. Default values are "V@".'{}() - *>~!^# .
CommandBlackList	Words that cannot be used in the reconcile command, listed in the above table. Default values are delete, drop, exec .

Note: After upgrading to v9.5, add these parameters manually and specify the default values.

Adding the Password

When you add the password or edit an existing one to be used on an ODBC database, specify the following properties:

- For a DSN connection:

Property	Description
DSN	A legal DSN name that will be used to access the database through ODBC. This parameter is required if the ConnectionStringFile parameter was not supplied in the platform, indicating that the connection method is DSN, or if the DSN parameter in the platform was not specified.

- For a Connection String connection:

Property	Description
Address	The address of the database server that is used as the value of the ADDRESS variable in the Connection String. This parameter is required.
Port	The port number of the database server. This parameter is optional.
Database	The database name that will be used as the value of the DATABASE variable in the Connection String. This parameter is optional.

Password Management Features

The CPM can change and verify MySQL Server passwords on remote machines, and can reconcile these passwords with local accounts. If a password is invalid, the CPM can generate a new password and replace the invalid password on the remote machine and its corresponding password in the Password Vault. The parameters that define these tasks are in the platform. A reconciliation account password can be specified either at platform level or at account level.

For more information, refer to *Configuring Accounts for Automatic Management*, page 377.

DB2 Accounts

Automatic password management is supported on DB2 accounts on IPv4 and IPv6.

Supported Platforms

The CPM supports remote password management for IBM DB2 database users on the following platforms:

Supported Windows Platforms

- IBM DB2 on Windows XP, Windows 2000, Windows 2003, WinNT platforms
For details about managing passwords in a Windows environment, refer to *Windows Domain Accounts*, page 384.

Platform

In the Platform Management page, make sure that the following target account platform is displayed:

- DB2Windows

Supported Unix Platforms

- IBM DB2 on the following Unix platforms: Red Hat Linux 8, Red Hat Enterprise Linux ES 3.0, Sun Solaris 5.8, IBM AIX 5, HP-UX 11.x

Connection Methods

On a Unix platform, this plug-in supports the following connection methods to connect to the remote machine:

- SSH
- Telnet

Platform

In the Platform Management page, make sure that the following target account platform is displayed, according to the connection method you have chosen.

- DB2 on Unix via SSH
- DB2 on Unix via Telnet

For more details about managing passwords in a Unix environment, refer to *Unix Accounts*, page 419.

Password Management Features

The CPM can change and verify DB2 passwords on remote machines. If a password is invalid, the CPM can generate a new password and replace the invalid password on the remote machine and its corresponding password in the Password Vault. The parameters that define these tasks are in the platform. A reconciliation account password can be specified either at platform level or at account level.

For more information, refer to *Configuring Accounts for Automatic Management*, page 377.

Informix Accounts

Automatic password management is supported on Informix accounts on IPv4 and IPv6.

Supported Platforms

The CPM supports remote password management for IBM Informix database users on the following platforms:

Supported Windows Platforms

- IBM Informix on Windows XP, Windows 2000, Windows 2003, WinNT platforms

Platform

In the Platform Management page, make sure that the following target account platform is displayed, according to the connection method you have chosen.

- InformixWindows

For details about managing passwords in a Windows environment, refer to *Windows Domain Accounts*, page 384.

Supported Unix Platforms

- IBM Informix on the following Unix platforms: Red Hat Linux 8, Red Hat Enterprise Linux ES 3.0, Sun Solaris 5.8, IBM AIX 5, HP-UX 11.x

Connection Methods

This platform supports the following connection methods to connect to the remote database:

- SSH
- Telnet

Platform

In the Platform Management page, make sure that the following target account platform is displayed, according to the connection method you have chosen.

- Informix on Unix via SSH
- Informix on Unix via Telnet

For more details about managing passwords in a Unix environment, refer to *Unix Accounts*, page 419.

As IBM Informix uses the operating system's users and passwords, its users' passwords can be managed by the existing operating system plug-ins.

Password Management Features

The CPM can change and verify Informix passwords on remote machines. If a password is invalid, the CPM can generate a new password and replace the invalid password on the remote machine and its corresponding password in the Password Vault. The parameters that define these tasks are in the platform. A reconciliation account password can be specified either at platform level or at account level.

For more information, refer to *Configuring Accounts for Automatic Management*, page 377.

Remote Access

HP iLO Accounts

The CPM supports remote password management for HP Integrated Lights-Out, out-of-band management devices on the following versions:

- iLO v2.0, 3.0 and 4.0

Connection Methods

This plug-in supports the following connection methods to connect to the remote machine:

- SSH
- Telnet

Platform

In the Platform Management page, make sure that one of the following target account platforms is displayed, according to the connection method you have chosen.

- HPiLO via SSH
- HPiLO via Telnet

Password Management Features

The CPM can change and verify HP iLO passwords on remote machines, and can reconcile these passwords with local accounts. If a password is invalid, the CPM can generate a new password and replace the invalid password on the remote machine and its corresponding password in the Password Vault. The parameters that define these tasks are in the platform. The reconciliation account can be specified either at platform level or at account level.

For more information, refer to *Configuring Accounts for Automatic Management*, page 377.

Dell DRAC Accounts

The CPM supports remote password management for Dell Remote Access Controller, out-of-band management device of the following version:

- DRAC 5
- DRAC 6

Connection Methods

This plug-in supports the following connection methods to connect to the remote machine:

- SSH
- Telnet

Platform

In the Platform Management page, make sure that one of the following target account platforms is displayed, according to the connection method you have chosen.

- DRAC via SSH
- DELLDRACTelnet

Password Management Features

The CPM can change and verify Dell DRAC passwords on remote machines, and can reconcile these passwords with local accounts. If a password is invalid, the CPM can generate a new password and replace the invalid password on the remote machine and its corresponding password in the Password Vault. The parameters that define these tasks are in the platform. The reconciliation account can be specified either at platform level or at account level.

For more information, refer to *Configuring Accounts for Automatic Management*, page 377.

Security Appliances

CheckPoint Firewall-1 NG Accounts

Supported Platforms

The CPM supports remote password management on CheckPoint Firewall-1 on the following platforms:

- CheckPoint Firewall-1

Platform

In the Platform Management page, make sure that the following target account platform is displayed:

- CheckPoint Firewall-1

Password Management Features

The CPM can change and verify CheckPoint Firewall-1 NG passwords on remote machines and store a corresponding password in the Password Vault. The parameters that define these tasks are in the platform. For more information, refer to *Configuring Accounts for Automatic Management*, page 377.

Supporting SmartCenter users

Currently, only SmartCenter users that have been created through the SmartDashboard application are supported.

Configuring the connection between the CPM machine and the SmartCenter module

In the SmartCenter module, create a client entity.

1. Open the **CheckPoint SmartDashboard** application.
2. In 'Servers and OPSEC applications ...', define a new CPMI client entity
3. Click **Communication** and initialize the communication using the key you supplied during the installation.

For example:

```
CN=my_opsec_app, O=London.mydomain.com. n56gts
```

If you don't remember the key, initialize it with the CheckPoint configuration utility.

4. A DN string appears next to the **Communication** button. Copy this string into the **ClientDN** property of the password object.

For example:

```
CN=my_opsec_app, O=London.mydomain.com. n56gts
```

5. In the **CPMI Permissions**, select **Administrator's Credentials**.

Determining the Server's Distinguished Name

1. Open the **CheckPoint SmartDashboard** application.
2. From the **Network Objects** menu, select the machine that hosts the SmartCenter module; the DN string appears next to the **Communication** button.

For example:

```
CN=cp_mgmt,0=paris.mydomain.com.n79vjo
```

This string will be used in the **ServerDN** property of the password object

Working with a SIC (Secure Internal Communication) file

The SIC file (usually called **opsec.p12**) is used to secure the communication and, therefore, it should be placed on the machine that hosts the CPM.

1. On the CPM machine, from a command line, run the **opsec_pull_cert.exe** utility to retrieve the SIC file from the SmartCenter module machine. This CheckPoint utility can be downloaded from https://supportcenter.checkpoint.com/supportcenter/portal/user/anon/page/default.psml/media-type/html?action=portlets.DCFileAction&eventSubmit_doGetdcdetails=&fileid=7387.
2. In the **SIC cert file** property of the password object, specify the full path and name of the sic certification file. By default, the SIC file is copied to the 'bin' subfolder of the CPM installation folder.

Notes:

1. When the SmartDashboard is activated, it locks the SmartCenter database and changes cannot be carried out on it. Therefore, it is recommended to configure the Firewall-1 password change plug-in to run during hours when the SmartDashboard is not active, such as during night hours.
2. The length of the password is limited to a maximum of 8 characters. A larger password length will cause the third party plug-in to fail.

For further explanations, refer to the CheckPoint OPSEC documentation.

NetScreen Firewall Accounts

Supported Platforms

The CPM supports remote password management in a NetScreen firewall environment on the following platforms:

- NetScreen version 5.3.0r2.0 and above

Connection Methods

This plug-in supports the following connection methods to connect to the remote machine:

- SSH
- Telnet

Platform

In the Platform Management page, make sure that one of the following target account platforms is displayed, according to the connection method you have chosen.

- NetScreen via SSH
- NetScreen via Telnet

Notes:

1. When a user's password is changed, NetScreen closes all open connections that belong to this user. Therefore, it is recommended to configure the CPM to change passwords on NetScreen platforms during a period when users are not connected to NetScreen, for example, during the night. Use the Time Frame options in the NetScreen platform to set the time when the CPM will run.
2. The number of concurrent connections can be limited in NetScreen environments. Specify the relevant value in the MaxConcurrentConnections parameter in the platform.

Password Management Features

The CPM can change and verify Netscreen Firewall passwords on remote machines and store a corresponding password in the Password Vault. The parameters that define these tasks are in the platform. For more information, refer to *Configuring Accounts for Automatic Management*, page 377.

Network Devices

Cisco Router Accounts

Supported Platforms

The CPM supports remote password management on Cisco routers machines on the following platforms:

- Cisco Routers that support IOS 12.3 or higher through Telnet

For the following modes:

- regular user
- enable
- terminal

This platform requires the router to have an installed and operational telnet daemon.

Connection Methods

This plug-in supports the following connection methods to connect to the remote machine:

- SSH
- Telnet

Platform

In the Platform Management page, make sure that one of the following target account platforms is displayed, according to the connection method you have chosen.

- Cisco router via SSH
- Cisco router via Telnet

CiscoUser

- Cisco user without password management privileges:

This password type requires a link to an additional password object that will enable the CPM to switch to 'enable' mode and change the password on the remote machine.

- Cisco user without logon privileges:

This password type requires a link to an additional password object that contains logon information that will enable the CPM to log onto the remote machine where the password will be changed.

Note: A Cisco user password object can be linked to both an enable password object and a logon password object at the same time.

CiscoEnable

- Cisco enable password object:

This password type contains logon information that will enable the CPM to change the password on the remote machine. As different logon information is required to log onto the remote machine, a link is specified to an additional password object that will enable the CPM to log onto the remote machine where the password will be changed.

Note: If you use an additional password to log onto the Cisco machine and change a user's or terminal password, the current user's or terminal password is not used or checked.

CiscoTerminal

- Cisco terminal without password management privileges:

This password type requires a link to an additional password object that will enable the CPM to switch to 'enable' mode and change the password on the remote machine.

- Cisco terminal without logon privileges:

This password type requires a link to an additional password object that contains logon information that will enable the CPM to log onto the remote machine where the password will be changed.

Note: A Cisco terminal password object can be linked to both an enable password object and a logon password object at the same time.

Password Management Features

The CPM can change and verify Cisco router passwords on remote machines. If a password is invalid, the CPM can generate a new password and replace the invalid password on the remote machine and its corresponding password in the Password Vault. The parameters that define these tasks are in the platform. A reconciliation account password can be specified either at platform level or at account level.

For more information, refer to *Configuring Accounts for Automatic Management*, page 377.

Cisco PIX Accounts

Supported Platforms

The CPM supports remote password management on Cisco PIX machines on the following platform:

- Cisco PIX machines, version 6.3 or higher

For the following modes:

- enable
- terminal

Connection Methods

This plug-in supports the following connection methods to connect to the remote machine:

- Telnet
- SSH – for both enable and terminal modes when the logon is with a regular user (not terminal)

Platform

In the Platform Management page, make sure that one of the following target account platforms is displayed, according to the connection method you have chosen.

- Cisco PIX via SSH
- Cisco PIX via Telnet

CiscoEnable

- Cisco enable password object:

This password type contains logon information that will enable the CPM to change the password on the remote machine. As different logon information is required to log onto the remote machine, a link is specified to an additional password object that will enable the CPM to log onto the remote machine where the password will be changed.

Note: If you use an additional password to log onto the Cisco machine and change a terminal password, the current terminal password is not used or checked.

CiscoTerminal

- Cisco terminal without password management privileges:

This password type requires a link to an additional password object that will enable the CPM to switch to 'enable' mode and change the password on the remote machine.

- Cisco terminal without logon privileges:

This password type requires a link to an additional password object that contains logon information that will enable the CPM to log onto the remote machine where the password will be changed.

Notes:

- A Cisco terminal password object can be linked to both an enable password object and a logon password object at the same time.
- If the terminal password object is not linked to a regular user password object or an enable password object for logon, the terminal password can only be changed using Telnet protocol.

Password Management Features

The CPM can change Cisco PIX passwords on remote machines and store a corresponding password in the Password Vault. The parameters that define this task are in the platform. For more information, refer to *Configuring Accounts for Automatic Management*, page 377.

Directories

Novell eDirectory Accounts

Supported Platforms

The CPM supports remote password management on Novell eDirectory machines on the following platforms:

- Novell eDirectory version 8.7.1 SMP or higher

Requirements

This plug-in requires the following Windows Script Host on the CPM machine:

- Cscript 5.6 or higher

By default, Windows XP and Windows 2003 include this Script Host, although Windows 2000 does not. The most recent version can be downloaded from the Microsoft web site under “Windows Script 5.6 for Windows XP and Windows 2000”.

Connection Methods

This plug-in supports the following connection methods to connect to the remote machine:

- LDAP plain protocol (default port – 389)
- LDAP secured protocol (default port – 636)

Platform

In the Platform Management page, make sure that one of the following target account platforms is displayed, according to the connection method you have chosen.

- Novell eDirectory server
- Novell-eDirectorySSL

Password Management Features

The CPM can change and verify Novell eDirectory passwords on remote machines. If a password is invalid, the CPM can generate a new password and replace the invalid password on the remote machine and its corresponding password in the Password Vault. The parameters that define these tasks are in the platform. A reconciliation account password can be specified either at platform level or at account level.

For more information, refer to *Configuring Accounts for Automatic Management*, page 377.

Using SSL to Connect to the Directory

In order to logon to the remote device with SSL, the user running the CyberArk Password Manager service requires a CA certificate that has signed the certificate used by the directory server. By default, the CPM is run by a local system account, therefore, the certificate should be installed in the local computer certificate store.

For more information about how to import a certificate to the local computer store, refer to *To Configure Auto-Detection Processes for SSL Connections*, page 416.

SunOne Directory Accounts

Supported Platforms

The CPM supports remote password management and change on the following SunOne Directories:

- SunOne Directory Server version 5.2

Requirements

The SunOne Device plug-in requires the following Windows Script Host on the CPM station:

- Cscript 5.6 or higher

By default, Windows XP and Windows 2003 include this Script Host, although Windows 2000 does not. The most recent version can be downloaded from the Microsoft web site under “Windows Script 5.6 for Windows XP and Windows 2000”.

Connection Methods

This plug-in supports the following connection methods to connect to the remote machine:

- LDAP plain protocol (default port – 389)
- LDAP secured protocol (default port – 636)

Platform

In the Platform Management page, make sure that the following target account platform is displayed, according to the connection method you have chosen.

- SunOne Directory
- SunOne directory via SSL

Password Management Features

The CPM can change and verify SunOne Directory passwords on remote machines. If a password is invalid, the CPM can generate a new password and replace the invalid password on the remote machine and its corresponding password in the Password Vault. The parameters that define these tasks are in the platform. A reconciliation account password can be specified either at platform level or at account level.

For more information, refer to *Configuring Accounts for Automatic Management*, page 377.

Using SSL to Connect to the Directory

In order to logon to the remote device with SSL, the user running the CyberArk Password Manager service requires a CA certificate that has signed the certificate used by the directory server. By default, the CPM is run by a local system account, therefore, the certificate should be installed in the local computer certificate store.

For more information about how to import a certificate to the local computer store, refer to *To Configure Auto-Detection Processes for SSL Connections*, page 416.

Applications

CyberArk Vault Accounts

Supported Platforms

The CPM supports remote password management in the CyberArk Vault on the following platforms:

- CyberArk Vault v5.0 and higher

Platform

In the Platform Management page, make sure that the following target account platform is displayed:

- CyberArk Vault

Password Management Features

The CPM can change and verify CyberArk Vault passwords on remote machines and store a corresponding password in the Password Vault. The parameters that define these tasks are in the platform. For more information, refer to *Configuring Accounts for Automatic Management*, page 377.

SAP Applications Accounts

Supported Platforms

The CPM supports remote password management and change on the following SAP application server:

- SAP NetWeaver 7.0 (2004s)

Requirements

The SAP Application Passwords plug-in requires the following packages on the CPM machine:

- Microsoft Visual C++ 2005 SP1 Redistributable Package (x86)

To download this package, do the following:

- i. Download **vc redistrib_x86.exe** (version 8.0.50727.762) from the Microsoft website. Use the following link:
<http://www.microsoft.com/downloads/details.aspx?familyid=200B2FD9-AE1A-4A14-984D-389C36F85647&displaylang=en>
- ii. Run the executable to install the package on the CPM machine.

- **SAPNWRFC** (SAP SDK for C/C++).

To download and install this package, do the following:

- i. From the SAP market place web site (service.sap.com), download the following packages:
 - NWRFC_2-20002217.SAR
 - SAPCAR (sapcar.exe)
- ii. Extract the SAR file:
 - `sapcar.exe -xvf NWRFC_2-20002217.SAR`
- iii. Make sure that the **nwrfcsdk/lib** directory was created on your local drive. This directory contains the RFC DLLs that are required for the SAP plug-in operation. The following table lists all the NWRFCSDK DLL Dependencies:

DLL Name	DLL File Version
sapnwrfc.dll	7110.0.15.6533
icudt34.dll	3.4.0.0
icuin34.dll	3.4.0.0
icuuc34.dll	3.4.0.0
libcuddecnumber.dll	7110.0.15.6533
libsapucum.dll	7110.0.15.6533

All the above DLL dependencies are required for the plug-in to work successfully.

- Port **3342** is used for communication between the SAP plug-in on the CPM machine and the SAP system. Make sure this port is open between the two machines.

Platform

In the Platform Management page, make sure that the following target account platform is displayed:

- SAP

Password Management Features

The CPM can change, verify, and reconcile SAP application passwords on remote machines. If a password is invalid, the CPM can generate a new password and replace the invalid password on the remote machine and its corresponding password in the Password Vault. The parameters that define these tasks are in the platform.

For more information, refer to *Configuring Accounts for Automatic Management*, page 377.

Supported Users

This plug-in supports the following users for SAP applications:

- ABAP users (built-in):
 - SAP*
 - DDIC
 - EARLYWATCH
- Java users (built-in):
 - j2ee_admin
- SAP Dialog Users

Configuring SAP Application Users

In the SAP Interface (or any other administration interface):

1. Create the target user that will be managed, if it doesn't exist already.

Note: If you receive an "Inconsistency with Address" error message when working with the user, make sure you have provided all required fields in the user definition. For example, the "Last Name" field is required.
2. Create the logon user that will be used to log onto the SAP application server and manage the target user's credentials.
3. Set the logon user as a privileged user, as follows:
 - i. Create a new role (use /nPFCG).
 - ii. In the Authorization tab, add the name of the new Role to be added or changed (if it already exists).
 - iii. Select Expert mode for profile generation, then click **+Manually** on the menu.
 - iv. Select the following authorization objects and edit the specified fields:

Authorization Object	Field	Value
S_RFC	ACTVT (Action)	16 (execute)
	RFC_NAME (RFC object to be protected)	* (all the function group) To set the minimal function group: <ul style="list-style-type: none"> ▪ SYST ▪ ME_USER ▪ SUSO ▪ SU_USER
	RFC_TYPE (Type of RFC object)	FUGR (Function group)

- v. Update the logon user as follows:
 - Link the Role to the administrator user:
 - a. Select the user, then click **Change**.
 - b. In the 'Role' tab, add the new role.
 - Add the "S_A.SYSTEM" profile to the administrator user:
 - a. Select the user, then click **Change**.
 - b. In the 'Profile' tab, add the S_A.SYSTEM profile.
- Note:** Users in group SUPER can only be maintained by administrators that have the **S_A.SYSTEM** or **SAP_ALL** predefined profiles. However, it is not recommended to give this user the SAP_ALL profile due to security considerations.

Additional Logon Password

The extra logon account, which represents the logon credentials of a privileged SAP user, is mandatory for SAP applications. This account is used to logon to the SAP Server, change the target user password and reconcile the password when needed. You can add a link to it, whether or not it is managed by the CPM.

For more information, refer to *Linked Accounts*, page 220.

Windows Services Accounts

Automatic password management is supported on Windows services accounts on IPv4 and IPv6.

Supported Platforms

The CPM can synchronize a Windows account password with all other occurrences of the same password in different Windows Services, and can manage service dependencies on the following platforms:

- Windows 2000, Windows 2003, Windows 2008/2008R2 with Service Pack 1, Windows 2012/2012R2
- Windows XP, Windows Vista, Windows 7 with Service Pack 1, Windows 8
- Microsoft SQL Server 2005, Microsoft SQL Server 2008
- Microsoft SQL Cluster Service 2005, Microsoft SQL Cluster Service 2008

Note: Other services that are installed as Cluster resources are currently not supported.

For more information, refer to *Managing Service Accounts*, page 151.

Configuring Automatic Management for Windows 2008/Vista

1. On the remote device, enable the following:
 - Client for Microsoft Networks
 - File and Printer Sharing for Microsoft Networks
2. Open the firewall to enable the following:
 - Remote Service Management

3. Configure the user who will connect to the Windows service:
 - Make sure that the user is a domain user and is a member of the 'Administrators group' on the remote machine.or,
 - i. Make sure that the user is a member of the 'Administrator's group'.
 - ii. Disable the UAC for the 'Administrator's group':
 - a. In the Local Security Policy, select **Local Policies**.
 - b. In **Security Options**, disable **User Account Control: Run all administrators in Admin Approval Mode**.

Platform

In the Platform Management page, make sure that the following service account platform is displayed:

- Windows Service

Some Services are required to restart after their password is changed. This plug-in can be configured to restart the Service after the password has been changed successfully. As some Services take some time to start or stop, specify a higher timeout in the plug-in's platform.

Additional Logon Password

If an extra password is required to log onto the machine where the Windows Service is installed, you can add a link to the extra password that will be used to log onto the remote machine. The extra password can be a domain or a local Windows account password. If an extra password is not defined, the Windows Account password will be used to log onto the remote machine.

The user that logs onto the remote machine requires the following permissions:

- Remote access
- Reset passwords

The additional logon user's password may or may not be managed by the CPM.

For more information, refer to *Linked Accounts*, page 220.

Service Dependencies

When working with service dependencies, all services accounts on the remote machine must be managed by the CPM. It is highly recommended to use the auto-detection feature to automatically detect, provision, and manage all service accounts.

During standard service dependency management, if a service is dependent on another service on the same remote machine, when the CPM tries to change the service account password, its service accounts in the Vault will be disabled and a corresponding message will be written in the CPM log. This means that all dependent services will be handled by the root of the dependent services.

In the PVWA, on the CPM tab in the Account page, the following reason will be displayed: (CPM)DependentService.

For more information about configuring service dependencies, refer to the Service Dependencies parameters in the Privileged Account Security Reference Guide.

Windows Scheduled Tasks Accounts

Automatic password management is supported on Windows Scheduled Tasks accounts on IPv4 and IPv6.

Supported Platforms

The CPM can synchronize a Windows account password with all other occurrences of the same password in different Windows scheduled tasks on the following platforms:

- Windows 2000, Windows 2003, Windows 2008/2008R2 with Service Pack 1, Windows 2012/2012R2
- Windows XP, Windows Vista, Windows 7 with Service Pack 1, Windows 8

Note: The CPM must be installed on Windows 2008 R2 with Service Pack 1 server in order to manage Windows Scheduled Tasks on Windows 7, Windows 2008 Server, and Windows Vista.

For more information about multiple copies of a password, refer to *Managing Service Accounts*, page 151.

Note: In Windows 2008, the password for scheduled tasks can be stored separately to the task. The CPM does not support password management for these tasks.

Platform

In the Platform Management page, make sure that the following service account platform is displayed:

- Scheduled Task

Additional Logon Password

If an extra password is required to log onto the machine where the Windows Service is installed, you can add a link to the extra password that will be used to log onto the remote machine. The extra password can be a domain or a local Windows account password. If an extra password is not defined, the Windows Account password will be used to log onto the remote machine.

The user that logs onto the remote machine requires the following permissions:

- Remote access
- Reset passwords

The additional logon user's password may or may not be managed by the CPM.

For more information, refer to *Linked Accounts*, page 220.

Windows IIS Application Pools Accounts

Automatic password management is supported on Windows IIS Application Pools accounts on IPv4 and IPv6.

Supported Platforms

The CPM supports remote password management and change on the following IIS Application Pools:

- Application Pools on IIS 6.0 (Windows 2003)
- Application Pools on IIS 7.0 with “IIS 6 management compatibility” role service (Windows 2008)
- Application Pools on IIS 8.0 with “IIS 6 management compatibility” role service (Windows 2012)
- Application Pools on IIS 8.5 with “IIS 6 management compatibility” role service (Windows 2012R2)

Notes: The Windows IIS Application Pools plug-in cannot change values in an application pool located on the local machine (the machine where the CPM runs).

The CPM supports remote password management for Windows IIS Application Pools Accounts on the following platforms:

- Windows 2003, Windows 2008/2008R2 with Service Pack 1, Windows 2012/2012R2

Requirements

The IIS Application Pool plug-in requires the following Windows Script Host on the CPM machine:

- Cscript 5.6 or higher

By default, Windows XP, Windows 2003 and Windows 2008 include this Script Host, although Windows 2000 does not. The most recent version can be downloaded from the Microsoft web site under “Windows Script 5.6 for Windows XP and Windows 2000”.

Required Authorizations

The user that is used to access the remote machine where the application pool exists can be a local or domain user, but it must be part of the Administrators group. In addition, make sure that the Administrators group has the relevant authorizations on WMI root\microsoftiisv2 namespace, as follows:

1. From the Computer Management console, right-click **WMI Control** and select **Properties**.
2. Select the **Security** tab, then select **MicrosoftIISv2** namespace, then click **Security**; the ‘Security for ...’ window appears.
3. Select the user who will run the plug-in, then select all the permissions and click **OK**.

Configuring Automatic Management for Windows 2008 and higher

1. On the remote device, enable the following:
 - Client for Microsoft Networks
 - File and Printer Sharing for Microsoft Networks
 2. Open the firewall to enable the following:
 - Windows Management Instrumentation (WMI)
 3. Install the 'IIS6 Management Compatibility' role service.
 4. Configure the user who will connect to the IIS Application Pools:
 - Make sure that the user is a domain user and is a member of the 'Administrators group' on the remote machine.
- Or,
- i. Make sure that the user is a member of the 'Administrator's group'.
 - ii. Disable the UAC for the 'Administrator's group':
 - a. In the Local Security Policy, select **Local Policies**.
 - b. In **Security Options**, disable **User Account Control: Run all administrators in Admin Approval Mode**.

Platform

In the Platform Management page, make sure that the following service account platform is displayed

- IIS Application Pool

Additional Logon Password

If an extra password is required to log onto the machine where the Windows Service is installed, you can add a link to the extra password that will be used to log onto the remote machine. The extra password can be a domain or a local Windows account password. If an extra password is not defined, the Windows Account password will be used to log onto the remote machine.

The user that logs onto the remote machine requires the following permissions:

- Remote access
- Reset passwords

The additional logon user's password may or may not be managed by the CPM.

For more information, refer to *Linked Accounts*, page 220.

Windows IIS Directory Security (Anonymous Access) Accounts

Automatic password management is supported on Windows IIS Directory Security accounts on IPv4 and IPv6.

Supported Platforms

The CPM supports remote password management and change for IIS Anonymous passwords on the following platforms:

- Windows 2003, Windows 2008/2008R2 with Service Pack 1, Windows 2012/2012R2

Notes: The Windows IIS Directory Security plug-in cannot change Directory Security values located on the local machine (the machine where the CPM runs).

Requirements

The IIS Anonymous plug-in requires the following Windows Script Host on the CPM machine:

- Cscript 5.6 or higher

By default, Windows 2003, Windows 2008, and Windows 2012 include this Script Host.

Required Authorizations

The user that is used to access the remote machine where the WebSite/virtual directory exists can be a local or domain user, but it must be part of the Administrators group. In addition, make sure that the Administrators group has the relevant authorizations on WMI root\microsoftiisv2 namespace, as follows:

1. From the Computer Management console, right-click **WMI Control** and select **Properties**.
2. Select the **Security** tab, then select **MicrosoftIISv2** namespace, then click **Security**; the 'Security for ...' window appears.
3. Select the user who will run the plug-in, then select all the permissions and click **OK**.

Configuring Compatibility for IIS 6.0

If the IIS on the target machine is higher than IIS 6.0, the target machine must be configured for compatibility with IIS 6.0.

- On Windows 2008, IIS 7.0 is installed by default.
- On Windows 2012, IIS 8.0 is installed by default.
- On Windows 2012R2, IIS 8.5 is installed by default.

The following steps describe how to configure Windows 2008 and higher:

1. In the Server Manager, configure the web server role services:
 - On Windows 2008, right-click on **Web Server (IIS)**, then click **Add Role Services**.
 - On Windows 2012/2012R2, in **Add roles and features** display the Web Server Role (IIS) Services.

2. Install the following IIS 6.0 compatibility services:

- IIS 6 Management Compatibility
- IIS 6 Metabase Compatibility
- IIS 6 WMI Compatibility
- IIS 6 Scripting Tools
- IIS 6 Management Console

Configuring Automatic Management for Windows 2008 and higher

1. On the remote device, enable the following:

- Client for Microsoft Networks
- File and Printer Sharing for Microsoft Networks

2. Open the firewall to enable the following:

- Windows Management Instrumentation (WMI)

3. Install the 'IIS6 Management Compatibility' role service.

4. Configure the user who will connect to the WebSite:

- Make sure that the user is a domain user and is a member of the 'Administrators group' on the remote machine.

Or,

- i. Make sure that the user is a member of the 'Administrator's group'.
- ii. Disable the UAC for the 'Administrator's group':
 - a. In the Local Security Policy, select **Local Policies**.
 - b. In **Security Options**, disable **User Account Control: Run all administrators in Admin Approval Mode**.

5. Restart the remote device.

Platform

In the Platform Management page, make sure that the following service account platform is displayed:

- IIS Anonymous User

Additional Logon Password

If an extra password is required to log onto the machine where the Windows Service is installed, you can add a link to the extra password that will be used to log onto the remote machine. The extra password can be a domain or a local Windows account password. If an extra password is not defined, the Windows Account password will be used to log onto the remote machine.

Note: If the extra logon account is a domain user, the domain name must be specified in the address password property, **not** the IP address.

The user that logs onto the remote machine requires the following permissions:

- Remote access
- Reset passwords

The additional logon user's password may or may not be managed by the CPM.

For more information, refer to *Linked Accounts*, page 220.

COM+ Applications Accounts

Automatic password management is supported on COM+ applications accounts on IPv4 and IPv6.

Supported Platforms

The CPM supports remote password management for COM+ applications on the following platforms:

- Windows 2003, Windows 2008/2008R2 with Service Pack 1, Windows 2012/2012R2

Requirements

Configuring Automatic Management for Windows 2008 and higher

1. On the remote device, enable the following:
 - Client for Microsoft Networks
 - File and Printer Sharing for Microsoft Networks
2. Open the firewall to enable the following:
 - Remote Service Management
3. Configure the user who will connect to the COM+ application:
 - Make sure that the user is a domain user and is a member of the 'Administrators group' on the remote machine.or,
 - i. Make sure that the user is a member of the 'Administrator's group'.
 - ii. Disable the UAC for the 'Administrator's group':
 - a. In the Local Security Policy, select **Local Policies**.
 - b. In **Security Options**, disable **User Account Control: Run all administrators in Admin Approval Mode**.

Configure Network COM+ Access

On the remote device, enable Network COM+ Access:

- **In Windows 2003:**
 1. In the Control Panel, open the **Add or Remove Programs** window, then click **Add/Remove Windows Components**; the Windows Component Wizard appears.
 2. In the Application Server tab, select **Enable network COM+ access**, then click **Next**; the new configuration is set and then the Completing the Windows Components Wizard window appears.
 3. Click **Finish** to complete configuration.
- **In Windows 2008 and 2008R2:**
 1. Open the Server Manager.
 2. Right-click on **Application Server**, then select **Add Role Services**.
 3. Add **COM+ Network Access**.
- **In Windows 2012 and 2012R2:**
 1. In the Server Manager, select the **Application Server**, then display the **Application Server Role Services**.
 2. Add **COM+ Network Access**.

Platform

In the Platform Management page, make sure that the following service account platform is displayed:

- COM-Plus Application

Additional Logon Password

As an extra account is required to log onto the remote machine where the COM+ service account is installed, you can add a link to the extra account. The extra account can be a domain or a local Windows account, but the user must be an administrator on the remote machine. If an extra account is not defined, the Windows Account password will be used to log onto the remote machine.

Note: If the extra logon account is a domain user, the domain name must be specified in the address password property, **not** the IP address.

The user that logs onto the remote machine requires the following permissions:

- Remote access
- Reset passwords

The additional logon user's password may or may not be managed by the CPM.

For more information, refer to *Linked Accounts*, page 220.

Windows Registry Accounts

Supported Platforms

The CPM enables application accounts (such as database accounts) to be changed according to enterprise policy, and can push the new application credentials to the relevant Windows Registry keys and values (e.g. to database connection strings) following password changes, ensuring that applications can continue operating without code changes on the following platforms:

- Windows XP
- Windows Server 2003
- Windows Server 2008/2008R2 with Service Pack 1
- Windows Server 2012/2012R2

The Windows Registry plug-in supports remote password management and change on the following registry trees:

- HKEY_LOCAL_MACHINE
- HKEY_USERS

Notes: The Windows Registry plug-in supports only registry value type String (*REG_SZ*).

The Windows Registry plug-in cannot change values in a registry located on the local machine (the machine where the CPM runs).

Requirements

The Windows Registry plug-in requires the following Windows Script Host on the CPM machine:

- Cscript 5.6 or higher

By default, Windows XP, Windows Server 2003 and Windows Server 2008 include this Script Host. The most recent version can be downloaded from the Microsoft web site under “Windows Script 5.6 for Windows XP and Windows 2000”.

Required Authorizations

The user that is used to access the remote machine where the registry exists can be a local or domain user, but it must be part of the Administrators group. In addition, make sure that the Administrators group has the relevant authorizations on WMI root\default namespace as follows:

1. From the Computer Management console, right-click **WMI Control** and select **Properties**; the Properties window appears.
2. In the **Security** tab, select **Default** namespace, then click **Security**; the ‘Security for ...’ window appears.
3. Select the user who will run the plug-in, then select all the permissions and click **OK**.

Required Protocol

The Windows Registry plug-in requires the following protocol:

- DCOM

This will enable RPC (Remote Procedure Call) so that the CPM can manage registry passwords on remote computers.

1. Open the DCOM port, **TCP135**.
2. Enable remote computer management for the remote computer

Configuring Automatic Management for Windows 2008

1. On the remote device, enable the following:
 - Client for Microsoft Networks
 - File and Printer Sharing for Microsoft Networks
 2. Open the firewall to enable the following:
 - Windows Management Instrumentation (WMI)
 3. Configure the user who will connect to the Windows Registries:
 - Make sure that this user is a domain user and is a member of the ‘Administrators group’ on the remote machine.
- Or,
- Disable the UAC for this user on each of the remote machines where this account will be used as reconcile account.

Platform

In the Platform Management page, make sure that the following service account platform is displayed:

- Windows Registry

Additional Logon Password

You can add a link to the extra password that will be used to log onto the remote machine, which represents the logon credentials the Windows Administrator user on the remote machine. The additional logon user's password may or may not be managed by the CPM.

For more information, refer to *Linked Accounts*, page 220.

Additional Settings

When passwords are part of a larger string, you can specify the relevant prefix and postfix strings in the associated service account platform. This will enable the CPM to identify passwords and replace them.

- To Specify a Prefix for the Registry Value
 1. Click **ADMINISTRATION** to display the **System Configuration** page, then click **Platform Management** to display a list of supported target account platforms.
 2. Select Service Account Platforms, then select the platform to configure, and click **Edit**; the configuration page for the selected platform appears.
 3. Expand **Automatic Password Management**, and then select **Additional Policy Settings**; the prefix property is displayed in the Properties list.
 4. In the **prefix** property, specify the prefix. This is the string that appears before any password occurrence in the registry value. For example, if the registry value is `DSN= cim.dm; UID=Admin; PWD=1234; DBQ=CIM`, specify **PWD=** as the prefix.
- To Specify a Postfix for the Registry Value
 1. Click **ADMINISTRATION** to display the **System Configuration** page, then click **Platform Management** to display a list of supported target account platforms.
 2. Select the platform to configure, then click **Edit**; the configuration page for the selected platform appears.
 3. Expand **Automatic Password Management**, and then select **Additional Policy Settings**; the postfix property is displayed in the Properties list.
 4. In the **postfix** property, specify the longest possible **postfix**. If the postfix contains only one character, this character must be specified in the platform in the **PasswordForbiddenChars** parameter. For example, if the postfix is '&', specify **PasswordForbiddenChars=&**.

For example, if the registry value is `DSN= cim.dm; UID=Admin; PWD=1234; DBQ=CIM`, specify **;DBQ=CIM** as the postfix

Note: If the password inside the registry **contains** the postfix, the CPM will not be able to change the password.

Facebook Accounts

The CPM supports remote password change for user accounts and pages for Facebook's web site on the following platforms:

- Internet Explorer (installed on the CPM machine), versions 10 and 11

Platform

In the Platform Management page, make sure that the following target account platform is displayed:

- Facebook

By default, this platform is configured to manage accounts for Facebook pages. The following instructions describe how to configure the plug-in to manage Facebook user accounts.

1. Click **ADMINISTRATION** to display the **System Configuration** page, then click Platform Management to display a list of supported target account platforms.
2. Select the Facebook platform to configure, then click **Edit**; the configuration page for the platform appears.
3. Display the **Extra Info** parameters.
4. In the **Type** value, change the value from **Page** to **User**.

Configuring the Facebook account to use HTTPS

1. Log in to Facebook using the account to configure, and display the **Account Settings**; the Account Settings page appears.
2. Click **Security**; the **Security Settings** page appears.
3. In the **Secure Browsing** line, click **Edit**; the secure browsing option is displayed.
4. Select **Browse Facebook on a secure connection (https) when possible**, then click **Save Changes**; the new security settings are enabled and the Security Settings page is displayed again.

Re-enabling this Plug-in

This plug-in is automatically disabled when it encounters a CAPTCHA security check. In order to resume work, do the following:

- Login to Facebook from the CPM machine and answer the question,
and
- Delete all cookies and history on the CPM machine.

Managing Accounts Stored in Configuration Files

The CPM enables organizations to manage application accounts that are stored externally to the Vault in the following types of files:

- **Plain text** – Passwords in plain text files can be specified anywhere in any format. The CPM identifies passwords using a regular expression.
- **INI files** – Passwords in ini files are specified in a particular section and parameter. The CPM uses these details to identify passwords and change them.
- **XML files** – Passwords in xml files are specified in XmlElements. The CPM uses XPath for these XmlElements and XmlAttributes to identify passwords and change them.
- **Web configuration files** – Passwords are stored in the same way as in xml files. However, any changes made in web configuration files cause the application to restart. Account management for accounts that are managed with the web configuration service account must be initiated manually in the PVWA so that users can control the application restart.

In all of the above file formats, the password value can be stored in either of the following ways:

- **Direct account password** – The password stored in the configuration file is exactly the same as the password stored in the Vault.
- **Encrypted account password** – The password stored in the configuration file is an encrypted version of the password stored in the Vault.

Connection Protocols

This plug-in supports the following protocols to connect to the remote machine:

- **Windows** – Windows file and printer sharing protocol
- **Unix** – SSH protocol

Required Authorizations

The account used to access the file requires the following permissions to access the configuration file on the remote machine:

- **Windows:**
 - **Permissions**

The CPM can connect to a remote machine using a logon account that has the following permissions:

 - View the configuration file
 - Edit the configuration file
 - Create files (if the CPM is configured to create a backup password file)

If the password will be encrypted, the local system account used to run the CPM service requires the following file system permissions for the encryption program executable:

 - Administrative permissions
 - **Communication**
 - On the remote machine, make sure that the following options are enabled:
 - Client for Microsoft Networks
 - File and Printer Sharing for Microsoft Networks
 - On the CPM machine, if DEP is supported, disable it.

- **Unix**
 - **Permissions**

The CPM can connect to a remote machine using a Unix/Linux connection with a logon account that has the following permissions:

 - SSH login permissions
 - View the password file
 - Edit the password file
 - Create files (if the CPM is configured to create a backup password file)
 - **Communication**
 - Open port **22** to enable SSH and SFTP activities.

Platform

In the Platform Management page, make sure that one of the following service account platforms is displayed, depending on the service account to implement:

- **Text Config File** – To manage passwords stored in plain text files.
- **INI Config File** – To manage passwords stored in INI files.
- **XML Config File** – To manage passwords stored in XML files.
- **Web Config File** – To manage passwords stored in web configuration files.

Encrypting Passwords

Passwords stored in configuration files can be encrypted using an external command.

To Encrypt Passwords in Configuration Files

1. In the Platform Management page, select Service Account Platforms.
2. Select the platform that will manage the service account, and click Edit; the service account settings page appears.
3. Display the **Additional Policy Settings**, and specify the following parameters:
 - **Encryption Command** – The full path of the encryption command that will encrypt the password. The encryption file can be stored in any location on the CPM machine.

This command sends the current password as its parameter. For example, if you specify "C:\Program Files (x86)\CyberArk\Password Manager\bin\EncExe", the actual command would be "C:\Program Files (x86)\CyberArk\Password Manager\bin\EncExe <currpass>", and the output would be the new encrypted password.

If the current password parameter is empty, the original new password will be inserted in the file.
 - **Encryption Regex** - The regex parameter that handles the output of the Encryption Command parameter. If this parameter is not defined, it will behave as if "(.*)" has been specified. This parameter is only relevant when the Encryption Command parameter is defined.
4. Click **Apply** to save the new platform values and stay in the settings page, or,
Click **OK** to save them and return to the System Configuration page.

Additional Logon Passwords

If the CPM will manage a password in a configuration file on a remote machine that requires an additional password to log onto the remote machine, create an additional password object to contain the extra user's logon details, and link it to the original password object. For more information, refer to *Linked Accounts*, page 220.

Usage

In the relevant platform, in the UI & Workflows parameters, add the ID of the service account in the list of Usages. For more information, refer to *Service Account Platforms*, page 114.

Adding Service accounts

When you add service account for an account stored in a configuration file, specify the following parameters:

To Add a TextConfigFile Service Account

1. In the Account Details page, display the **TextConfigFile** service account pane.
2. Specify the following required properties:

Property	Description
Address	The address of the remote machine where the Text configuration file is located.
File Path	The full file path including name and extension of the configuration file that contains the password. For example, in Unix, "/home/passwords/filename". For example in Windows, "C:\NewShare\passwords\FileName.txt".
Password Regex	The password regular expression that matches the line of the password in the configuration file. Specify a prefix and a postfix to allow the CPM to identify the entire line, and specify the password in parenthesis "(" and ")". This single-groups the password and defines the location of the password to change. Example 1: Specify Password=(.*) ; to replace the new password with all the characters between Password= and ; (semi-colon). Example 2: Specify Password=(.....) ; to replace the new password with eight characters after Password= . If the ; (semi-colon) is not found after eight characters, this regex will not match the line in the password file and will not replace the password successfully. The CPM will replace all occurrences of the specified regex in the password file. If the password regex is empty, the CPM will return an error message.
ConnectionType	The connection type that will be used to access the target machine where the configuration file resides. Valid values are: <ul style="list-style-type: none"> ▪ Windows File Sharing ▪ SSH

3. Specify the following optional properties:

Property	Description
Port	The port that is used for non-standard SSH ports. Default is 22 if not set. This parameter is only relevant for SSH protocol.
Backup Password File	Whether or not a backup configuration file will be created. Specify Yes or No . When this parameter is set to Yes, the password file will be saved in a subfolder of the location where the password file is stored, before the password content is changed. The backup file will be named "Backup__ %fileName%" where %fileName% will be the suffix after the last trailing '/' or '\' and will reside in the same directory as the original password file.

To Add an INIConfigFile Usage

1. In the Account Details page, display the **INIConfigFile** pane.
2. Specify the following required properties:

Property	Description
Address	The address of the remote machine where the INI configuration file is saved.
File Path	The full file path including name and extension of the configuration file that contains the password. For example, in Unix, "/home/passwords/filename". For example in Windows, "C:\NewShare\passwords\File Name.ini".
INI Section	The ini section that contains the password string. If more than one ini sections contain a password to manage, create multiple service accounts for this account or use the TextConfigFile usage.
INI Parameter Name	The name of the parameter in the configuration file that contains the password. If more than one parameter in the same ini section contains a password to manage, create multiple service accounts for this account or use the TextConfigFile usage.
ConnectionType	The connection type that will be used to access the target machine where the configuration file resides. Valid values are: <ul style="list-style-type: none"> ▪ Windows File Sharing ▪ SSH

3. Specify the following optional properties:

Property	Description
Port	The port that is used for non-standard SSH ports. Default is 22 if not set. This parameter is only relevant for SSH protocol.
Backup Password File	Whether or not a backup configuration file will be created. Specify Yes or No . When this parameter is set to Yes, the password file will be saved in a subfolder of the location where the password file is stored, before the password content is changed. The backup file will be named "Backup__ %fileName%" where %fileName% will be the suffix after the last trailing '/' or '\' and will reside in the same directory as the original password file.

To Add an XMLConfigFile Usage

1. In the Account Details page, display the **XMLConfigFile** pane.
2. Specify the following required properties:

Property	Description
Address	The address of the remote machine where the XML configuration file is located.
File Path	The full file path including name and extension of the configuration file that contains the password. For example, in Unix, "/home/passwords/fileName". For example in Windows, "C:\NewShare\passwords\FileName.xml".
XML Element	The XPath that represents the xml element which contains the text/attribute to change. For more information about XPath, refer to http://www.w3schools.com/XPath/xpath_syntax.asp .
ConnectionType	The connection type that will be used to access the target machine where the configuration file resides. Valid values are: <ul style="list-style-type: none"> ▪ Windows File Sharing ▪ SSH

3. Specify the following optional properties:

Property	Description
Port	The port that is used for non-standard SSH ports. Default is 22 if not set. This parameter is only relevant for SSH protocol.
Backup Password File	Whether or not a backup configuration file will be created. Specify Yes or No . When this parameter is set to Yes, the password file will be saved in a subfolder of the location where the password file is stored, before the password content is changed. The backup file will be named "Backup__ %fileName%" where %fileName% will be the suffix after the last trailing '/' or '\' and will reside in the same directory as the original password file.

Property	Description
Password Regex	<p>The password regex that matches the line of the password in the configuration file. Specify a prefix and a postfix to allow the CPM to identify the entire line, and specify the password in parenthesis "(" and ")". This single-groups the password and defines the location of the password to change.</p> <p>Example 1: Specify Password=(.*); to replace the new password with all the characters between Password= and ; (semi-colon).</p> <p>Example 2: Specify Password=(.....); to replace the new password with eight characters after Password=. If the ; (semi-colon) is not found after eight characters, this regex will not match the line in the password file and will not replace the password successfully.</p> <p>The CPM will replace all occurrences of the specified regex in the password file.</p> <p>If the password regex is empty, the CPM will return an error message.</p>
XML Attribute	<p>The xml attribute of the xml element (defined in the XML Element parameter) to change. If this parameter is empty, the new password value will be written in the text of the xml element.</p>

To Add a WebConfigFile Usage

1. In the Account Details page, display the **XMLConfigFile** pane.
2. Specify the following required properties:

Property	Description
Address	The address of the remote machine where the WebConfig configuration file is located.
File Path	<p>The full file path including name and extension of the file that contains the password.</p> <p>For example, in Unix, "/home/passwords/filename".</p> <p>For example in Windows, "C:\NewShare\passwords\FileName.xml".</p>
XML Element	The XPath that represents the xml element which contains the text/attribute to change.
ConnectionType	<p>The connection type that will be used to access the target machine where the configuration file resides. Valid values are:</p> <ul style="list-style-type: none"> ▪ Windows File Sharing ▪ SSH

3. Specify the following optional properties:

Property	Description
Port	<p>The port that is used for non-standard SSH ports. Default is 22 if not set.</p> <p>This parameter is only relevant for SSH protocol.</p>

Property	Description
Backup Password File	<p>Whether or not a backup configuration file will be created. Specify Yes or No.</p> <p>When this parameter is set to Yes, the password file will be saved in a subfolder of the location where the password file is stored, before the password content is changed.</p> <p>The backup file will be named "Backup__ %fileName%" where %fileName% will be the suffix after the last trailing '/' or '\' and will reside in the same directory as the original password file.</p>
Password Regex	<p>The password regex that matches the line of the password in the configuration file. Specify a prefix and a postfix to allow the CPM to identify the entire line, and specify the password in parenthesis "(" and ")". This single-groups the password and defines the location of the password to change.</p> <p>Example 1: Specify Password=(.*); to replace the new password with all the characters between Password= and ; (semi-colon).</p> <p>Example 2: Specify Password=(.....); to replace the new password with eight characters after Password=. If the ; (semi-colon) is not found after eight characters, this regex will not match the line in the password file and will not replace the password successfully.</p> <p>The CPM will replace all occurrences of the specified regex in the password file.</p> <p>If the password regex is empty, the CPM will return an error message.</p>
XML Attribute	<p>The xml attribute of the xml element (defined in the XML Element parameter) to change. If this parameter is empty, the new password value will be written in the text of the xml element.</p>

Running Multiple Service Accounts Concurrently

At any time, only one configuration file service account process can manage a specific configuration file. If multiple processes specify the same Address and FilePath properties, the processes will run consecutively. However, if two service accounts specify the same configuration file, but one specifies an IP address and the other specifies the FQDN address, the CPM might run these service accounts simultaneously, which may cause an error.

Managing Comments in INI Configuration Files

Comments that appear in the same line as passwords will be managed according to the following guidelines:

- Lines that start with '#' (hash) or ';' (semi-colon) are line comments.
- The INIConfigFile service account recognizes inline comments that are marked with a space followed by ';' (space followed by a semi-colon), and will treat them as comments, as shown in the following example:

Previous password: Password=Abc123 ; comment

New password: Password=Def456 ; comment

- The INIConfigFile service account **does not** recognize inline comments that are marked with a hash (#), and will treat them as part of the password value.

- Semi-colons that are not preceded by a space can be included in password values managed by the INIConfigFile service accounts, as shown in the following example:

Previous password: `Password=Abc;123`

New password: `Password=Def456`

Note: In lines that use a semi-colon as part of a password, you cannot specify inline comments with ' ;' (space followed by a semi-colon).

Managing Accounts Stored in Databases

The CPM enables organizations to manage application accounts that are stored externally to the Vault in database tables. Implementing this plug-in will help you eliminate hardcoded application accounts stored within specific database tables.

Supported Platforms

The CPM supports remote password management on Oracle databases on the following platforms:

- Microsoft SQL Server 7 and above
- Oracle Database V8i or higher

Note: If this version of Oracle does not support the version of Oracle that is installed in your enterprise, install a different version of the Oracle Instant Client on the CPM machine.

ODBC Support

The machine that runs the CPM must support ODBC, version 2.7 and higher. If the machine does not support this version of ODBC, download the latest **MDAC_typ.exe** from the Microsoft downloads site.

During CPM installation, the Oracle Instant Client, version 11, is installed. This enables the CPM to use the Oracle ODBC driver, and connect to Oracle databases without the need for TNSNames.

Platform

In the Platform Management page, make sure that the following service account platform is displayed:

- Database String platform

Connection Methods

This plug-in supports the following connection methods to connect to remote databases:

- DSN

To Connect to the Remote Database with DSN

1. Create a System DSN for each database in the CPM machine.
2. Use the testing option in the DSN to test the connection between the CPM machine and the database server.

Note: Make sure that the DSN is a system DSN and not a User DSN.

Configuring the SQL Statement Template

The SQL statement template enables users to perform specific or complicated SQL commands.

The SQL statement includes account properties or parameters specified in the Additional Policy Settings section in the Database String platform. It can contain any number of these variables, enclosed within '%' (percentage) signs. These variables are replaced during run time with the appropriate values.

The following table lists the variables that can be specified:

Property	Description
DSN	The name of the DSN connection that will be used.
Address	The address of the remote machine.
Port	The port used to access the remote machine.
Database	The name of the database that contains the passwords to manage.
TableName	The name of the table on the remote database that contains the passwords to manage.
ColumnName	The name of the column that contains passwords in the table in the remote database.
UniqueIDColumnName	The unique primary key column name that will be used to identify specific records containing passwords for the account.
UniqueIDColumnValue	The unique primary key column value that will be used to identify specific records containing passwords for the account

The statement template must start and end on the same line without any line separators. It cannot be a multiline statement.

To ensure that these variables will be used as values and not as part of the command, when possible enclose them with quotation marks, as shown in the example below.

For example, to add the user name property to the statement template, use the following pattern:

```
"%UserName%"
```

The following example shows the command used to change a user password on an SQL database server:

```
update %TableName% set %ColumnName% =
'%prefix%%NewPassword%%postfix%' where %UniqueIDColumnName% =
'%UniqueIDColumnValue%'
```

To Configure the Statement Template File

1. In the Database String platform, in the **Additional Policy Settings** section, check the value of the following parameter:

Parameters	Description
ChangeCommand	The legal SQL statement that will be used to change the string on the required database.
ConnectionCommand	The SQL statement that will be used as a connection string. This statement must be the same as that used in the associated master account. This parameter is required if the DSN parameter is not supplied either at platform level or password level, indicating that the connection method is DSN-Less.

For more details about platform parameters, refer to the Privileged Account Security Reference Guide.

2. Edit the default SQL statement template so that it includes account properties and/or parameters specified in the Additional Policy Settings section in the Database String platform.
 - To reduce the risk of a security hazard, in the **Additional Policy Settings** section of the platform, specify the following parameters:

Parameters	Description
CommandForbiddenCharacters	Characters that cannot be used in the parameters of the change command, listed in the above table. Default values are "V@".'{}() -!*>~!^# .
CommandBlackList	Words that cannot be used in the change command, listed in the above table. Default values are delete, drop, exec .

Note: After upgrading to v9.5, add these parameters manually and specify the default values.

3. Execute the full statement manually on the database server.
4. After making sure that the new statement works correctly, click Save to save the statement template in the DBString policy.

Usage

In the relevant platform, in the UI & Workflows parameters, add the following ID of the Database String service account in the list of Usages:

- DBString

For more information, refer to *Service Account Platforms*, page 114.

Password Management Features

The CPM can change passwords on remote machines. If a password is invalid, the CPM can generate a new password and replace the invalid password on the remote machine and its corresponding password in the Password Vault. The parameters that define these tasks are in the platform. For more information, refer to *Configuring Accounts for Automatic Management*, page 377.

Additional Logon Passwords

If the CPM will manage a password in a configuration file on a remote machine that requires an additional password to log onto the remote machine, create an additional password object to contain the extra user's logon details, and link it to the original password object. For more information, refer to *Linked Accounts*, page 220.

Adding Service Accounts

When you add a service account for an account stored in a database table, specify the following parameters:

1. In the Account Details page, display the **DatabaseString** pane.
2. Specify the following required property:

Property	Description
UsageDisplayName	The display name for the service account defined by the user

3. Specify the following optional properties:

Property	Description
DSN (ODBC)	The name of the DSN connection that will be used.
Address	The address of the remote machine.
Port	The port used to access the remote machine.
Database	The name of the database that contains the passwords to manage.
TableName	The name of the table on the remote database that contains the passwords to manage.
ColumnName	The name of the column that contains passwords in the table in the remote database.
UniqueIDColumnName	The unique primary key column name that will be used to identify specific records containing passwords for the account.
UniqueIDColumnValue	The unique primary key column value that will be used to identify specific records containing passwords for the account

4. Click **Save**; the database string is added to the list of files for automatic management.

Managing Accounts through a Web Interface

The CPM enables organizations to manage accounts for applications that have a 'web only' management interface. CyberArk's generic web plug-in enables CyberArk professional services to develop custom plug-ins for specific customers to manage web interface passwords, opening a whole new world of systems and devices that can be managed by the Privileged Account Security solution.

This replaces APIs and CLIs that cannot manage password changes on web sites or with tools that enforce a web-only management interface.

For more information about managing accounts in web applications, contact your CyberArk support representative.

Cloud Services

Amazon Web Services (AWS) Accounts

The CPM supports remote password management for user accounts on the following platforms:

- Amazon Web Services (AWS)

Platform

In the Platform Management page, make sure that the following target account platform is displayed:

- AWS Management Console

Password Management Features

The CPM can change and reconcile AWS passwords on remote machines. If a password is invalid, the CPM can generate a new password and replace the invalid password on the remote machine and its corresponding password in the Password Vault. The parameters that define these tasks are in the platform. A reconciliation account password can be specified either at platform level or at account level.

You can also configure the CPM to perform periodic password reset operations (using reconciliation accounts) instead of a password change operation. This is configured at platform level.

In the associated logon and reconciliation account, specify the following account properties:

- Access Key
- Secret Access Key

Note: To use the AWS CPM Plug-in to change and/or reconcile credentials, you must specify a username.

For more information, refer to *Configuring Accounts for Automatic Management*, page 377.

Configuring Service Accounts

You can configure additional service accounts that are required for use in different resources, such as Windows services or Windows scheduled tasks. These service accounts are referred to in the platform parameters. Required and optional properties, displayed columns, and linked accounts are set for each usage.

Adding Service Account Platforms

To Add a New Service Account

1. Click **ADMINISTRATION** to display the **System Configuration** page, then click Platform Management to display a list of supported target account platforms.
2. Click **Service Account Platforms** to display a list of supported service account platforms.
3. Select an existing platform that is similar to the new service account platform, then click **Duplicate**; the Duplicate Platform window appears.
4. Type the name and a description of the new service account platform, then click **Save & Close** to create the new platform.
5. Select the new service account platform, and then click **Edit**; the configuration page for the selected platform appears.
6. Change existing parameter values and/or add new values to define the new platform.
7. Click **Apply** to save the new configurations and apply them immediately,
or,
Click **Save** to save the new configurations and apply them after the period of time specified in the **RefreshPeriod** parameter.

Deleting Service Account Platforms

If you are sure that you will not support accounts on a certain service account platform, you can delete the platform.

To Delete a Service Account Platform

1. In the list of Service Account Platforms, select the platform to delete, then click **Delete**; a confirmation message appears.
2. Click **Yes** to delete this platform,
or,
Click **No** to close the message and return to the Service Account Platforms list.

Password Vault Web Access

The Password Vault Web Access enables both end users and administrators to access and manage privileged accounts from any local or remote location through a web client.

This chapter describes how to configure the Password Vault Web Access application and begin working with it.

This chapter comprises the following sections:

- Configuring the PVWA
- Configuring the Mobile Password Vault Web Access
- Direct Access to PVWA Pages
- Logging
- Disaster Recovery

Configuring the PVWA

The PVWA can be configured to provide a variety of functions that enable you to manage your Privileged Account Security implementation. All configurations can be specified by authorized users in the ADMINISTRATION page. For more information, refer to *Configuring the System through PVWA*, page 918.

This section describes how to configure and implement the following:

- General PVWA Configurations
- Configuring the Accounts List
- User Preferences
- Adding Safe Members from LDAP Directories
- Searching for Accounts
- Identifying Accounts
- Configuring Statistics
- Accounts Check-out and Check-in
- Dual Control
- Integrating with Ticketing Systems
- Configuring Dual Control together with Ticketing Integration
- Configuring Transparent Connections
- Accessing PVWA from another Web Application
- Reports
- Logging
- Caching
- Configuring the Account Retrieval Form
- Configuring Split Password Mode
- Adding Accounts
- Adding Safes
- Configuring the PVWA Interface

General PVWA Configurations

The following parameters, in the **General** parameters of the Web Access Options, define general parameters for the PVWA.

Viewing Activities

The following parameter enables you to view the activities that have been carried out on an account in the Activities pane in the Account Details page.

- The **ActivityReportPeriod** determines the default number of days that will be included in the list of activities in the Activities pane.

Viewing Accounts

The following parameters, in the **General** parameters of the Web Access Options, determine the account properties that will be displayed, how the password value will be displayed, and the password management errors that can be viewed in the Password Vault Web Access.

Displaying Password Values

- The **PasswordRevealingTime** parameter determines the number of seconds that the password will be displayed on the screen.

Displaying the Platform Name

- The **DisplayPolicyNameInList** parameter determines whether or not the platform name will be displayed in the Accounts list. If this parameter is not specified, or if it is set to 'No', the platform name will appear.

Viewing CPM Errors

- If the CPM does not change a password successfully, a warning message is displayed. The **CPMErrorsToDisplay** parameter determines the number of recent CPM error messages that will be displayed when you click on the warning message.

Controlling Column Width in the PVWA

The following parameters, in the **Account Display Columns** parameters, define the properties that will appear in the accounts list. The order of the properties determines the order in which this information will be displayed.

The width of the columns in the PVWA can be set to a specific width. In this way, users can see the entire content of columns that contain important information without having to resize them each time.

- **Name** – The name of the column. This may be any file categories except internal file categories or any of the following: Safe, Folder, Name, LastUsed, LastUsedBy, LastModified, LastModifiedBy, CreatedBy, CreatedAt, Size. A maximum of five column widths may be specified.
- **Width** – The size of the column in pixels.

Viewing Files

The Files List

The following parameters determine the file properties that will be displayed in the Password Vault Web Access.

- The **File Display Columns** parameters specify the file properties that will appear in the files lists. The order in which the properties appear determines the order in which this information will be displayed in the list.
- The **DisplayFileOthersColumn** parameter, in the **General** parameters, determines whether or not an additional column in the files list displays the properties that are not specified in the 'File Display Columns' parameters.

Retrieving and Storing Files

The following parameters determine how files in the Password Vault can be accessed and stored through the Password Vault Web Access.

- The **AllowOpenFiles** parameter, in the **General** parameters, determines whether or not users will be able to open files through a linked filename or an icon.
Note: If you run PVWA on an IE browser, it is not recommended to enable this parameter, due to a potential browser security issue.
- The **maxRequestLength** parameter determines the maximum size of requests that are sent to the IIS, limiting the size of files that can be uploaded by the Password Vault Web Access. This parameter is specified in the web.config file.

File Download Timeout

- The **FileDownloadTimeout** parameter, in the **General** parameters, determines the maximum number of minutes that is allocated to downloading a file, after which the download will fail.

Sending Links to Accounts and Files

In the Password Vault Web Access, you can send links by email to accounts and files that are stored in the Password Vault. This option opens the user's default email application and creates a new message with the link in it. The following parameters, in the **General** parameters, enable you to configure the contents of the message:

- The **PasswordLinkSubject** parameter specifies the text that appears in the subject field of a message that contains a link to an account in the Password Vault.
- The **FileLinkSubject** parameter specifies the text that appears in the subject field of a message that contains a link to a file in the Password Vault.

Refreshing Interval Settings

- The **RefreshPeriod** parameter, in the **General** parameters, determines how frequently (in minutes) the Web Access configurations are read by the Password Vault Web Access. Any configuration changes will only take effect after the specified number of minutes has passed and the configurations have been re-read. To apply changes immediately, click **Apply**.

Displaying New and Locked Accounts and Files

The PWWA can update the accounts and files lists automatically to display new accounts that have been added to the Vault, as well as accounts that have been locked by other users. The lists will only display new and locked accounts and files in the Safes where the user who is logged on has ownership.

The following properties, in the **General** parameters, determine which new accounts are displayed in the New Accounts view.

- The **AutoCheckForNewAndLockedObjects** parameter determines whether or not the PWWA will automatically check the Vault for new and locked accounts and files to include in the accounts/files list. The default value is **No**.
- The **NewAccountsHistoryPeriod** parameter determines the number of days during which new accounts were created that are displayed in the New Accounts view in the Accounts page. This number of days appears in the tooltip that appears when users point to **New Accounts**. The default value is **14**.

If the PWWA is not configured to update accounts and files lists automatically, the user will be able to initiate a list update manually in the PWWA.

Removing White Spaces from Passwords

The PWWA can remove the white spaces which are included inadvertently at the beginning or end of passwords when accounts are added or changed manually. White spaces includes non-printing characters such as space, tab, line feeds, etc.

- The **RemoveWhitespacesFromPasswords** parameter determines whether or not the PWWA will remove leading and trailing white spaces from passwords that are specified manually.

Viewing CPM Details

The following parameters, in the **General** parameters, enable you to control the contents of the CPM pane in the Account Details or File Details page.

- The **DisplayGroupMembersInObjectDetails** parameter determines whether or not group members will be displayed in the CPM tab. The default value is **No**.
- The **RequireManageSafeToClearLinkedAccount** parameter determines whether or not users require the **Manage Safe** permission in order to enable the **Clear** button to disassociated linked accounts. The default value is **No**.

Managing CPM Activity

The following parameters, in the **General** parameters, enable you to control CPM activity.

- The **FirstDayOfWeek** parameter defines the first day of the week for the CPM. The default value is **Monday**.
- The **ManualChangeGroupMembersLimit** parameter defines the maximum number of accounts in a group to support when changing the password of the group directly in the Vault (without CPM involvement). The default value is **500**.
- The **MaxDisplayedGroupMembers** parameter defines the maximum number of group members that will be displayed in the CPM tab when the user clicks 'Display'. The default value is **100**.
- The **CPMFailureEventDays** parameter determines the number of days' events to display in the CPM activity log window. The default value is **14**.

Displaying a PWVA Startup Message

You can customize the PVWA to display a message in the PVWA authentication page. You can specify any message you want to display.

The following parameters, in the **General** parameters, configure the message box, and determine the message that will be displayed.

- The **DisplayUserLoginMessage** parameter determines whether or not a message will be displayed in the PVWA logon page. The default value is **No**.
- The **UserLoginBackground** parameter determines the background RGB color of the message box. The default value is **#ffffff**.
- The **UserLoginMessage** parameter specifies the message that will be displayed. There is no default message.
- The **UserLoginMessageWidth** parameter determines the width of the message box. The default value is **290** pixels.
- The **UserLoginMessageHeight** parameter determines the height of the message box. The default value is **200** pixels.

Displaying the User's Time Zone

The following parameter, in the **General** parameters, enables you to configure the PVWA to display the time zone on the web server as part of the date every time the time is displayed, including account activity, requests, auditing, PSM, etc.

- The **DisplayTimezoneInDates** parameter determines whether or not the time zone on the web server will be displayed as part of the date format.

Note: After changing this parameter, restart the PVWA.

Authenticating to the Vault

The following parameter, in the **General** parameters, defines the source address used to authenticate to the Vault from the PVWA. This affects network areas authentication.

- The **AuthenticationSourceAddress** parameter determines whether the source address used to authenticate users to the Vault will be taken from the PVWA's server or the end users' IP address. The default value is **Client Address**, indicating the end user's PVWA address is used to authenticate to the Vault.

The following parameter, in DBParm.ini, enables the PVWA to automatically manage users whose Vault user is suspended after they tried to log onto the PVWA with a wrong password more than the maximum permitted number of login violations, and they can logon again successfully after a predefined time period.

- The **UserLockoutPeriodInMinutes** parameter defines the minimum time in minutes after which a suspended user is automatically reactivated. Users are suspended if they exceed the maximum number of login violations. After the specified number of minutes has passed, users can log onto the PVWA again successfully. The default value is -1, which indicates that users must be reactivated manually.

Viewing Generated Passwords

The following parameter, in the **General** parameters, determines which users are able to generate passwords and view them.

- The **AllowViewingGeneratedPassword** parameter determines whether or not users who do not have the 'Retrieve' permission for accounts in a Safe can view passwords immediately after generating them, or specify a new password manually. If this parameter is set to 'No', these users will not be able to generate or specify new passwords. The default value is **No**.

Enabling Users to Manage Account Properties

The following parameter, in the **General** parameters, determines which users are able to add new values to lists of account properties.

- The **EnableAddingNewValueToListProperty** parameter determines whether or not users with relevant permissions will be able to add new values to lists of valid property values. The default value is **Yes**.

Enabling Access to the CyberArk Viewfinity Console

The following parameter, in the **General** parameters, enables you to access the CyberArk Viewfinity Console where you can configure the Privileged Endpoint Protection policy for your organization. When CyberArk Viewfinity is configured for Windows authentication, you can access the CyberArk Viewfinity Console with single sign on.

- The **ViewfinityURL** parameter defines the URL where you can access the Privileged Endpoint Protection policy and configure it. If this parameter value is not specified, the Privileged Endpoint Protection Policy link in the System Configuration page is not displayed. Specify the URL of the CyberArk Viewfinity Console using the following structure: <Viewfinity URL>/vfmain.aspx. for example, <http://viewfinity.mycompany.com/vfmain.aspx>.

Displaying Sign in Information

The following parameters, in the **General Settings** of the **Authentication Methods** parameters determine the type of sign in information that will be displayed for each user who accesses the system. This information takes into account all sign ins and attempted sign ins from any network where users can access the PWWA.

- The **DisplayLoginFailureInfo** parameter determines whether or not to display information about failed logins since the last successful login. This is only relevant for CyberArk authentication. The default value is **No**.
- The **DisplaySuccessfulLoginInfo** parameter determines whether or not to display information about the last successful login. This is only relevant for CyberArk authentication. The default value is **No**.
- The **AutomaticallyDisplayLoginInformation** parameter determines whether or not to automatically display the login information determined by the **DisplayLoginFailureInfo** and **DisplaySuccessfulLoginInfo** parameters after login. The default value is **No**.

Configuring the Accounts List

The following parameters, in the **Accounts UI Preferences** parameters of the Web Access Options, define the configuration for the Accounts page.

Displaying Predefined Views

The following parameters determine how the predefined account views are displayed.

- The **PageSize** parameter defines the maximum number of accounts to display in the grid in each page. The default value is **25**.
- The **DisplayDeletedItems** parameter determines whether or not deleted accounts will be displayed in the grid. The default value is **Yes**.
- The **DefaultView** parameter defines the default view that is shown when the Accounts List is displayed. Each user can configure his own preferences. The default value is **Recently**.
- The **MaxDisplayedRecords** parameter defines the maximum number of accounts or files that may be displayed in a view. The default value is **20000**.
- The **MaxPageSize** parameter defines the maximum number of records that may be displayed in a single page. The default value is **100**.

Operational View

- The **GridOperationalViewGroup** parameter in the **Operational Views** parameters, defines the groups that users must belong to so that they are authorized to view the Operational Views. The default value is **PVWAMonitor**.

Requests View

- The **Visible** parameter in the **Requests Views** parameters, determines whether or not the Requests Views are displayed. The default value is **Yes**.

Account View

- The **DisplayDeletedAccountsView** parameter in the **Account Views** parameters, determines whether or not the Deleted Accounts View is displayed. The default value is **Yes**.

Configuring Toolbar Actions

The following parameters, in **Toolbar Actions**, define the actions that can be initiated from the toolbar in the Accounts page. Toolbar actions are visible according to user permissions.

- Each set of **Action** parameters defines a single toolbar button. This set includes the following parameters:
 - The **Name** parameter defines the unique name of an action that will be available in the Accounts list.
 - The **Visible** parameter determines whether or not the toolbar action will be visible, by default.

Viewing Accounts and Service Accounts

The following parameters, in **Accounts** and **Usages**, define how the Accounts and service account lists are displayed.

- Each set of **DisplayedColumns** parameters defines the columns displayed in the list. In addition to being determined at system level, each user can customize their own accounts lists.
 - The **SortBy** parameter determines the default column by which to sort the grid.

Each set of **Column** parameters defines a single column to display in the list. This set includes the following parameters:

- The **Name** parameter specifies the name of the account property that is displayed in this column.
- The **DisplayName** parameter specifies the displayed name of the column header. If this is not specified, the default column name is displayed.
- The **Width** parameter defines the width of the column. This is specified in pixels.
- The **DataType** parameter specifies the type of information that will be displayed in the column.
- The **Visible** parameter determines whether or not the column will be visible, by default.

Each set of **Grid Actions** parameters defines the actions that will be available from the objects list.

Each set of **Action** parameters defines an action that will be available from the objects list. This set includes the following parameters:

- The **Name** parameter specifies the unique name of an action that will be available from the accounts list.
- The **DisplayInGrid** parameter determines whether or not the action will be displayed in the reports list.

The **DisplayInActionMenu** parameter determines whether or not the action will be displayed in the action menu.

Accounts Feed

Configuring the Accounts Feed

The following parameters in the CACPMScanner.exe.config file enable you to determine how the CPM Scanner works. This file is stored in the Scanner subfolder of the Password Manager installation folder.

Notes:

- All parameters that are not described in this guide must not be changed.
- For more information about CPM Scanner logs, refer to *Configuring Logs*, page 503.

Configuration and Management

The following parameters define the connection to the Vault.

- The **VaultFile** parameter defines the full path name of the Vault configuration file of the Vault where the discovered accounts will be onboarded. Specify a full path name. The default value is **C:\Program Files (x86)\CyberArk\Password Manager\Vault\Vault.ini**.
- The **ConfigurationCredentialFile** parameter defines the full path name of the user credentials file that will be used to connect to the Vault. Specify a full path name. The default value is **C:\Program Files (x86)\CyberArk\Password Manager\Vault\User.ini**.

Configuring Discovery Filters

The following parameters determine the CPM Scanner filters.

- The **AddFilterAccountsInactiveDays** parameter defines the number of days that a computer in the Active Directory is inactive and is therefore excluded from the scan. Specify a positive number or -1 to disable filtering. The default value is -1.
- The **AccountTypeScanFilter** parameter defines the type of accounts that will be scanned. Specify one of the following:
 - **DomainAccounts** – Only domain users.
 - **Local Accounts** – Only local users.
 - **All** – Domain and local users. This is the default value.
- The **GroupTypeScanFilter** parameter defines the local groups that will be scanned for accounts. Specify one of the following:
 - **PrivilegedGroups** – Administrators and Power Users.
 - **NonPrivilegedGroups** – All other local groups.
 - **All** – Both privileged and non-privileged local groups. This is the default value.
- The **ScanScheduledTasks** parameter determines whether or not Scheduled Tasks will be scanned. Specify Yes or No. The default value is **Yes**.
- The **ScanWindowServices** parameter determines whether or not Windows Services will be scanned. Specify Yes or No. The default value is **Yes**.

Optimizing the CPM Scanner

The following parameters optimize the CPM Scanner.

- The **UseLDAPs** parameter defines whether or not to use LDAPs when connecting to the Active Directory. Specify Yes or No. The default value is **No**.
- The **BundleTransaction** parameter determines whether or not communication with the Vault will be bundled. Do not change this parameter. Specify Yes or No. The default is **Yes**.
- The **MaxThreadNumber** parameter determines how many machines will be scanned simultaneously during each discovery process. It is recommended to specify a number between 10 and 20. The default value is **10**.

Configuring Logs

The following parameters determine log settings for the CPM Scanner.

- The **LogFolder** parameter defines the path where the CACPMScanner log is stored. Specify a full path name. The default value is **C:\Program Files (x86)\CyberArk\Password Manager\Logs**.
- The **ConsoleLogActive** parameter defines whether or not a console log will be created when the CPM Scanner is run. Specify Yes or No. The default value is **Yes**.
- The **TraceLogActive** parameter defines whether or not a trace log will be created each time a discovery process is run. Specify Yes or No. The default value is **Yes**.
- The **TraceLogPath** parameter defines the path where the trace log is created. Specify a folder under the Scanner subfolder of the Password Manager installation folder. The default value is **\log**.
- The **ConsoleLogPath** parameter defines the path where the console log is created. Specify a folder under the Scanner subfolder of the Password Manager installation folder. The default value is **\log**.

Managing the Accounts Feed

The CyberArk Central Policy Manager Scanner service scans machines and discovers privileged accounts and their dependencies. A scanner is installed with each CPM so that you can scan all distributed networks in your organization. For more information about managing the CyberArk Central Policy Manager Scanner service, refer to *Administering the CyberArk Central Policy Manager Scanner*, page 504.

You can manage the discovery processes, view the results and onboard accounts in the PVWA.

Enabling and Disabling the Accounts Feed in the PVWA

The **ShowAccountsDiscovery** parameter, in the PVWA General Options, determines whether or not a link to the Accounts Discovery page is displayed in the Accounts page. By default, this parameter is enabled.

Administrating the CyberArk Central Policy Manager Scanner

The CyberArk Central Policy Manager Scanner service is installed on the CPM machine automatically during CPM installation.

To Stop the CyberArk Central Policy Manager Scanner service

1. From the **Start** menu, select **Settings**, then **Control Panel**.
2. From the list of Control Panel options, select **Administrative Tools**, then **Services**; the Services window appears.
3. Stop the **CyberArk Central Policy Manager Scanner** service.

When you are not working with the Accounts Feed you can disable the scanning functionality to reduce the workload on the Vault in complex environments.

To Start the CyberArk Central Policy Manager Scanner service

1. From the **Start** menu, select **Settings**, then **Control Panel**.
2. From the list of Control Panel options, select **Administrative Tools**, then **Services**; the Services window appears.
3. Start the **CyberArk Central Policy Manager Scanner** service.

CyberArk Central Policy Manager Scanner Logs

All activities that are carried out by the CyberArk Central Policy Manager Scanner service are written in log files and stored in subfolders of the Password Manager installation folder.

The following log files contain the activities of the CyberArk Central Manager Scanner:

- **CACPMScanner.log** - This file contains informational messages and errors that refer to CPM Scanner function. This log is meant for the system administrator who needs to monitor the status of the CPM Scanner. This log file is stored in the Logs subfolder of the Password Manager installation folder.
- **DNAConsole.log** – This file indicates when the discovery process began and information about any general errors that occurred. This log file is stored in the Scanner\Log subfolder of the Password Manager installation folder.
- **DNATrace-<timestamp>-PM.log** – This file contains detailed information about each scan. The timestamp represents the date and time when the discovery process started. This log file is stored in the Scanner\Log subfolder of the Password Manager installation folder.

User Preferences

The following parameters, in the **Users** parameters of the Web Access Options, define PVWA users, groups and Safes.

Users who are owners of the Safe where User Preferences are stored can update and save their preferences in the Password Vault Web Access. These preferences determine which information and how much will be displayed when the user logs on.

- The **UserPreferencesSafe** parameter specifies the Safe where the user's personal preferences in the Password Vault Web Access interface are stored. By default, the user preferences are stored in the PVUserPrefs Safe and users who wish to change their preferences should be members of the PVWAUsers group in that Safe.
- The **ConfigurationSafe** parameter specifies the name of the Safe where the Password Vault Web Access configuration files are stored. These are the PVConfiguration.xml, Policies.xml, and SafeTemplate.xml files which are all configured in the System Configuration page.

Adding Safe Members from LDAP Directories

The following parameters, in the **LDAP Search** parameters of the Web Access Options, define searches on the LDAP directory.

The PVWA can search the LDAP directories for users who will be added as Password Vault users, according to the profile set for each directory. The following parameters enable you to specify search criteria so that the search is focused and therefore optimized, resulting in quick and accurate results.

- The **SearchType** parameter specifies the type of search to perform. The values are described in the table below:

Value	Description
BeginsWith	The search will identify the specified search criteria if it appears at the beginning of the specified search field. This is the default value.
EndsWith	The search will identify the specified search criteria if it appears at the end of the specified search field.
Contains	The search will identify the specified search criteria if it appears anywhere in the specified search field.
ExactMatch	The search will identify the specified search criteria if it is identical to the specified search field.

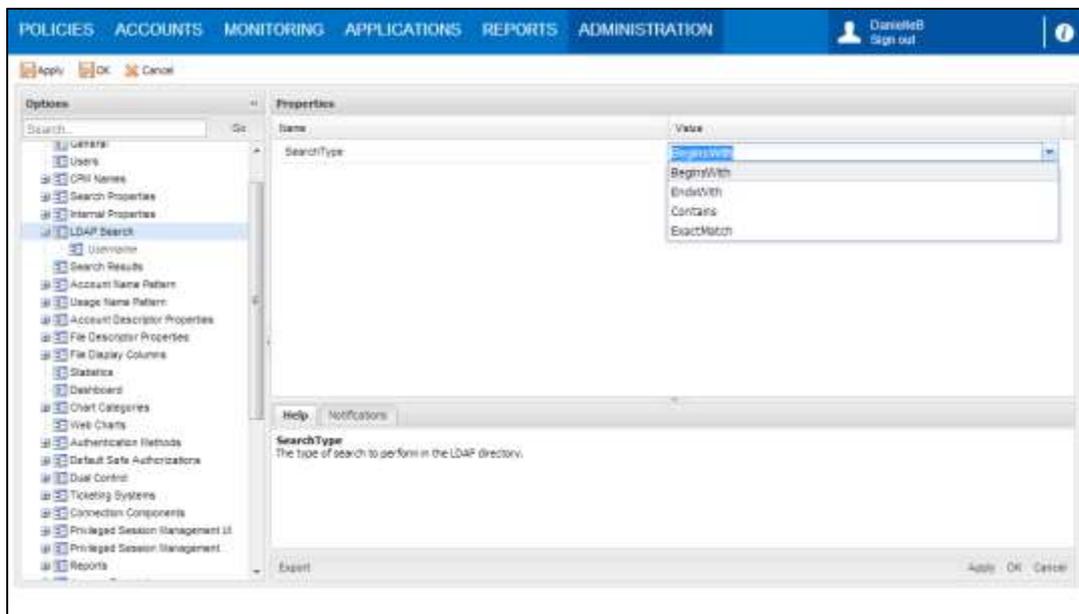
Note: The above values are not case-sensitive.

- The **SearchField** parameter specifies the field in the profile that is set for the LDAP directory that will be searched for the specified text. Some possible values are described in the table below:

Value	Description
Username	The search will be based on the Username field. This is the default value.
FirstName	The search will be based on the users' FirstName field.
LastName	The search will be based on the users' LastName field.
Email	The search will be based on the users' email field.

Note: The above values are not case-sensitive.

In the following example, the search for LDAP users will look for users whose names begin with the characters that are specified in the PWWA. The search field under LDAP Search determines that the search will be carried out based on the fields that correspond to the 'Username' parameter in the profile that has been set for the LDAP directory.



Searching for Accounts and Files

The following parameters define searches in the Password Vault Web Access, as well as how the search results will be displayed.

When a search is carried out for an account or file, the Password Vault Web Access searches according to specified properties in the Vault.

- The **Search Properties** parameters specify the keywords and locations that determine how the search for accounts or files will be carried out in the Vault.
- The **AutoOpenObjectDetailsInSearch** parameter in the **General** parameters determines whether the Files Details page for a file will be displayed if a search results in that single file. This parameter is only relevant for files.

The following parameters in the **Search Results** parameters define the information that will be displayed in the Search Results page.

- The **ExecutionMaxDuration** parameter specifies the maximum number of seconds that a search process will last. By default, the value of this parameter uses the execution timeout value of the application.
- The **MaxDisplayedRecords** parameter specifies the maximum number of accounts or files that will be displayed in the search results pages when the user clicks **Show All**. The default value is **500** accounts.
- The **MaxDisplayedUsagesRecords** parameter defines the maximum number of service accounts that will be displayed in the Usage tabs in the Account Details page. The default value is **100** service accounts.
- The **WideAccountsSearch** parameter determines whether or not the PVWA will search for accounts using the legacy method of searching in Safe, folder and account name, as well as in property names and properties, or will search for accounts based on Safe name and account property values only. The new search method is substantially faster, especially in large environments, but is only relevant if using account properties. This parameter only affects the search when the user supplies a pattern and is ignored if a pattern is not specified.
- The **DisplaySearchAllAccountsWarning** parameter determines whether or not a warning is displayed to end users before they start a search in all the accounts they have access to. This is recommended in large implementations where users have access to a large number of accounts, and a search operation in all of these accounts may take a while. The default value is **Yes**.

Identifying Accounts and Files

Accounts and files can be identified by their properties, as well as by their name, enabling you to recognize them easily. The following parameters enable you to determine the format of the account or file identifier, as well as the name that is given automatically to an account. If none of the properties that are used in the identifier are specified in the account, the account name will be displayed.

Specifying an Account Name Pattern

When an account is created through the Password Vault Web Access, you can either set a name or choose to create it automatically based on **Account Name Pattern** parameters. These parameters specify the account properties that will comprise the account name and the order in which they will appear.

- The **AutoGenerate** parameter determines whether the default setting will be to generate the account name automatically or whether the user will specify a custom account name.
- The **Separator** parameter determines the character that will be used to separate the different parts of the account name.

Specifying a Service Account Name Pattern

The name of a service account can also be generated automatically according to **Usage Name Pattern** parameters. These parameters specify the account properties that will comprise the service account's name and the order in which they will appear.

- The **Separator** parameter determines the character that will be used to separate the different parts of the account name.

Identifying Accounts and Files

Specified properties can be used to identify accounts and files. The order in which the properties are specified is the order in which they will appear in the identifier.

- The properties that are used to identify accounts are specified in the **Account Descriptor Properties**.
- The properties that are used to identify files are specified in the **File Descriptor Properties**.

Configuring Statistics

The following parameters, in the **Statistics** section of the Web Access Options, define the scope of statistical information about files that is displayed to users.

The Frequently and Recently panes display the files that the user has accessed. The files that appear in these panes are displayed according to predefined parameters that determine the number of days and the number of files to include. The contents of these lists is determined by files that have been added, retrieved or changed at some point during the specified number of days.

The Frequently Pane

- The **FrequentlyDays** parameter determines how many days will be included in the list.
- The **FrequentlyCount** parameter determines the number of files that will appear in this list.

The Recently Pane

- The **RecentlyDays** parameter determines how many days will be included in the list.
- The **RecentlyCount** parameter determines the number of files that will appear in this list.

The following general parameters define the records that are displayed in both the Frequently and Recently lists.

- The **MaxRecords** parameter specifies the maximum number of records that will be displayed in the Frequently or the Recently list. The default value is **1000** records.
- The **ShowDeletedAccountsInFrequentlyRecently** parameter determines whether or not deleted files will be displayed in the Frequently and Recently lists. The default value is **Yes**.

Accounts Check-out and Check-in

Auditing and control requirements demand full identification and monitoring of users who access privileged accounts during any given period. In addition, to guarantee accountability, each user who accesses a privileged account must be the only one to do so.

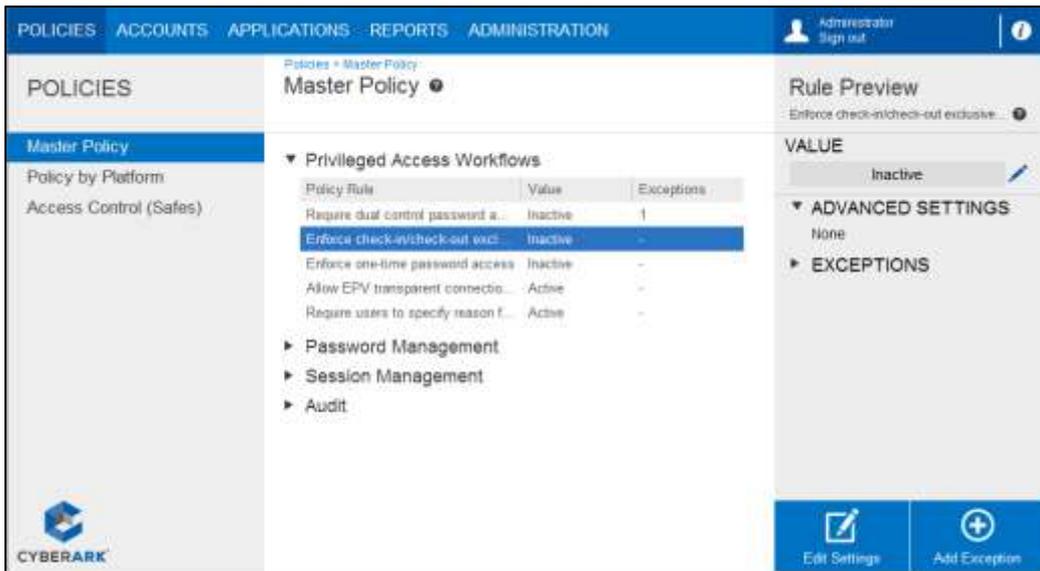
If a CPM is installed, accounts that are not released immediately by the user can be released automatically after a predetermined period of time. Alternatively, they can be released manually, which prompts an immediate password change.

Passwords that are not managed by the CPM must be released manually. A user who is not authorized to release an exclusive account can initiate a process that sends a notification to an authorized user, who will then release the account and change the password. For more information about configuring this process, refer to *Releasing Exclusive Accounts*, page 247.

Exclusive account check-in and check-out is configured at system level in the Master Policy, and applies to individual accounts as well as to account groups. For more information, refer to *Account Groups*, page 215.

To Define Exclusive Accounts in the Master Policy

1. Click **POLICIES** to display the Master Policy.
2. In **Privileged Access Workflows**, select **Enforce check-in/check-out exclusive access**.
3. In the **Rule Preview** pane, click the **Edit** icon; the following options appear:
 - **Active** – This rule will be applied at Master Policy level to all platforms, unless an Exception overrides it.
 - **Inactive** – The rule will not be applied at all.



4. Select **Active**, then click the **Save** icon to save the new rule status.
5. If a CPM has been assigned to the Safe where the accounts are stored, make sure that the CPM user, by default PasswordManager, has the following authorizations:
 - Unlock accounts
 - Manage Safe
6. Click **ADMINISTRATION** to display the **System Configuration** page, then click **Platform Management** to display a list of supported target account platforms.
7. Select the platform to configure, then click **Edit**; the configuration page for the selected platform appears.
8. In **Automatic Password Management**, set the following parameters:
 - In **Privileged Account Management**:
 - **MinValidityPeriod** – This parameter determines the number of minutes after which an exclusive account associated with this platform will be released automatically by the CPM.
 - **ResetOverridesMinValidity** – This parameter enables the user to immediately release a locked account manually through the Password Vault Web Access, overriding the 'MinValidityPeriod' parameter. Set this parameter to Yes.

- In Password Change:
 - **AllowManualChange** – This parameter ensures that exclusive accounts will be changed after they are manually checked-in by end users. Set this parameter to Yes.

All the accounts stored in the configured Safe that are managed by these platforms will be treated as exclusive accounts and will be changed automatically whenever they are returned to the Safe.

To Define Notifications about Releasing Accounts that are Managed Manually

When the Master Policy's 'Enforce check-in/check-out exclusive access' rule is active, you can create a platform to initiate a notification process when an account that is not managed by the CPM needs to be released. This sends a corresponding ENE notification to predefined users who change the account and release it.

1. Create a platform for unmanaged accounts:
 - i. Click **ADMINISTRATION** to display the **System Configuration** page, then click Platform Management to display a list of supported target account platforms.
 - ii. Select an existing platform that is similar to the new target account platform, then click **Duplicate**; the Duplicate Platform window appears.
 - iii. Type the name and a description of the new platform, then click **Save & Close** to create the new platform.
 - iv. Select the new target account platform, and then click **Edit**; the configuration page for the selected platform appears.
 - v. Set the following properties:

Properties	Setting
In the Password Change parameters:	
▪ AllowManualChange	No
▪ PerformPeriodicChange	No
▪ OneTimePassword	No
In the Password Verification parameters:	
▪ VFAllowManualVerification	No
▪ VFPerformPeriodicVerification	No
In the Password Reconciliation parameters	
▪ RCAAllowManualReconciliation	No
▪ RCAAutomaticReconcileWhenUnsynced	No
In the Notification parameters:	
▪ NFNotifyOnUnreleasedPasswords	Yes
In the UI & Workflows parameters:	
▪ ForceManualPasswordChange	Yes

- vi. Change any additional parameter values and/or add new values to define the new platform.
- vii. Click **Apply** to save the new configurations and apply them immediately.

2. Create a notification template:
 - i. In the System Configuration page, click **Notification Settings**; the Notification Settings page appears.
 - ii. Expand **NotificationAgentRules**; the list of configured notification rules is displayed.
 - iii. Right-click the **Account must be changed and released** then, from the pop-up menu, select **Copy**.
 - iv. Right-click **NotificationAgentRules** then, from the pop-up menu, select **Paste**; the copied rule is added to the list of configured rules.
 - v. Select the new rule and change the following properties:
 - **ID** – Set to **211**.
 - **Client ID** – Set to **PVWA**.
 - vi. Click **Apply** to save the modified recipients list and stay in the Notification Settings page.
3. By default, this notification is sent to the Vault Admins group and the user who has locked the account. You can change these recipients so that all users who are authorized to update passwords will receive this notification.
 - i. Expand **EventNotificationEngineRecipients** and right-click **EPVUnrelease** then, from the pop-up menu, select **Delete**.
 - ii. Right-click **EPVUnrelease** again and, from the pop-up menu, select **Add RecipientObject**; a new recipient object is created.
 - iii. Specify the following properties:
 - **Type** – Specify **VaultQuery**.
 - **QueryType** – Specify **Owners**.
 - **Filter** – Specify **PermissionBased**.
 - **Value** – Specify **Update**.
 - iv. Click **OK** to save the recipients list and return to the System Configuration page.

Dual Control

The **Dual Control** parameters enable you to benefit from the Vault's dual control mechanism. This means that whenever a user tries to retrieve an account, a request is created and confirmation must be received from authorized users.

Dual control is configured at system level in the Master Policy. For more information, refer to *Dual Control*, page 251.

- Click **ADMINISTRATION** to display the **System Configuration** page, then click **Options** and display the Dual Control parameters.

Creating Requests

- The **FromTime** parameter states the default value for the 'From time' in the request when a time frame is specified.
- The **ToTime** parameter states the default value for the 'To time' in the request when a time frame is specified.
- The **Timeframe** parameter specifies the number of days after the 'From Date' specified in the request that the 'To Date' will display.
- The **ForceTimeframe** parameter determines whether or not the user is required to specify a time frame when he creates a request.
- The **MaximumTimeframe** parameter specifies the maximum number of days that can be specified in the request.
- The **MultipleAccessChecked** parameter determines whether or not multiple access will be a default setting.
- The **ForceMultipleAccess** parameter determines whether or not the request must be for multiple access to the account or file during the specified time frame.
- The **AllowTimeframe** parameter determines whether or not the access timeframe and multiple access features will be displayed when a new request is created. If this parameter is set to 'No', the access timeframe and multiple access features will be hidden.
- The **AllowMultipleAccess** parameter determines whether or not the multiple access features will be displayed when a new request is created. If this parameter is set to 'No', the multiple access features will be hidden.

Viewing Requests

- The **AllowViewingHandledRequests** parameter determines whether users who are authorized to confirm requests will be able to see requests they have already handled.

Confirming/Rejecting Requests

- The **ForceConfirmationReason** parameter determines whether or not an authorized user is required to specify a reason when confirming or rejecting a request.

Configuring Confirmation by Direct Managers

To enforce the advanced Only direct manager can approve password access requests setting, the Vault must be configured to recognize LDAP directories so that it can identify the direct managers of users who create requests.

Notes:

- Direct managers must belong to a group that is defined as a direct manager group. A confirmer who is a direct owner of a Safe cannot confirm requests as a direct manager.
- A direct manager who is a member of multiple groups which are owners of the same Safe, cannot confirm requests as a direct manager either.

For more information about integrating with LDAP directories, refer to the Privileged Account Security Installation Guide for details.

To Configure Confirmation by Direct Managers

In the PVWA:

1. Log onto the PVWA as an administrator user. Make sure that this user belongs to the **Vault Admins** group.
2. Add the LDAP user/group of managers who will confirm requests as a Safe owner of the Safe that contains the privileged accounts for which users require confirmation before accessing them.
 - i. In the Safes list, select the relevant Safe, then click **Members**; the Safe Details page appears.
 - ii. In the Members tab, click **Add Member**; the Add Safe Member window appears.
 - iii. In the **Search** edit box, enter either part of the name of the LDAP user/group to add as a Safe member or the whole name. You can also leave the Search edit box empty to search for all groups.
 - iv. In the **Search In** drop-down box, select **LDAP**, then click **Search**; a list of users and groups in the external directory whose names match the specified keyword is displayed.
 - v. Select the LDAP group to add, then select the permissions that the LDAP group will have in the Safe. Specifically, select **Authorize password requests**.

Add Safe Member

Search: Search In:

Selected Search: AD Display 3 result(s)

Name	Business Email	Full Name
admin2	admin2@domqa.com	admin2
Administrator	Administrator@do...	
Administrators		

View Audit log
 View Safe Members
 Workflow
 Authorize password requests
 Level 1
 Level 2
 Access Safe without confirmation
 Advanced

- vi. Click **Add**; the LDAP group is added as a Safe owner with the selected permissions and confirmation appears at the bottom of the screen.
 - vii. Click **Close**; the Safe Details page appears and displays the new Safe member in the Members list.
3. Configure LDAP integration to determine the LDAP Managers' group that will confirm requests.
 - i. Click **ADMINISTRATION** to display the **System Configuration** page, then click **LDAP Integration**; the LDAP Integration page appears.
 - ii. Expand **Profiles**, then select the LDAP profile to configure; the profile properties are displayed in the Properties pane.
 - iii. Specify the following property to determine the Managers' group that will confirm requests:
 - The **ManagersGroupDN** specifies the name of an LDAP attribute in the LDAP that defines a specific user/group of managers who will confirm direct manager requests for a specific user/group. These groups are created for specific users/groups according to the LDAP hierarchy. For example, if you specify **ManagersGroupDN=DirectManagersGroup**, the Vault will search LDAP users for the DirectManagersGroup attribute to identify users and groups who will be able to confirm requests at managerial level.
 4. Click **Apply** to save the new configurations and apply them immediately,

or,

Click **Save** to save the new configurations and apply them after the period of time specified in the **RefreshPeriod** parameter.

Integrating with Ticketing Systems

The Privileged Account Security solution can integrate with ticketing systems to validate open tickets or create new tickets and release accounts when a user requests to access a privileged or shared account. In order to perform the online validation, an integration module should be developed. For more information, refer to *Developing an External Module for Ticketing Integration*, page 519.

You can also require end users to specify a Ticketing System name and a ticket ID, without the online validation against the ticketing system.

Requirements

- An active ticketing system.

Before Configuration

The PWWA uses an account stored in the Vault to log onto the ticketing system.

- From the PWWA, create an account that will be used to connect to the ticketing system in the **PWWATicketingSystem** Safe.

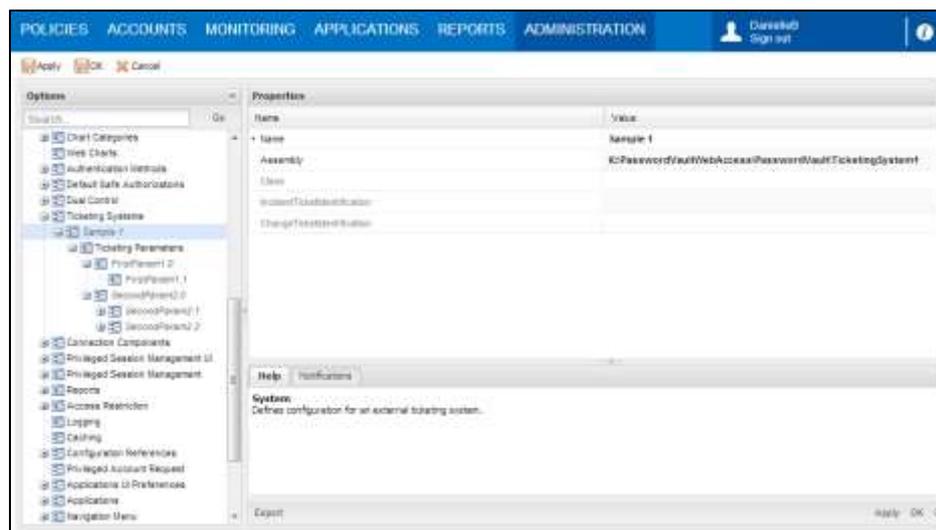
To Configure Integration with Ticketing Systems

1. Define the ticketing system to integrate:
 - Click **ADMINISTRATION** to display the **System Configuration** page, then click **Options**, and display the **Ticketing Systems** parameters:
 - Specify the ticketing system that can be specified during requests:
 - i. Right-click on **Ticketing Systems**, then select **Add System**; a page is added for a new ticketing system and its main parameters are displayed.
 - ii. Specify the parameters for the system:
 - Name** – The name of the ticketing system.
 - Assembly** – An assembly name or the full pathname of the assembly file where the validator module exists. This parameter can also specify a custom module that you have developed yourself. For more information, refer to *Developing an External Module for Ticketing Integration*, page 519.
 - Class** – A full name (including namespace) of the validator class. Leave this parameter blank to use the first validator class that is found.
- You can specify more than one ticketing system by adding as many ticketing systems as you need.
- Specify the account in the Vault that will be used to connect to the ticketing system:
 - i. Right-click on the system page, then select **Add ConnectionDetails**; a page is added for you to specify the connection account details.

- ii. Specify the connection details for the system:
 - **Safe** – The Safe where the account used to connect to the ticketing system is stored.
 - **Folder** – The folder in the Safe where the account used to connect the ticketing system is stored.
 - **File** – The name of the account that is used to connect to the ticketing system.
 - The connection account is retrieved from the Vault by the internal PVWA application user, PVWAApUser (by default). The PVWA installation automatically creates a Safe ('PVWATicketingSystem') that can be accessed by this user. If you store the connection account in a different Safe, the PVWA application user must have the 'Retrieve passwords' and 'List passwords' permissions on the account or on the entire Safe.
- Specify any additional parameters that will be passed to the ticketing system integration module:
 - i. Right-click on the Ticketing System, then select **Add TicketingParameters**; a new ticketing parameters section is added.
 - ii. Right-click on **Ticketing Parameters**, then select **Add Parameter**; a new parameter is added.
 - iii. Specify the name and value of the new ticketing parameter.

You can specify as many parameters as the ticketing system integration module requires. The parameters can be specified in different hierarchies and levels and are transferred to the external ticketing module in an xml object (XmlNode type).

The following example shows how dynamic parameters can be defined for each ticketing system and automatically passed to the integration module in an xml format:



The following XML would be passed to the integration module:

```
<TicketingParameters>
  <Parameter Value="FirstParamValue1.0" Name="FirstParam1.0">
    <Parameter Name="FirstParam1.1" Value="FirstParam1.1value" />
  </Parameter>
  <Parameter Value="SecondParam2.0Value" Name="SecondParam2.0">
    <Parameter Name="SecondParam2.1" Value="SecondParam2.1Value" />
    <Parameter Name="SecondParam2.2" Value="SecondParam2.2Value" />
  </Parameter>
</TicketingParameters>
```

2. Enable the ticketing system integration for the relevant platforms:
 - i. Click **ADMINISTRATION** to display the **System Configuration** page, then click **Platform Management** to display a list of supported target account platforms.
 - ii. Select the platform to configure, then click **Edit**; the configuration page for the selected platform appears.
 - iii. Expand **UI & Workflows**, and display the **TicketingSystem** parameters.
 - iv. Specify the following parameters:
 - **EnterTicketingInfo** – Whether or not users will be required to enter ticketing system information before being able to retrieve an account.
 - **ValidateTicketNumber** – Whether or not to perform an online validation or creation of a ticket ID against the ticketing system.
 - v. You can enable only some of the active ticketing systems for a platform. If you do not specify the list of active ticketing systems, all the systems defined at system level will be available when accessing accounts associated to this platform. However, if you define specific ticketing systems at platform level, only those ticketing systems will be displayed in the lists of ticketing systems that can be selected by users in the Show Password window when they access privileged accounts associated to the platform.
 - a. In the selected platform, expand **Ticketing Systems**, then expand **ActiveTicketingSystems**.
 - b. Select **TicketingSystem** then, in the Name property, specify the name of the active ticketing system that is configured at system level. Specify the name of the ticketing system in exactly the same way as it is defined at system level. This ticketing system will be displayed in the list of ticketing systems for accounts associated to this platform.
3. Click **Apply** to save the new ticketing configuration and stay in the Options configuration page,
or,
Click **OK** to save the new configuration and return to the System Configuration page.

Developing an External Module for Ticketing Integration

The Privileged Account Security solution provides the infrastructure to integrate with any ticketing system and either validate open tickets or create them. An external .Net module enables you to implement this feature.

During installation, a sample module called **MyTicketingValidator** is copied to the installation folder, by default **C:\inetpub\wwwroot\PasswordVault**. This file can be used as a template for your own tailored implementation.

The following instructions explain how you can develop your own external ticketing integration module. For more information, contact CyberArk support.

To Create an External Module for Ticketing Information

1. Create a new Visual Studio project for the external module.
2. From the new project, add a reference to **CyberArk.PasswordVault.PublicInterfaces.dll**. This DLL can be found in the PVWA 'bin' folder.
3. Copy and add the external module sample to the project and rename it appropriately.
4. Open the renamed external module, and specify the parameters that you require. In the sample file, all the information that needs to be replaced is marked with 'TODO'.
 - i. Change the name of the namespace. For example, specify the following: `MyCompany.TicketingValidation`.
 - ii. Change the name of the class. For example, specify the following: `ExternalTicketValidation`.
 - iii. Create the code that will verify or create tickets:
 - Implement the `ValidateTicket` (`IValidationParametersEx` parameters, out `ITicketOutput` ticketingOutput) method of the `ITicketValidatorEx` interface.
 - The parameters listed in the following table are sent to this function by the PVWA through `IValidationParametersEx` parameters.

Note: Existing integration modules, created in previous versions, which implement the `ITicketVaildatorEx` or `ITicketVaildator` interfaces are still supported in this version.

Parameter	Specifies
TicketId	The ticket ID entered by the user
SafeName	The Safe where the requested account is stored.
FolderName	The folder where the requested account is stored.
ObjectName	The name of the requested account.
MachineAddress	The address of the machine specified on the account, if it exists.
TransparentMachine Address	The address of the machine used in a transparent connection, if it exists.
UserName	The name of the user specified on the requested account.
PolicyId	The name of the platform specified on the requested account.

Parameter	Specifies
AdditionalProperties	All other properties of the account requested by the user (Dictionary type).
RequestingUser	The name of the user requesting the account.
RequestingUser FirstName	The first name of the user requesting the account.
RequestingUser Surname	The surname of the user requesting the account.
BusinessEmail	The email address of the user requesting the account.
ProvidedReason	The reason provided by the user for requesting the account.
XMLNodeParameters	Additional parameters defined in the ticketing system configuration, as <code>XmlNode</code> type.
TicketingConnection Account	The details of the account used to connect to the ticketing system, as <code>ITicketingConnectionAccount</code> type.
SystemName	The name of the ticketing system selected by the user.
DualControl	Indicates whether dual control is enforced in the current password request.
DualControlRequest Confirmed	Indicates whether a dual control request was already confirmed or not, so that the ticketing module can decide which changes to implement in its logic. If the dual control request was not yet confirmed, the value will be false, otherwise it will be true. This property is relevant only when the DualControl property is set to Yes .

- The following table lists the `ITicketingConnectionAccount` interface:

Parameter	Specifies
Address	The Address property of the connection account
UserName	The Username property of the connection account
Password	The password used to connect to the ticketing system (string type)
Safe	The Safe where the connection account resides
Folder	The Safe where the connection account resides
ObjectName	The name of the connection account object in the Vault
Properties	The full list of properties of the connection account (Dictionary<string, string> type)

- Specify the parameters or messages that will be returned to the PVWA after a validation or creation process through `ITicketOutput` `ticketingOutput`:

Parameter	Specifies
UserMessage	An error message that will be returned if the ticket was not approved. If the ticket validation fails (returns false), this message will be displayed to the user requesting the password (string type).
TicketAuditOutput	The text that will be written in the audit log record of the Retrieve Password operation. This text may include any information on the ticket, including the ticket ID, its current status, etc. The text will be included in the reason field of the audit record as follows: <code>"(Ticketing System=sys name)(Ticket ID=ticket id)(Ticketing Audit=audit output) reason"</code>
TicketId	The validated ticket ID. This ticket ID will be written to the audit log.
RequestStartDate	The ticket start date from the ticketing system (DateTime type).
RequestEndDate	The ticket end date from the ticketing system (DateTime type).
AdditionalOutput	Additional output returned to the PVWA. For future use.

- The function should return a confirmation that the ticket was validated or created successfully, as a Boolean return value. If a `True` value is returned, the requested password will be displayed to the user or used for transparent connection, based on the user operation. If a `False` value is returned, the `ITicketOutput.UserMessage` will be displayed to the user and the password will not be displayed or used.
- Compile the external module DLL you created in the previous steps and copy it to the PVWA 'bin' folder. This action will cause the PVWA web application to restart.
 - Integrate the external module with PVWA by adding a new Ticketing System configuration to PVWA:
 - In the PVWA, click **ADMINISTRATION** to display the System Configuration page, then click **Options**.
 - In the **Ticketing Systems** section, add a new **Ticketing System** section.
 - In the **Name** parameter, specify the display name of the new ticketing system.
 - In the Assembly parameter, specify the name of the extension module you created.
 - If the DLL is in the PVWA's bin folder (c:\inetpub\wwwroot\PasswordVault\Bin), specify the module filename without the path and without the .dll suffix.
 - If the DLL is placed in any other folder, specify its full path, including the suffix.
 - Optionally, in the **Class** parameter, specify the validator class name. This is the class that implements the `ITicketValidatorEx` interface.

- vi. In the **Connection Details** parameter, specify the location of the account in the Vault that is used to connect to the ticketing system. This is made available to the external module through the `IVValidationParametersEx.TicketingConnectionAccount` property.
 - vii. In the **Ticketing Parameters** parameter, specify the custom parameters to send to the ticketing system. This is made available to the external module in XML format through the `IVValidationParametersEx.XMLNodeParameters` property.
7. The following additional considerations can also be included when you develop an integration module:
- The validation of the ticket can include validations of the status of the ticket, the ticket timeframe, etc.
 - If the ticketing system is not available, consider whether the requested password should be returned to the end user or not. If you decide to return a success value in this case, it is recommended to include this information in the returned audit output.

The next time a user accesses a password that requires a ticket to be validated or created before the password can be retrieved, PWVA will call the integration module to validate or create the request in the specified ticketing system.

IDebugLog

The `IDebugLog` enables you to add debug logs. This interface is part of the `CyberArk.PasswordVault.PublicInterfaces` assembly.

The debug log interface provides the following method:

```
void LogWrite(string format, params object[] arg);
```

The `IDebugLog` interface is implemented by the `ValidationParametersEx` class, which is received by the `Validate` method of the ticket validator module. Therefore, the logger can be used by performing the following cast:

```
IDebugLog log = (IDebugLog)parameters;
```

All log writes are mapped to the user's session log. The debug level needs to be set to high for the debug messages to appear in the log. The name of the ticket validator module will prepend any log write, e.g.:

```
EPVBL001D [Ticketing system AcmeTicketing] Test message
```

Configuring Dual Control together with Ticketing Integration

You can configure dual control together with ticketing integration so that users can only access privileged accounts after ensuring that they have a valid open ticket and/or receiving manual confirmation from authorized managers. This enforces validations of requests against an open ticket as well as enforcing a dual control approval workflow. Users can access accounts as follows:

- **Standard: Combined ticketing system integration and dual control** – Before accessing a privileged account, users are required to specify a ticketing system and the ID of a ticket that will be validated against the ticketing system in addition to creating a dual control request. According to configuration, tickets can be validated before the request is sent, after confirmation has been received from authorized managers, or at both stages. After the ticket has been validated and the user has received request confirmation, they can access the account. This workflow supports all operations for accounts, including Show, Copy and Connect and PSM.
- **Emergency: Ticketing system integration and dual control based on ticket type** – In this workflow, users are required to specify a ticketing system and the ID of a ticket that will be validated against the ticketing system. Depending on the type of ticket that has been specified, users may or may not be required to create a dual control request. For example, the system can allow users to access a privileged account after they provide a valid open incident ticket, while enforcing an approval workflow when the user provides a valid open change ticket. If dual control is not implemented for this ticket type, the ticket is immediately validated against the ticketing system and the account can be accessed.

Ticketing system integration and dual control are configured separately, and can be enforced independently of each other. These new workflows allow you to enforce different validation workflows for various ticketing systems and/or ticket types. This flexibility enables you to manage and audit multiple workflows, according to specific enterprise standards.

Step 1 - Activating Dual Control

1. Click **POLICIES**, then click **Master Policy** to display the Master Policy page.
2. In Privileged Access Workflows, select **Require dual control password access approval**.
3. In the **Rule Preview** pane, click the **VALUE** edit icon and then click **Active**.

Dual control is now configured at Master Policy level and users who wish to retrieve accounts and files are required to request access confirmation from at least one authorized user. By default, requests are retained for 30 days.

For more information about dual control, refer to the relevant sections:

- For information about setting confirmation permissions, defining which users require confirmation, and more, refer to *Dual Control*, page 251.
- For information about changing default settings, refer to *Dual Control*, page 513.

Step 2 – Defining Ticketing Systems (system level)

1. Click **ADMINISTRATION**, then in the System Configuration page click **Options**; the Web Access Options are displayed.
2. Expand the **Ticketing Systems** parameters, then add the relevant ticketing systems and configure their settings.

The defined ticketing systems can be enabled or disabled separately in each platform. For more information, refer to Step 3.

If you have already implemented a ticketing system integration module, refer to *Integrating with Ticketing Systems*, page 516, for updated interfaces/properties in this module.

3. To allow enabling dual control based on ticket type, set the following parameters in the ticketing System properties:
 - **IncidentTicketIdentification** – This parameter defines a regular expression that identifies **incident** tickets. If the ticketing system you are configuring only handles incident tickets, specify `.*` to match all ticket IDs that are specified for this ticketing system. If the ticketing system you are configuring only handles other ticket types, leave this parameter empty.
 - **ChangeTicketIdentification** – This parameter defines a regular expression that identifies **change** tickets. If the ticketing system you are configuring only handles change tickets, specify `.*` to match all ticket IDs that are specified for this ticketing system. If the ticketing system you are configuring only handles other ticket types, leave this parameter empty.

Note: If the ticket ID specified by the user matches both regular expressions, the ticket will be considered an Incident.
4. Click **OK** to save the new ticketing system configurations and return to the System Configuration page.

To Activate Dual Control together with Ticketing System Integration

1. In the System Configuration page, click **Options**; the Web Access Options are displayed.
2. Select **Dual Control**; the PWA Dual Control settings are displayed in the Properties list.
3. Set the **AllowDualControlWithTicketingIntegration** property. This determines whether or not both dual control and ticketing system integration can be implemented together. Set this parameter to **Yes**.

Step 3 - Configuring Platforms for Ticketing Integration with/without Dual Control

1. Click **ADMINISTRATION** to display the System Configuration page, then display **Platform Management**.
2. Select the Device that will use the ticketing system, then click **Edit**.
3. In **UI & Workflows**, display the **Ticketing System** parameters.
4. To enable ticketing system integration for this platform, specify the following parameters:
 - **EnterTicketingInfo** – Whether or not users will be required to enter ticketing system information before being able to retrieve an account. Set this parameter to **Yes**.
 - **ValidateTicketNumber** – Whether or not to perform an online validation against the ticketing system. To perform online validations of specified tickets against the ticketing system when users access a privileged account associated to this platform, set this parameter to **Yes**.

At this point, after you save the settings, all accounts that are assigned to a platform that is configured with dual control will require users to specify a valid ticket and send a request for confirmation, before they can access the account.

Advanced configurations

The following advanced configurations are **optional**.

5. You can enable only some of the ticketing systems that were defined for this platform. If you do not specify the list of active ticketing systems, all the systems defined at system level will be available when accessing accounts of this platform. However, if you define specific ticketing systems in a platform, only those ticketing systems will be displayed in the lists of ticketing systems that can be selected by users in the Show Password window when trying to access privileged accounts associated to the platform.

To activate specific ticketing systems for this platform

- i. In the selected platform, expand **Ticketing Systems**, then expand **ActiveTicketingSystems**.
- ii. Select **TicketingSystem** then, in the Name property, specify the name of the active ticketing system that is configured at system level. Specify the name of the ticketing system in exactly the same way as it is defined at system level. This ticketing system will be displayed in the list of ticketing systems for accounts associated to this platform.

6. You can define when the ticket ID specified by the user will be validated, by specifying the following parameters:
- **TicketValidationTimingWithDualControl** – Determines when the ticket will be validated. Options are:
 - **WhenSendingRequest** – The ticket will be validated when the user sends a request to access a privileged account.
 - **WhenAccessingAccountAfterConfirmation** – The ticket will be validated when the user accesses the account after the request has been confirmed by an authorized user. This is the **default** value.
 - **WhenSendingRequestAndWhenAccessingAccount** – The ticket will be validated twice; once when the user sends a request to access a privileged account and again when the user accesses the account after the request has been confirmed by an authorized user.

This is useful for implementations where the required validation differs according to the timing of the validation. For instance, you may want to validate that users access privileged accounts within the ticket timeframe. It is reasonable to perform this type of validation only when the user accesses the account and not when a request to access the account is sent, as this is usually done ahead of the change ticket timeframe. You can set different validations when users send requests and when they access accounts (after confirmation). For more information, refer to *Integrating with Ticketing Systems*, page 516.

7. You can define whether dual control approval workflow is needed for specific ticket types or always.

Note: Before setting these workflows at platform level, make sure that the **IncidentTicketIdentification** and **ChangeTicketIdentification** parameters are set at system level. For more information, refer to Step 2.

- **DisableDualControlForIncidentTickets** – Whether or not dual control will be disabled for incident tickets. The default value is **No**, indicating that dual control is enforced for incident tickets.
 - **DisableDualControlForChangeTickets** – Whether or not dual control will be disabled for change tickets. The default value is **No**, indicating that dual control is enforced for change tickets.
8. Click **Apply** to apply the new configurations immediately,
or,
Click **OK** to save the new configurations and return to the System Configuration page.

Configuring Transparent Connections

The Privileged Account Security solution can be configured to enable users to transparently log onto target Windows machines and SSH devices directly from the PVWA application. The Master Policy determines whether or not users will be able to access target machines transparently, and if users can view the password that is used to log on. Other configurations include setting global system parameters in the Web Access Options. Most of the global settings can be overridden at platform level, and some settings can be configured at account level.

Currently, the PVWA supports transparent connections to remote Windows machines and SSH devices on the following platforms:

Windows connections:

- Windows Domain Accounts
- Windows Local Accounts
- Windows Local Desktop Accounts

Note: To enable transparent connections, the end user's machine must have Microsoft RDP Client 5.2 or higher installed.

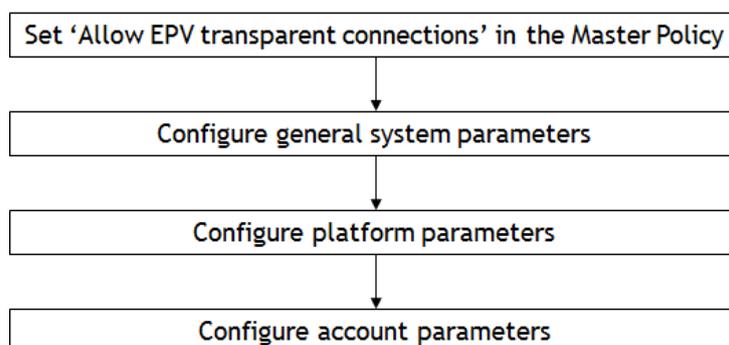
Privileged SSO connections:

- Unix via SSH
- DB2 on Unix via SSH
- Informix on Unix via SSH
- NetScreenSSH
- CiscoPixSSH
- CiscoSSH
- Any SSH device

Transparent connections are supported on IPv4 and IPv6 for the following platforms:

- Windows
- Unix via SSH
- DB2 on Unix via SSH
- InformixUnixSSH

Configuring transparent connections comprises several stages. These stages are described below in the order in which they must be performed:



Requirements

To connect to remote Windows machines:

- Configure the remote machine to allow remote connections.

To connect to remote SSH devices:

The following version of Java must be installed on the local end user machine:

- JRE (Java Runtime Environment) 1.4 or higher

All passwords that will be used to log onto remote devices must have the following account properties:

- **Username** – The name of the user that will be used to log onto the remote device.
- **Address** – The IP/DNS address of the remote machine (for Windows Desktop and SSH passwords) or the Windows Domain name (for Windows domain accounts).

Passwords that are used to connect to Windows machines must supply the user domain name as well as the address of the machine. The domain name can be taken from the Address property if the Address specifies a short NETBIOS domain name. Otherwise, the short NETBIOS domain name must be specified in a new account property called **Logon To**.

If the CPM is configured for password auto-detection, this property will be set automatically when a password is detected. For more information refer to *Configuring Automatic Provisioning*, page 390. However, if a password is added or edited manually, the user can specify it with all the other account properties. The platform can be configured to resolve the domain name from the existing Address account property, if possible.

Notes: The PVWA will only be able to detect the logon domain name automatically in the following scenarios:

- The PVWA is installed on Windows 2008 or higher.
- The PVWA is installed in the same domain as the remote machine or on a domain that is trusted by the remote machine's domain.

Configuring Transparent Connections

Note: These instructions are for configuring transparent connections, **not** PSM connections. For more information about configuring PSM connections, refer to *Configuring PSM Connections*, page 621.

1. Give users the appropriate Safe permissions:
 - i. Click **POLICIES**, then click **Access Control (Safes)**.
 - ii. In the Safes page, open the Safe where the accounts that will be used to connect to remote devices transparently are stored.
 - iii. Give users who will use accounts the appropriate permissions in the Safe(s) where the accounts are stored:
 - Retrieve accounts

Note: The **AllowViewingPasswords** parameter in the platform Connection Components section determines whether or not the user will be able to view and copy the password. Users who have the "Manage Safe" and "Retrieve accounts" authorizations can view the password, regardless of this parameter setting.

2. Configure the general system parameters:

The **Connection Components** settings in Web Access Options define privileged SSO and transparent connections to remote devices, and determine how each connection will be performed. By default, all the connection components are configured. You can view them in the Web Access Options.

- i. Click ADMINISTRATION, and in the System Configuration page, click Options; the Web Access Options page appears.
- ii. Click **Connection Components**, and select either **RDP** or **SSH** to display the parameters for the connection component to configure.
- iii. See the table below for more details about the allowed parameters:
 - a. **Component parameters** include parameters that enable the remote connection.
 - b. **User parameters** include parameters that prompt users for more information.
- iv. Some parameters are defined automatically during installation and others can be added manually.

To add a new parameter:

 - a. Right-click on **Component Parameters** or **User parameters**, then select **Add Parameter**; a new parameter is added.
 - b. Specify the name and value of each property, then click **OK** to save the new parameters.

The Connection Components parameters are described in the table below:

Parameter	Description	RDP	SSH	Override at platform level	Override at account level
ConnectionComponent RDP/SSH (root level)					
ID	A unique ID that identifies the connection parameters. A default SSH connection is configured during installation.	✓	✓	✗	✗
FullScreen	Whether or not the remote desktop window will be opened in full screen mode. The full screen mode opens a new window with an additional window for logon. You can toggle between screen modes with Alt+Ctrl+Break. Default value: No.	✓	✓	✗	✗
Height	The height in pixels of the desktop resolution on the remote machine. The height of the window that is opened on the remote desktop is calculated from this parameter. Default value: 768 pixels.	✓	✓	✗	✗

Parameter	Description	RDP	SSH	Override at platform level	Override at account level
Width	The width in pixels of the desktop resolution on the remote machine. The width of the window that is opened on the remote desktop is calculated from this parameter. Default value: 1024 pixels.	✓	✓	✗	✗
EnableWindow Scrollbar	Whether or not scrollbars will be added to the transparent connection logon window. Default value: No.	✓	✗	✗	✗
Type	Specifies the interface that is used for the connection. This is an internal parameter <ul style="list-style-type: none"> ◆ The default value for privileged SSO connections is: 'CyberArk.TransparentConnection.SSH.SSHConnectionComponent, CyberArk.PasswordVault.TransparentConnection.SSH' ◆ The default value for Windows transparent connections is: 'CyberArk.PasswordVault.TransparentConnection.RDP.RDPConnectionComponent, CyberArk.PasswordVault.TransparentConnection.RDP' 	✓	✓	✗	✗
DisplayName	Defines the display name of the connection component.	✓	✓	✗	✗
Component Parameters					
Port	The port used to connect to the remote device. If this parameter is not set as an account property, it must be specified in the Component parameters. The port number can be overridden by specifying it in individual platforms. <ul style="list-style-type: none"> ▪ Default port for SSH connections: 22 ▪ Default port for Windows transparent connections: 3389. 	✓	✓	✓	✓

Parameter	Description	RDP	SSH	Override at platform level	Override at account level
Advanced Settings4. Authentication Level	<p>Encrypts the RDP session between the user's computer and the remote device. Valid values are:</p> <ul style="list-style-type: none"> ▪ 0 – No SSL encryption ▪ 1 – Requires SSL connection ▪ 2 – Try to use an SSL connection. If the SSL isn't available, display a warning message asking the user whether to connect without SSL encryption. 	✓	✗	✓	✗
SSH Specific parameters	<p>These parameters configure SSH connections. Note: These parameters are for transparent SSH connections, not for PSM-SSH connections.</p>	✗	✓	✓	✗
protocol	<p>The preferred protocol.</p> <ul style="list-style-type: none"> ▪ Acceptable values: auto/ssh1/ssh2 ▪ Default value: auto 	✗	✓	✓	✗
proxy-type	<p>The type of proxy server to use to connect.</p> <ul style="list-style-type: none"> ▪ Acceptable values: none/http/socks4/socks5 ▪ Default value: none 	✗	✓	✓	✗
proxy-host	The name of the proxy server to use to connect.	✗	✓	✓	✗
proxy-port	The port on the proxy server to use to connect.	✗	✓	✓	✗
proxy-user	The username to authenticate with a proxy server.	✗	✓	✓	✗
ssh1-cipher	<p>The name of the block cipher to use in ssh1.</p> <ul style="list-style-type: none"> ▪ Acceptable values: Blowfish-cbc/3des-cbc/idea-cbc ▪ Default value: blowfish-cbc 	✗	✓	✓	✗
ascii-line	<p>Use ASCII Line-draw-characters instead of drawing.</p> <ul style="list-style-type: none"> ▪ Acceptable values: true/false 	✗	✓	✓	✗
auto-linefeed	<p>Auto-linefeed.</p> <ul style="list-style-type: none"> ▪ Acceptable values: true/false 	✗	✓	✓	✗
autowrap	<p>Auto-wrap lines if the output reaches the edge of the window.</p> <ul style="list-style-type: none"> ▪ Acceptable values: true/false 	✗	✓	✓	✗

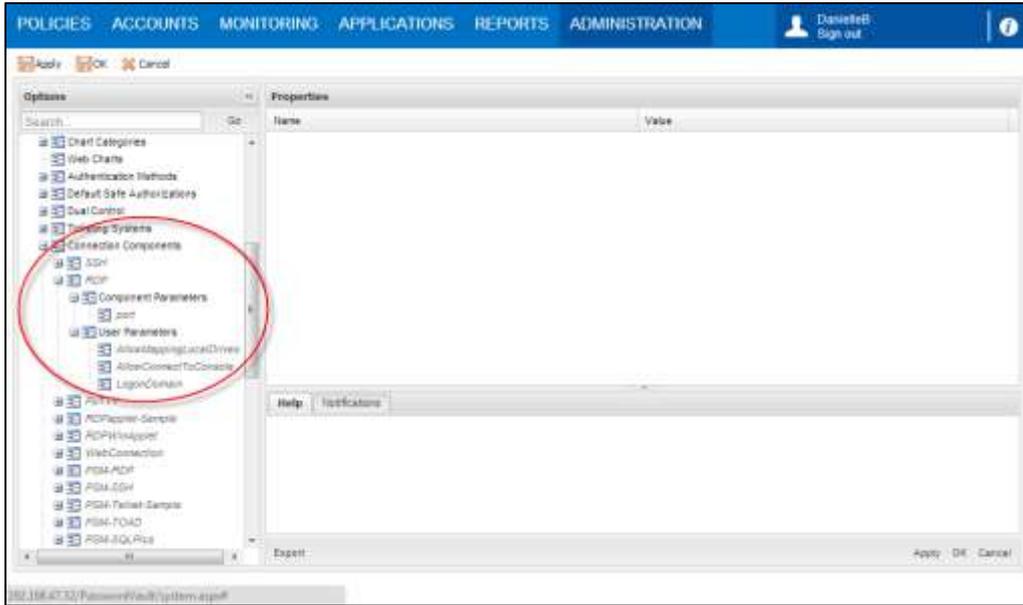
Parameter	Description	RDP	SSH	Override at platform level	Override at account level
backspace-send	What to send on BACKSPACE: BS (^h, 0x08), DEL (^?, 0x7f), or ERASE (^E[3~]).	x	✓	✓	x
bg-color	Background color (or ',, '). Default value: white.	x	✓	✓	x
copy-select	Copy when selected with mouse. ▪ Acceptable values: true/false	x	✓	✓	x
cursor-color	Cursor color (<name> or '<r>,<g>,') (name of colors are: black, red, green, yellow, blue, magenta, cyan, white, i_black, i_red, i_green, i_yellow, i_blue, i_magenta, i_cyan, i_white).	x	✓	✓	x
delete-send	Character to send on DELETE: BS (^h, 0x08), DEL (^?, 0x7f), or ERASE (^E[3~]).	x	✓	✓	x
fg-color	Foreground color (<name> or '<r>,<g>,'). Default value: black.	x	✓	✓	x
font-name	The name of the font to use in the terminal.	x	✓	✓	x
font-size	The size of the font to use in the terminal.	x	✓	✓	x
insert-mode	Toggles insert mode. ▪ Acceptable values: true/false	x	✓	✓	x
paste-button	The mouse button that will be used to paste. ▪ Acceptable values: shift+left/middle/right	x	✓	✓	x
save-lines	The number of lines to save in the scrollbar buffer.	x	✓	✓	x
scrollbar	The scrollbar position. ▪ Acceptable values: none/left/right ▪ Default value: right	x	✓	✓	x
visible-cursor	Toggles if cursor is visible or not. ▪ Acceptable values: true/false	x	✓	✓	x

Parameter	Description	RDP	SSH	Override at platform level	Override at account level
User Parameters					
AllowMappingLocalDrives	Whether or not the local hard drives will be redirected to the remote server. Note: This is not supported for remote devices that run on Windows 2000.	✓	✗	✓	✗
AllowConnectToConsole	Whether the PVWA will connect to the administrative console of the remote machine or will open a new remote session.	✓	✗	✓	✗
LogonDomain	The Windows account property whose value will be used to resolve the logon domain. Note: If this parameter is specified in the account, it will not be displayed in the connection dialog.	✓	✗	✓	✓

Each User Parameter has the following settings:

Name	The name of the parameter.
DisplayName	The exact way that the parameter name will be displayed in the connection window.
Value	The default value of this parameter.
Visible	Whether or not the user will be prompted for this parameter before the connection is established.
Required	Whether or not the user is required to provide this information for the remote connection to be activated.
Type	The type that will be used to modify the appearance or behavior of a parameter UI field. Note: This is an internal parameter and must not be changed.
EnforceInDualControlRequest	Whether or not the user will be required to provide information in order to create a dual control request.

The following example shows the Connection Components parameters in Web Access Options:



3. Configure the platform parameters. These parameters can be configured to override transparent connection configurations set at general system level (in the previous step).
 - i. In the Platform Management page, select the platform to configure, then click **Edit**; the platform settings page appears.
 - ii. Expand UI & Workflows, then expand **Connection Components**.
 - iii. Select an existing connection component or create a new one, then configure it using the parameters described below.
 - iv. See the table below for more details about the allowed parameters:
 - a. **Override Component Parameters** include parameters that override the general component parameters.
 - b. **Override User Parameters** include parameters that override the general user parameters.
 - v. To define a subsection of override parameters, right-click on the component to configure, then select **Add Override Component Parameters** or **Add Override User Parameters**; a new section for these override parameters is created.
 - vi. To add parameters to these sections, right-click the name of the section, then select **Add Parameter**; a new parameter is added.
 - vii. Specify the name and value of each parameter, then click **OK** to save the new parameters.

The following table lists the platform's Connection Components parameters:

Parameter	Description	RDP	SSH	Override at account level
Component Connection Settings				
TransparentConnectionDefault	Defines the default connection component for direct transparent connections to remote machines.	✓	✓	✗
PSMConnectionDefault	Defines the default connection component for connections via PSM.	✓	✓	✗
EnforceTransparentConnectionInDualControl	Whether or not transparent connections will be enforced in dual control.	✓	✓	✗
Component Parameters				
Port	The port used to connect to the remote device. If this parameter is not set as an account property, it must be specified in the Component parameters, as shown in the example above. The port number can be overridden by specifying it in individual platforms. <ul style="list-style-type: none"> Default port for SSH connections: 22 Default port for Windows transparent connections: 3389 	✓	✓	✓
SSH Specific parameters	These parameters configure SSH connections. Note: These parameters are for transparent SSH connections, not for PSM-SSH connections.	✗	✓	✗
protocol	The preferred protocol. <ul style="list-style-type: none"> Acceptable values: auto/ssh1/ssh2 Default value: auto 	✗	✓	✗
proxy-type	The type of proxy server to use to connect. <ul style="list-style-type: none"> Acceptable values: none/http/socks4/ socks5 Default value: none 	✗	✓	✗
proxy-host	The name of the proxy server to use to connect.	✗	✓	✗
proxy-port	The port on the proxy server to use to connect.	✗	✓	✗

Parameter	Description	RDP	SSH	Override at account level
proxy-user	The username to authenticate with a proxy server.	x	✓	x
ssh1-cipher	The name of the block cipher to use in ssh1. <ul style="list-style-type: none"> Acceptable values: Blowfish-cbc/3des-cbc/idea-cbc Default value: blowfish-cbc 	x	✓	x
ascii-line	Use ASCII Line-draw-characters instead of drawing. <ul style="list-style-type: none"> Acceptable values: true/false 	x	✓	x
auto-linefeed	Auto-linefeed. <ul style="list-style-type: none"> Acceptable values: true/false 	x	✓	x
autowrap	Auto-wrap lines if the output reaches the edge of the window. <ul style="list-style-type: none"> Acceptable values: true/false 	x	✓	x
backspace-send	What to send on BACKSPACE: BS (^h, 0x08), DEL (^?, 0x7f), or ERASE (^E[3~]).	x	✓	x
bg-color	Background color (or ',, '). Default value: white.	x	✓	x
copy-select	Copy when selected with mouse. <ul style="list-style-type: none"> Acceptable values: true/false 	x	✓	x
cursor-color	Cursor color (<name> or '<r>,<g>,') (name of colors are: black, red, green, yellow, blue, magenta, cyan, white, i_black, i_red, i_green, i_yellow, i_blue, i_magenta, i_cyan, i_white).	x	✓	x
delete-send	Character to send on DELETE: BS (^h, 0x08), DEL (^?, 0x7f), or ERASE (^E[3~]).	x	✓	x
fg-color	Foreground color (<name> or '<r>,<g>,'). Default value: black.	x	✓	x
font-name	The name of the font to use in the terminal.	x	✓	x

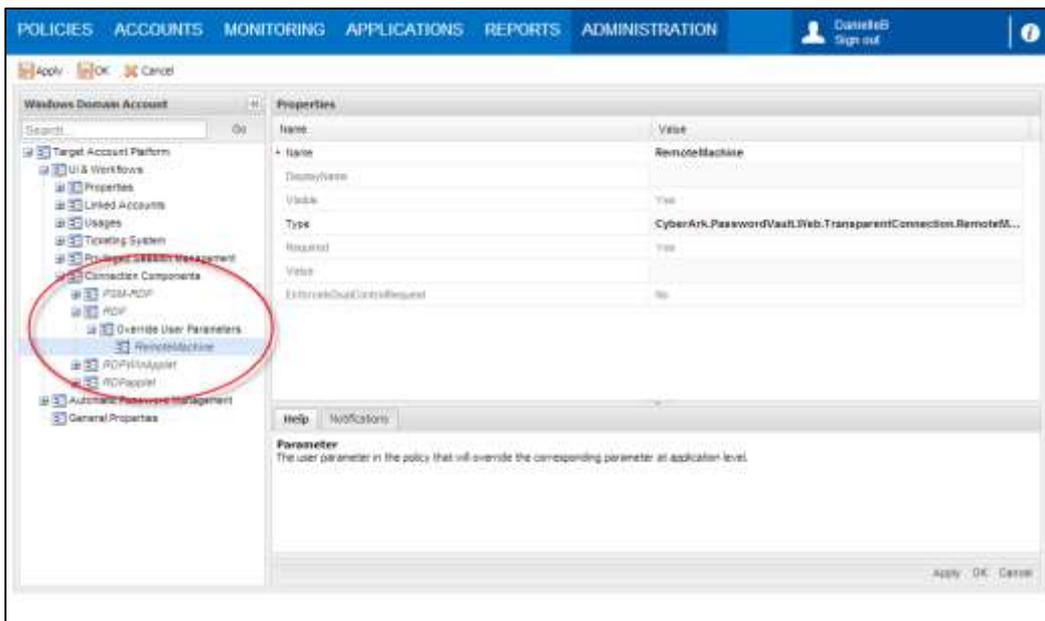
Parameter	Description	RDP	SSH	Override at account level
font-size	The size of the font to use in the terminal.	x	✓	x
insert-mode	Toggles insert mode. <ul style="list-style-type: none"> Acceptable values: true/false 	x	✓	x
paste-button	The mouse button that will be used to paste. <ul style="list-style-type: none"> Acceptable values: shift+left/middle/right 	x	✓	x
save-lines	The number of lines to save in the scrollbar buffer.	x	✓	x
scrollbar	The scrollbar position. <ul style="list-style-type: none"> Acceptable values: none/left/right Default value: right 	x	✓	x
visible-cursor	Toggles if cursor is visible or not. <ul style="list-style-type: none"> Acceptable values: true/false 	x	✓	x
User Parameters				
AllowMappingLocalDrives	Whether or not the local hard drives will be redirected to the remote server. Note: This is not supported for remote devices that run on Windows 2000.	✓	x	x
AllowConnectToConsole	Whether the PVWA will connect to the administrative console of the remote machine or will open a new remote session.	✓	x	x
LogonDomain	The Windows account property whose value will be used to resolve the logon domain. Note: If this parameter is specified in the account, it will not be displayed in the connection dialog.	✓	x	✓
RemoteMachine	The name of the remote device to connect to.	Windows Domain account only	SSH Domain account only (eg, NIS, KEON, etc.)	x

Parameter	Description	RDP	SSH	Override at account level
-----------	-------------	-----	-----	---------------------------

Each User Parameter has the following settings:

Name	The name of the parameter.
DisplayName	The exact way that the parameter name will be displayed in the connection window.
Value	The default value of this parameter.
Visible	Whether or not the user will be prompted for this parameter before the connection is established.
Required	Whether or not the user is required to provide this information for the remote connection to be activated.
Type	The type that will be used to modify the appearance or behavior of a parameter UI field. Note: This is an internal parameter and must not be changed.

The following example shows the Connection Components parameters in the default Windows Domain Account platform:



4. Configure the account parameters. The parameters that can be configured are described in the table below:
- i. Add the account that will be used to log onto the remote device transparently, or,
In the Accounts Details page of the account that will be used to log onto the remote device transparently, click **Edit**.
 - ii. In the Optional Properties section, specify either of the following parameters:

Parameter	Description	RDP	SSH
Port	The port used to connect to the remote device. If this parameter is not set as an account property, it must be specified in the Component parameters, as shown in the example above. The port number can be overridden by specifying it in individual platforms. <ul style="list-style-type: none"> ▪ The default port for privileged SSO connections is 22. ▪ The default port for Windows transparent connections is 3389. 	✓	✓
LogonDomain	The Windows account property whose value will be used to resolve the logon domain.	✓	✗

5. Click **Apply** to save the new parameter values and stay in the platform's settings page,
or,
Click **OK** to save them and return to the System Configuration page. The changes will be applied after the period of time specified in the **ConfigurationRefresh Interval** parameter.

Configuring Transparent Connections to Websites and Web Applications

CyberArk's web transparent connection enables users to use privileged accounts information stored in the Privileged Account Security solution to simply "click to connect" to a target web interface. This connection can be used to automatically access enterprise applications such as proprietary enterprise applications, as well as websites (SaaS) such as corporate Facebook accounts, corporate LinkedIn accounts, Salesforce, and many others.

Known Limitations

The web transparent connection is not supported in the following websites:

- Websites that use captcha authentication, due to the automatic authentication mechanism. For more details see <http://en.wikipedia.org/wiki/CAPTCHA>.
- Websites that use Flash technology in their authentication page.

The following example shows elements used in HTML code that uses Flash:

```
<embed src="https://mywebsite.com/ui/login.swf" ...  
type="application/x-shockwave-flash">
```

- Microsoft Live.com websites (such as Hotmail, Skydrive).
- Websites in which the login button is defined as an image, and not as a button. The following HTML code shows an example of a login button that is defined as an image:

```
<input type="image" ... onclick="...">
```

In addition, the web transparent connection technology may not support other websites. If you have problems connecting to a website, refer to the *Troubleshooting*, page 546.

Configuring a New Website Platform

The listed in the Website device represent websites whose accounts will be managed in the Vault. You can create as many platforms as necessary to manage different website accounts. A web transparent connection to the Facebook platform is automatically configured during installation.

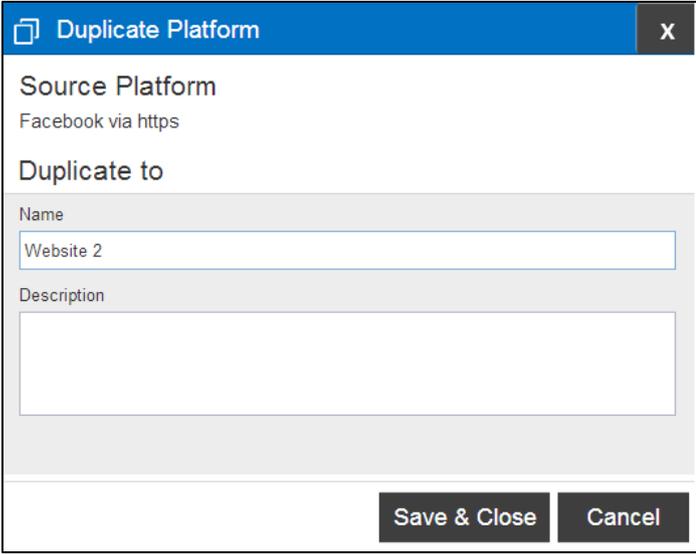
For example, you can create a platform called 'SalesForce' to manage accounts used to log onto salesforce.com. The platform stores all the configurations that are required to enable the account to log onto Salesforce automatically.

Before configuring a new website platform as described below, enable the **Allow EPV transparent connections ('Click to connect')** rule in the Master Policy.

Step 1: Duplicate or add a platform

Note: This step is only relevant if a platform for the website is not yet defined.

1. Click **ADMINISTRATION** to display the **System Configuration** page, then click **Platform Management** to display a list of supported target account platforms.
2. Select an existing platform that is similar to the new target account platform, then click **Duplicate**; the Duplicate Platform window appears.



The image shows a 'Duplicate Platform' dialog box. The title bar is blue with a white icon and the text 'Duplicate Platform' and a close button 'X'. The main area is white and contains the following fields:

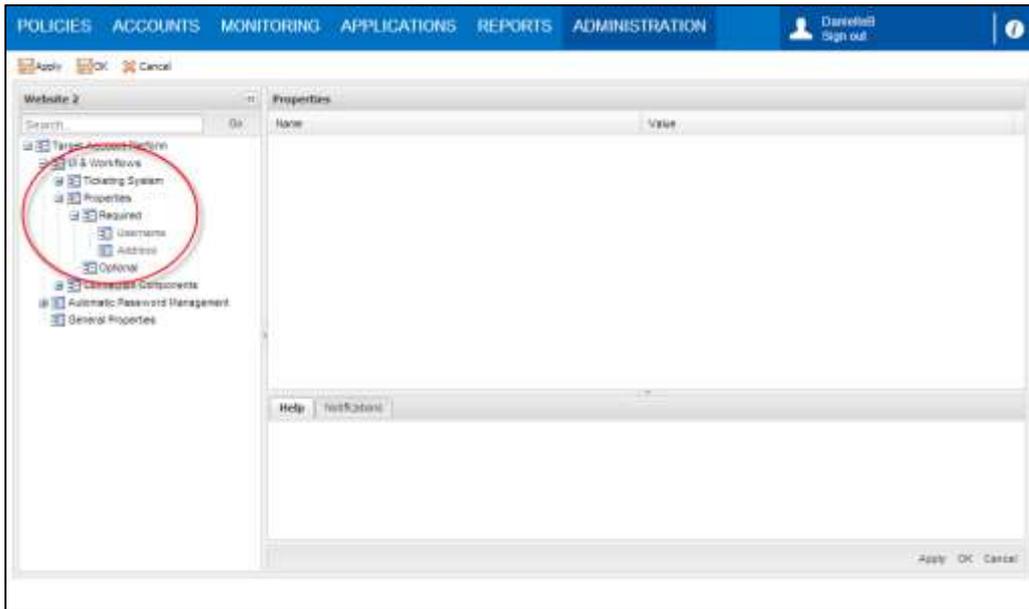
- Source Platform:** Facebook via https
- Duplicate to:**
 - Name:** Website 2
 - Description:** (empty text area)

At the bottom of the dialog are two buttons: 'Save & Close' and 'Cancel'.

3. Type the name and a description of the new platform, then click **Save & Close** to create the new platform; the new platform appears in the list of Target Account Platforms.
4. Select the new platform then click **Edit**; the Target Account settings page appears.
5. Expand **UI & Workflows**, then **Properties**, and then expand **Required**; a list of required properties is displayed.
6. For websites with more than just the username and password login fields, add the additional login fields:
 - i. Right-click **Required**, then from the pop-up menu, select **Add Property**; a new property is added to the list of required properties.
 - ii. Select the new property then, in the Properties list, specify the **Name** of the new property. This is the name of a field that appears on the authentication page of the target website.

Note: By default, the username property is a required property for all devices. If there are only two fields in the authentication page (username and password), you do not have to reconfigure it by creating a new property for this field.
 - iii. Create new properties for each field on the login form of the target website. When the account for this website is created, users will be prompted for this information.
 - iv. For each of these additional properties, create an account property in the Digital Vault. For more information, refer to *Defining Custom Account Properties*, page 161.

The following example shows a configured platform for the Website device.

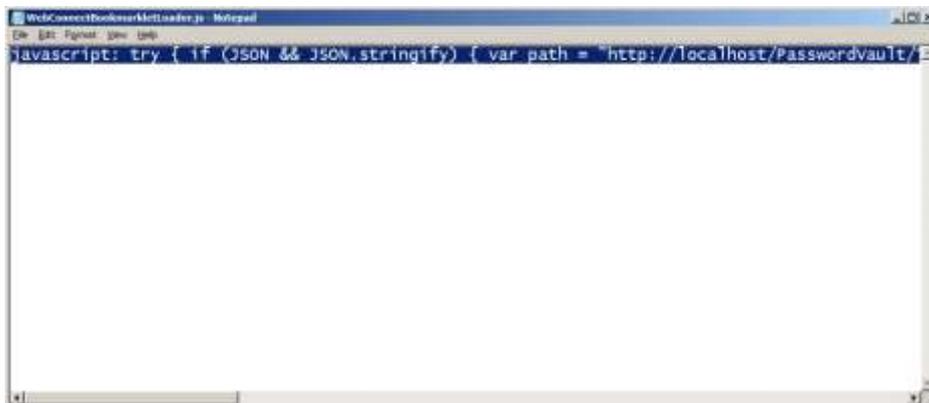


Step 2: Collect the required information from the target website for the PVWA connection component

This step explains how to use a bookmarklet to collect the required information from the target website that is required by the PVWA to create a connection.

Note: A bookmarklet is similar to a bookmark in a browser that contains a JavaScript code which runs locally on the browser. The bookmarklet that you will create in the following step gathers information from the target website that you require for the PVWA to connect to it.

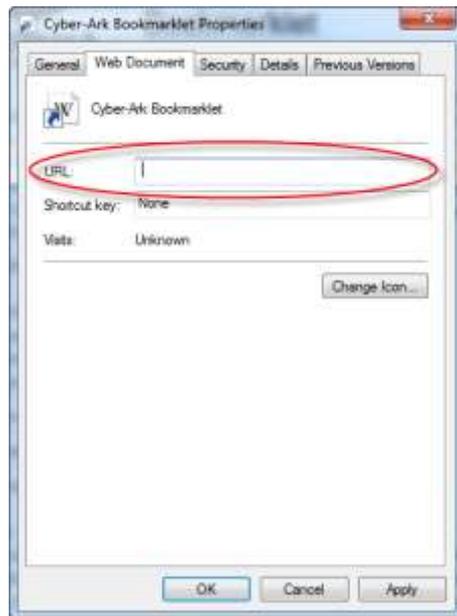
1. Edit the JavaScript file:
 - i. On the IIS, in the PasswordVault installation folder, display the following folder: **TransparentConnection\ConnectionScripts**.
 - ii. Open **WebConnectBookmarkletLoader.js** in editing mode, for example, with Notepad.
 - iii. In the URL, replace 'localhost' with the address of your PVWA. Use either an IP or a DNS address.
 - iv. Copy the content of the loader script. This text starts with **javascript:**.



- v. Save the JavaScript file.

2. Create a bookmarklet:

- i. Open your browser and add a new bookmark.
- ii. In the Name field, specify **CyberArk Bookmarklet**.
- iii. Display the Properties of the new bookmark:
 - Internet Explorer: In the Favorites menu, right- click the new bookmark then select Properties.
 - Firefox: In the Bookmarks menu, right- click the new bookmark then select Properties.
 - Chrome:
 - a. Click the **Tools** icon, then select **Bookmarks**.
 - b. Right-click the new bookmark, then select **Edit**.



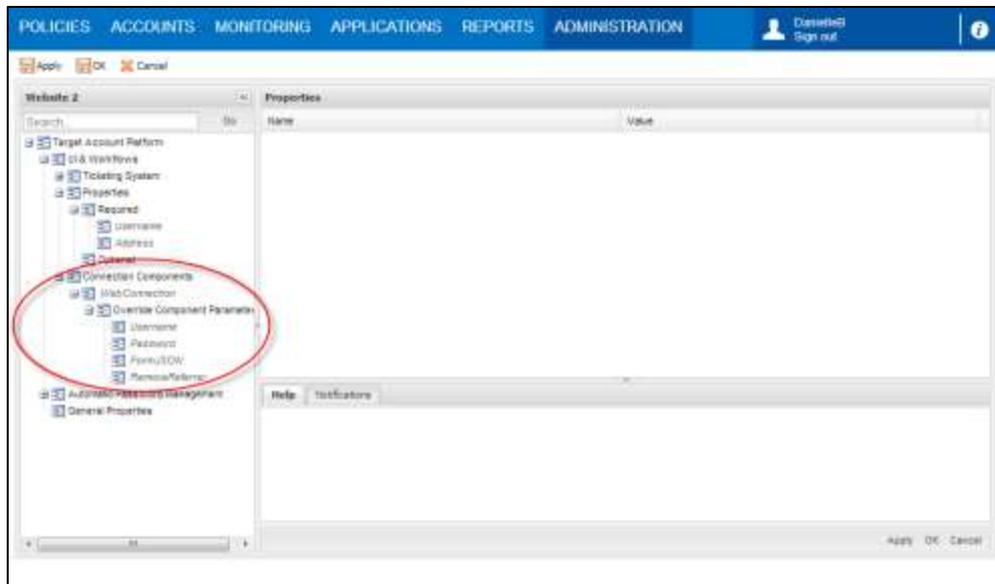
- iv. In the Bookmarklet Properties window, in the URL field, replace the current URL with the text that you copied from the WebConnectBookmarkletLoader.js file.
 - v. Click **OK** or **Save** (depending on your browser) to save the changes in the bookmarklet.
3. Run the bookmarklet to gather the required information:
- Display the authentication page of the target website, then click the bookmarklet.
Note: Do not specify any information in the authentication page.
- A pop-up window appears with all the information required by the PWA to create a connection component to the target website.
Do not close this window.

Step 3: Configure the “WebConnection” connection component in the platform

1. Create a connection component for the target website:
 - i. Select the platform to configure, then click **Edit**; the platform settings page appears.
 - ii. Expand **UI & Workflows**, then right-click **Connection Components** and select **Add Connection Component**; a new connection component is added to the list.
 - iii. Select the new connection component and specify the following properties:
 - **ID** – Specify **WebConnection**.
 - iv. Right-click the new connection component, WebConnection, then select **Add Override Component Parameters**; a new Override Component Parameters section for these parameters is created. This is where you will add the login fields displayed in the information collected and displayed by the bookmarklet.
 - v. For each collected login field (Username, Password, FormJSON), right-click **Override Component Parameters**, then select **Add Parameter**; a new parameter is added.
 - vi. In the Properties list, specify the following:
 - **Name** – The name of the login field collected by the bookmarklet.
 - **Value** – The value of the specified login field. Make sure you copy and paste all the relevant text collected by the bookmarklet.

Repeat this step to create parameters for all the **login fields** collected by the bookmarklet. By default, these parameters are **Username**, **Password**, and **FormJSON**.

The following example displays the parameters in a connection component for the bookmarklet displayed above.



The following table lists all the parameters that can be used in connection components to enable transparent connections to remote websites.

Parameter	Description
Username	Represents the value of the username that is used to log onto the remote website. It is specified in the 'name' attribute in the details collected by the CyberArk bookmarklet.
Password	Represents the value of the password that is used to log onto the remote website with the username specified above. It is specified in the 'name' attribute in the details collected by the CyberArk bookmarklet.
FormJSON	Represents the JSON representation of the login form. When connecting to a website using a web transparent connection, this JSON object converts into an HTML form, into which the connection component injects the required username and password and then submits the form.
DisplayDebuggingMessage	Whether or not debugging messages are enabled for connection components in platforms. This parameter can be set in the Connection Components parameters in Web Access Options to apply to every platforms that uses this connection component, or in individual platforms to enable debugging messages for specific platforms. Valid values: Yes/No. Default value: No.
ConnectionScriptName	The name of the connection script to use to connect to remote websites. In web transparent connections, specify GenericWebConnector.js. This parameter is configured in the Connection Component parameters in Web Access Options.
SubmitFunction	A string that represents the Javascript code that runs just before the login form is submitted. It mimics the "onsubmit" attribute in the original login form. This attribute runs after the user clicks Connect or Login. In most cases, this string contains a reference to a Javascript function that validates connection details and can make changes in the login fields. When connecting with a web transparent connection, the 'onsubmit' attribute doesn't run automatically. Therefore, any required Javascript code must be configured manually. For example, when configuring a web transparent connection to Salesforce, this parameter is mandatory. This parameter is configured in the platform's connection component.

Parameter	Description
RemoveReferrer	<p>A parameter that is required in specific connection components, depending on the website to which the connection component will connect. For example, this parameter is required to connect to Facebook.</p> <p>This parameter is configured in the platform's connection component.</p> <p>Valid values: Yes/No. Default value: No.</p>

Some websites require you to add a JavaScript function that will run before completing the connection. If a website requires this function, the bookmarklet displays 'OnSubmit Function Detected'.

For more information about identifying and configuring this function, refer to *The bookmarklet detected an OnSubmit function*, page 546, in *Troubleshooting*.

2. Click **Apply** to apply the new platform configurations and stay in the platform settings page.
or,
Click **OK** to save the new platform configurations and return to the System Configuration page.

Troubleshooting

The following troubleshooting options guide you through the main issues that may prevent web transparent connections from connecting to remote websites or web applications. For more information, contact your CyberArk support representative.

The bookmarklet detected an OnSubmit function

Some websites require you to add a JavaScript function that will run before completing the connection. If a website requires this function, the bookmarklet displays 'OnSubmit Function Detected'. However, not all websites that show the 'OnSubmit function detected' in the information gathered by the bookmarklet require the SubmitFunction to be configured. Try connecting to the website first without configuring this parameter and, if the PVWA cannot connect, then configure it.

Note: The "onSubmit" function validates login sessions and parameters, and is also used to perform activities that are required for login. For example, it can move the value specified in the Username field into another field which is not shown to regular users, such as a hidden field or trigger a change the way the site will behave after the login form is submitted. This function is run in manual login procedures immediately after the user clicks submit or login.

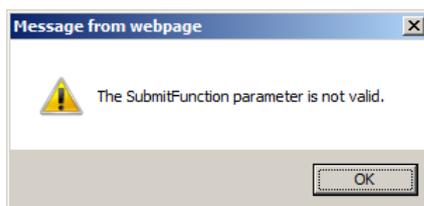
To add a JavaScript function

You can add a JavaScript function that will run when connecting to the remote website.

1. In the platform's connection component parameters, right-click **Override Component Parameters**, then select **Add Parameter**; a new parameter is added.
2. In the Properties list, specify the following:
 - **Name** – The name of the parameter required to log onto the website. Specify **SubmitFunction**.
 - **Value** – The value of the SubmitFunction parameter. Specify the function to execute. Make sure that all the code is included in one line.
3. Click **Apply** to apply the new configurations and stay in the platform settings page.
or,
Click **OK** to save the new configurations and return to the System Configuration page.

Can't connect to Salesforce in Internet Explorer using the SubmitFunction

The following message appears when you try to connect to Salesforce in Internet Explorer:



This issue can be solved by adding the SubmitFunction parameter to the WebConnection platform.

To configure a connection component to Salesforce in Internet Explorer

1. In the **WebConnection** connection component, right-click **Override Component Parameters**, then select **Add Parameter**; a new parameter is added.
2. In the Properties list, specify the following:
 - **Name** – The name of the parameter required to log onto the website. Specify **SubmitFunction**.
 - **Value** – The value of the SubmitFunction parameter. Specify `“document.getElementsByName(‘un’)[0].value = document.getElementsByName(‘username’)[0].value;”`. Make sure that all the code is included in one line.
3. Click **Apply** to apply the new configurations and stay in the platform settings page.
or,
Click **OK** to save the new configurations and return to the System Configuration page.

Transparent connections only work on the browser in which the bookmarklet was used

In some cases, connection components can only connect to the target website using the browser in which the bookmarklet was used to gather the information required for configuration.

To configure a connection for a single browser

- If all PVWA users only connect to the website on one browser, define a website platform using the bookmarklet run on the same browser,

To configure a connection for multiple browsers

1. In the website platform, create multiple connection components and name each one for a different browser. For example, GmailIEConnection and GmailFirefoxConnection.
2. In one of the browsers that will be used to connect to the website, create a bookmarklet and run it to gather the information that is required to define the platform connection component. For more information, refer to **Step 2: Collect the required information from the target website for the PVWA connection component**, page 542.
3. In the platform connection component for the browser on which you just ran the bookmarklet, define the override component parameters. For more information, refer to **Step 3: Configure the “WebConnection” connection component in the platform**, page 544.
4. Repeat steps 2 and 3 to configure platform connection components for each browser.

The next time users want to connect to a website with a specific account, they will be able to select the connection component that is configured for the relevant browser.

After clicking Connect, a new window opens and then immediately closes

The following procedure prevents a new window from opening and then immediately closing after you click **Connect** to start a transparent connection.

1. In the **WebConnection** connection component parameters, set **EnableToolbars** to **Yes**.
2. Click **Apply** to apply the new configurations and stay in the platform settings page.
or,
Click **OK** to save the new configurations and return to the System Configuration page.

Other Troubleshooting Options

The **RemoveReferrer** parameter can solve general troubleshooting issues. By default, this parameter is set to No, even if it is not configured.

Note: This parameter is mandatory for connections to certain websites, for example Facebook.

1. In the platform's connection component parameters, right-click **Override Component Parameters**, then select **Add Parameter**; a new parameter is added.
2. In the Properties list, specify the following:
 - **Name** – The name of the parameter required to log onto the website. Specify **RemoveReferrer**.
 - **Value** – The value of the RemoveReferrer parameter. Specify **Yes**.
3. Click **Apply** to apply the new configurations and stay in the platform settings page.
or,
Click **OK** to save the new configurations and return to the System Configuration page.

Displaying Debug Messages

Debug messages contain information that can help you identify connection and configuration issues, and solve them. Connection components include a debugging feature that is activated by the **DisplayDebuggingMessage** parameter in each connection component. This enables you and your support representative to work out and solve connection issues.

To Display Debug Messages

1. In the System Configuration page, click **Options**, then expand **Connection Components**; the defined connection components are displayed.
2. Expand the WebConnection connection component, and then expand **Component Parameters**; the defined parameters are displayed.
3. Add the debug message parameter:
 - i. Right-click **Component Parameters**, then select **Add Parameter**; a new parameter is added to the list.
 - ii. Select the new parameter and specify its properties.
 - iii. In the **Name** property, specify **DisplayDebuggingMessage**.
 - iv. In the **Value** property, specify **Yes**.
4. Click **Apply** to save the new parameter values and stay in the Web Access Options page,
or,
Click **OK** to save them and return to the System Configuration page.

Configuring PSM Connections and EPV RDP Connections that Require an External Tool

Users who want to access remote devices through the following connections require an external tool to be installed within the Password Vault Web Access:

- EPV RDP transparent connections from a non-Windows environment
- Through PSM from a Unix/Linux environment
- Through PSM when NLA authentication is enabled on the PSM server

Note: Previously, users who wanted to connect through PSM from Mac or non-IE browsers in Windows environments where NLA authentication was not enabled on the PSM server had to use an external tool. Now these users can connect to target machines without this tool through the built-in RDP file based solution. For more information about the RDP file based solution, refer to the table in *To Configure PSM Connections through an External Tool*, page 551.

Requirements

- **Java Virtual Machine (JVM)** – Required by the tool that enables end users to connect to non-IE browsers from their workstations. Download this tool from www.java.com.
- **HOB folder** – You will receive this folder from your CyberArk support representative.

From the installation package that you received from your CyberArk support representative, store this folder in the PVWA installation folder. By default, this folder is `c:\inetpub\wwwroot\PasswordVault`.

CyberArk License

- Your CyberArk license must allow the PVWA to integrate with this feature. For more information, contact your CyberArk support representative.

Configuring EPV Transparent Connections through an External Tool

1. Log onto the PVWA as an administrator user.
2. Configure the Connection Component in the Web Access Options:
 - i. In the System Configuration page, display the **Web Access Options**.
 - ii. Expand **Connection Components** then select **RDPApplet-Sample**.
 - iii. In the Properties list, change the unique ID that identifies this connection component from **RDPApplet-Sample** to **RDPApplet**.
The default Windows platforms are configured to support the **RDPApplet** connection component and do not require any configuration.
 - iv. Click **Apply** to apply the new Connection Component configurations,
or,
Click **OK** to save the new Connection Component configurations and return to the System Configuration page.

Configuring PSM Connections through an External Tool

Enable the external tool for PSM connections in the following scenarios:

- To connect through PSM from a non-Windows environment
- To connect through PSM when NLA authentication is enabled on the PSM server
- To avoid using ActiveX when connecting from IE browser

When this tool is enabled, it will be used for all PSM connections from non-IE browsers, even when the environment is a Windows environment.

Notes:

- Currently the external tool doesn't support connections when RD Gateway is configured in the environment.

To Configure PSM Connections through an External Tool

1. In the System Configuration page, click **Options**; the Web Access Options are displayed.
2. Click **Privileged Session Management UI**; the general Privileged Session Management UI properties are displayed.
3. In the Properties list, set the **NonIERemoteDesktopAccess** parameter to **HOB** to determine that this external tool, and not the built-in solution via an RDP file, will be used for establishing PSM RDP connections when the connection is not made with Microsoft RDP ActiveX. For more information about this parameter, refer to the Privileged Account Security Reference Guide.

For more information about the workflows you can enable with this parameter, refer to the table below.

For more information about configuring PSM connections for an intuitive user experience, refer to *Configuring the PSM Session User Experience*, page 610.

4. In the Properties list, set the **ConnectPSMWithRDPActiveX** parameter to either **ByBrowser** or **Never**, depending on whether you want Microsoft RDP ActiveX to be used to establish connections through IE or not:
 - **ByBrowser** – The external tool will be used to establish PSM connections from non-IE browsers, such as Firefox. Microsoft RDP ActiveX will be used to establish connections through IE.
 - **Never** – The external tool will be used to establish PSM connections from both IE and non-IE browsers. Microsoft RDP ActiveX will not be used to establish connections.

For more information about the workflows you can enable with this parameter, refer to the table below.

For more information about configuring PSM connections for an intuitive user experience, refer to *Configuring the PSM Session User Experience*, page 610.

5. Click **Apply** to apply the new Connection Component configurations,

or,

Click **OK** to save the new Connection Component configurations and return to the System Configuration page.

The following table lists the possible combinations of the **ConnectPSMWithRDPActiveX** and **NonIERemoteDesktopAccess** parameters and describes their requirements:

NonIERemote Desktop Access Connect PSMWith RDPActiveX	RDPFile			External Tool (HOB)		
		Connection Method	RemoteApp experience		Connection Method	RemoteApp experience
Always	IE	ActiveX	*	IE	ActiveX	*
	Non-IE	Not supported	Not supported	Non-IE	Not supported	Not supported
By Browser	IE	ActiveX	*	IE	ActiveX	*
	Non-IE	RDP file	✓	Non-IE	External tool	*
Never	IE	RDP file	✓	IE	External tool	*
	Non-IE	RDP file	✓	Non-IE	External tool	*
Requirements	<ul style="list-style-type: none"> Requires PSM 9.2 or higher and Vault/PVWA 9.2 or higher. RemoteApp user experience requires: <ul style="list-style-type: none"> PSM must be installed on Windows 2012R2. RDP client v6.1.7601 or above (RDP protocol version v7.1 or above) on end user machines. 			<ul style="list-style-type: none"> Requires HOB installation and special CyberArk license. 		

Notes:

- Connections from Unix/Linux environments are only supported with an external tool.
- Connections when NLA is enabled on the PSM server are only supported with an external tool.
- Connections with RD Gateway are only supported when connecting with ActiveX.

Customizing Connection Components

Enterprises can create custom connection components that integrate with the PVWA and enable users to connect transparently to remote machines with specific account credentials, using local programs/utilities/scripts located on users own machines (the connection component runs in the users machine internet browser context). A Java-based script that is included with the Privileged Account Security solution can be customized easily to create one or more connection components that enterprises can implement according to their own particular needs. Currently, generic connection components can only be customized for direct transparent connections, and cannot be configured for the PSM.

When a user tries to connect to a remote machine using an account that is associated with a customized connection component, a generic connection component scripting interface provides the details that are required to create a transparent connection through the customized script. All the account properties can be specified as well as details about the remote machines. In addition, extra accounts can be included for verification, reconciliation, or additional logon, if required. The entire process is transparent to the user who benefits from a continuous workflow and immediate connectivity.

The sample generic connection component included with the Privileged Account Security solution is called PuTTY. The Java script file is stored in the ConnectionScripts subfolder of the PVWAConfig Safe during installation. In addition, the corresponding connection component in the PVWA is defined in the Connection Components in Web Access Options. In order for the PuTTY sample to function properly, the PuTTY utility needs to be installed in the Program Files directory on each user machine, using this connection component.

Multiple custom connection components can be configured for each platform. When several connection components are available for an account, they are displayed in a drop-down list, from which users select the specific connection component to use when they connect transparently. For more information about configuring multiple connection components, refer to *Configuring Multiple Target Addresses*, page 560.

Defining Customized Connection Components

CyberArk's generic connection component scripting interface enables you to create custom connection components according to your enterprise's unique needs, using specific account credentials.

When a user tries to connect to a remote machine using the customized connection component, a file that contains an HTML script element is used. This script is written in javascript and uses a javascript scripting interface to run a specific program/utility/script.

- Create the file that contains an HTML script element using the function and methods listed below.

Customized Connection Components scripting interface

This javascript scripting interface includes the functions methods and properties listed below.

Note: All methods and properties are case-sensitive and must be specified exactly as they appear below.

Function:

The following function runs the customized script with the password stored in the account that is used to log onto the remote machine.

```
runCustomScript (Password)
```

Note: This function is case-sensitive and must be specified exactly as shown above.

Methods:

DebugMsg

This method accepts a string and activates an alert with this string when the debugging mode is configured.

HandleError

This method retrieves a string and displays it after an error has occurred, and then closes the transparent connection window.

GetProperty

This method accesses the information required to create a transparent connection and log onto a remote machine. It is used to retrieve internal properties, as shown in the following example:

```
var username = GetProperty('GenericUserName');
```

Some of the properties are actually a list of inner properties that are used to retrieve the value of an inner property, as shown below:

```
var AccountProperties = GetProperty("AccountProperties");
var LoggenOnUserName = AccountProperties["LoggedOnUserName"];
```

Some of the inner properties value can only be retrieved by accessing their "Value" key as seen in the shown example:

```
var properties = GetProperty("GenericParameters");
var MyAppPath = properties["ExePath"].Value;
```

The following table displays the internal properties that can be retrieved with the **GetProperty** method:

Properties	Description
GenericUserName	The user name specified in the account that will be used to log onto the remote machine.
GenericAddress	The address of the remote machine where the account will be used.
Safe	The name of the Safe where the account is stored.
Folder	The name of the folder where the account is stored.

Properties	Description
AccountName	The name of the account that will be used to log onto the remote machine.
LogonDomain	The name of the domain where the account will be used. If this property is not specified in the account, the value of this property is set to 'undefined'.
GenericParameters	<p>Component parameters that are defined in the Web Access Options and can be overridden at platform level. In addition to the properties that are defined in the PVWA, all the properties specified above can be specified, as well as the following parameters:</p> <ul style="list-style-type: none"> ▪ ConnectionScriptName - The name of the customized connection component script that defines the connection to the remote device. For more information, refer to <i>Configure a New Generic Connection Component</i>, page 558. ▪ DisplayDebuggingMessage - Whether or not debug messages are displayed. By default, it is not displayed, but it can be configured at connection component level and overwritten at platform level. For more information, refer to <i>Debugging the Connection Component</i>, page 560. <p>These parameters are accessed with the 'Value' attribute. For more information about component parameters, refer to <i>Configuring Transparent Connections</i>, page 527.</p>
UserParameters	<p>User parameters that are defined in the Web Access Options and can be overridden at platform level, including the following parameter:</p> <ul style="list-style-type: none"> ▪ RemoteMachine - The name of the remote device to connect to. <p>These parameters are accessed with the 'Value' attribute. For more information about user parameters, refer to <i>Configuring Transparent Connections</i>, page 527.</p>
AccountProperties	<p>A list that contains the following account properties:</p> <ul style="list-style-type: none"> ▪ AccountProperties – A list that contains all the properties of the account being used. ▪ LoggedOnUserName – The name of the PVWA user who is currently logged on. ▪ WindowsUserNa user name of the current Windows user. This property is relevant for Windows authentication.
ExtraPassword<i>	<p>Details about an extra password. Up to three extra passwords can be specified.</p> <p>These parameters are accessed with the 'Value' attribute. For more information about specifying extra passwords, refer to <i>Retrieving Extra Passwords</i>, page 556.</p>

Retrieving Extra Passwords

You can retrieve extra passwords for account reconciliation and verification, and additional logon through your custom connection component script. The type of extra password is denoted by the name of the property as shown below:

- **ExtraPassword1** – Usually represents an additional logon account.
- **ExtraPassword2** – Usually represents a verification account.
- **ExtraPassword3** – Usually represents a reconciliation account.

This property is a list of the following subproperties:

- **GenericUserName** - The user name specified in the linked account.
- **GenericAddress** - The address of the linked account.
- **AccountProperties** - A list that contains all the file categories of the linked account.
- **Safe** – The name of the Safe where the linked account is stored.
- **Folder** – The name of the folder where the account is stored.
- **AccountName** – The name of the linked account.
- **Password** – The password stored in the linked account. If the user requires confirmation in order to use this password, this confirmation must be received before the user can connect to the remote machine. The default reason provided with this property is “Getting an extra password as part of a generic transparent connection.” For more information about dual control, refer to refer to *Dual Control*, page 251.

Specify these parameters in your script by using the Value attribute, as shown in the following example:

```
var reconcileAccount = GetProperty("ExtraPassword3");
var reconcilePassword = reconcileAccount["Password"].Value;
```

For more information about extra passwords, refer to *Linked Accounts*, page 220.

Retrieving a Specific File Category of an Account

Some of the common file categories are exposed as properties, such as "UserName" and "Address" (as "GenericAddress" and "GenericUserName").

If there is a need for a specific file category that is not exposed as a property, it can be retrieved from "AccountProperties". The following example shows how to retrieve the value contained in the "DevceType" file category:

```
var accountProps = GetProperty("AccountProperties");
var innerAccountProps = accountProps["AccountProperties"];
var deviceType = innerAccountProps.DeviceType.Value;
```

Retrieving a Specific File Category of an Extra Password

The value of a file category of an extra password can be retrieved as shown below. The following example shows how to retrieve the value contained in the "Client" file category of the reconcile account (ExtraPassword3) of the account:

```
var accountProps = GetProperty("AccountProperties");
var reconcileAccount = accountProps["ExtraPassword3"];
var recocilePassword = reconcileAccount["Password"];
```

Example

The following example explains how to use the CyberArk's generic connection component scripting interface. The following script is in a file called **PuTTY.js** that is stored in the PVWAConfig Safe in the ConnectionScripts folder, and can be used to connect to a remote machine using the PuTTY application.

The following explanation describes the above script:

Step 1 – The script calls the runCustomScript function, which runs the customized script with the password stored in the account that is used to log onto the remote machine.

Step 2 – An ActiveX object is created in order to be able to excute local applications.

Step 3 – The generic connection component scripting interface retrieves the user name and address of the account, as well as the path of the PuTTY executable file. Each time the scripting interface retrieves information, a debug message displays it, ensuring that it is retrieved and that the transparent connection can be created successfully.

Step 4 – This is the command line string with all the parameters. In this example, the command includes the path of the PuTTY executable file, the user name, the password that is stored in the account being used to access the remote machine, and the address of the machine where the user initiated a transparent connection.

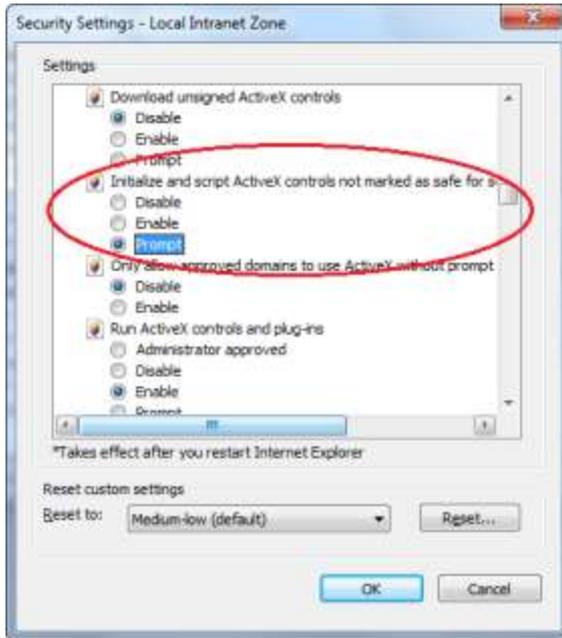
Step 5 – The script runs the command that creates the transparent connection to the remote machine.

Step 6 - After the program is executed, the window that runs the script is closed.

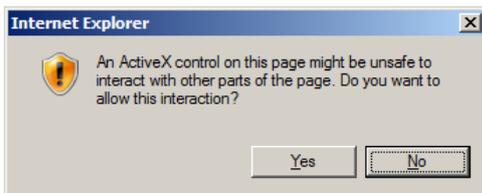
Enable Script Execution on each Client Machine

On each client machine, change the default security settings for Internet Explorer to enable the customized connection components use an ActiveX executable called **WScript.Shell**.

1. In Internet Explorer, from the **Tools** menu, display the **Internet Options**.
2. Display the **Security** tab, then click Custom Level; the Security Settings window for the Local Intranet/Trusted Sites Zone is displayed.
3. Scroll down to ActiveX controls and plug-ins, and display the Initialize and script ActiveX controls not marked as safe for scripting settings.
4. Set this setting to **Prompt**, then click **OK**.



Each time customized connection components are activated, the following message will appear:



Users must allow the script to run on their machines in order to connect to remote machines.

Configure a New Generic Connection Component

1. Define the customized connection component:
 - i. Log onto the PrivateArk Administrative Client as a Vault administrator.
 - ii. From the PWWAConfig Safe, from the ConnectionScripts folder, retrieve the default connection component script, **PuTTY.js.**, and save it with a different name. Use a name that indicates the type of connection component that it will define.
 - iii. Modify the java script file to define your customized connection component. For more information about creating customized scripts, refer to *Defining Customized Connection Components*, page 553.
 - iv. Save the customized connection component script file in the ConnectionScripts folder in the PWWAConfig Safe.
2. In the PWWA, add a new connection component:
 - i. Log onto the PWWA as a Vault administrator.
 - ii. In the System Configuration page, display the Web Access Options, then display **Connection Components**.
 - iii. Right-click on the PuTTY connection component and select **Copy**, then right-click on Connection Components and select **Paste Connection Component**; the connection component is added to the bottom of the list of connection components.

- iv. Select the new connection component and change the following properties:
 - **Id** – The unique ID that identifies the connection parameters. The ID that you specify here will be displayed in the list of Connections Components.
 - **DisplayName** – The display name of the connection component. This is only relevant if the display name differs from the ID.
 - v. In the **Type** property, specify the following value:
CyberArk.PasswordVault.TransparentConnection.Generic.GenericConnectionComponent, CyberArk.PasswordVault.TransparentConnection.Generic
 - vi. Expand the Connection Component, and display the Component Parameters.
 - vii. Select **ConnectionScriptName** then, in the Properties list, in the value property, specify the name of the new Java script.
 - viii. Create new component parameters or delete or modify existing component parameters that are relevant for the new script.
 - ix. If necessary, add user parameters that are relevant for the new script. Right-click the name of the Connection Component, select **Add User Parameters**, and add new parameters and values. For example, RemoteMachine.
 - x. Click **Apply** to save the new Connection Component configurations and activate the connection component immediately,
or,
Click **Save** to save the new Connection Component configurations and apply them after the period of time specified in the **RefreshPeriod** parameter.
3. Configure the relevant platforms for the new connection component script:
 - i. Click **ADMINISTRATION** to display the **System Configuration** page, then click Platform Management to display a list of supported target account platforms.
 - ii. Select the platform to configure, then click **Edit**; the settings page for the selected platform appears.
 - iii. Right-click **Connection Components**, then select **Add Connection Component**; a new connection component is added to the list.
 - iv. Select the new connection component and specify the following properties:
 - **ID** – Specify the unique ID of the new connection component. This is the unique ID that you specified when you configured the Connection Component in Web Access Options in the previous step.
 - **Enabled** – Set the Enabled property to **Yes** to enable the connection component.
 - v. Click **Apply** to apply the new platform configurations immediately,
or,
Click **OK** to save the new Connection Component configurations and return to the System Configuration page.

Debugging the Connection Component

The sample customized connection component includes a debugging feature that displays the exact command that is run and all the parameters that are used with it. This feature, activated by the **DisplayDebuggingMessage** parameter, displays retrieved information as a message after the user clicks Connect in the PVWA. This parameter can be configured at connection component level and overridden at platform level. By default, the **DisplayDebuggingMessage** parameter is not activated.

The number of debug messages that are displayed depends on how many times the **DebugMsg** method is specified in the script. For more information about this method, refer to *Customized Connection Components scripting interface*, page 554.

To Activate Debugging

1. Log onto the PVWA as a Vault administrator.
2. In the System Configuration page, display the Web Access Options, then display **Connection Components**.
3. Expand the connection component to configure, then display the Component Parameters.
4. Select the **DisplayDebuggingMessage** parameter and set it to **Yes**.
5. Click **Apply** to apply the new configurations immediately,

or,

Click **OK** to save the new configurations and return to the System Configuration page.

Debugging has been activated and messages will be displayed each time this connection component is used to log onto a remote machine.

Configuring Multiple Target Addresses

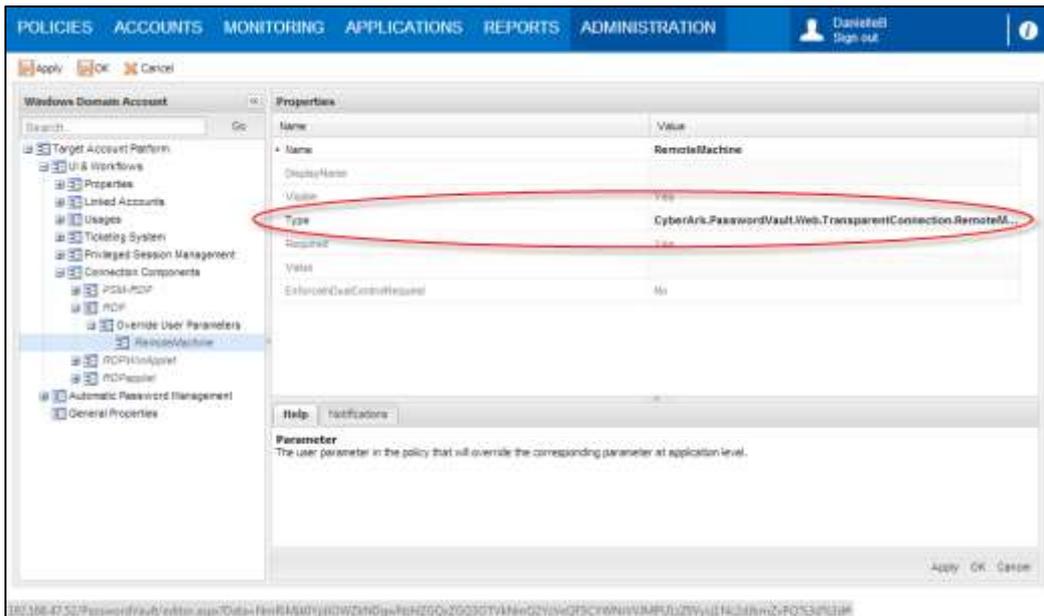
The PVWA can be configured to display multiple target addresses for users to select from when they create a request or connect to a remote machine transparently.

Note: This can be configured for Windows and Unix accounts.

1. Click **ADMINISTRATION** to display the **System Configuration** page, then click Platform Management to display a list of supported target account platforms.
2. Select the platform that will manage accounts used to access the remote machines that will be displayed in the multiple targets list, then click **Edit**; the settings page for the selected platform appears.
3. Display the **Connection Components**, and then expand the parameters in the component to configure.
4. Display the **Override User Parameters**.
5. In the **RemoteMachine** parameters specify the following text in the **Type** property:

```
CyberArk.PasswordVault.Web.TransparentConnection.RemoteMachineUser
Parameter, CyberArk.PasswordVault.Web
```

The following example shows the configuration for the RemoteMachine parameter in the Windows Domain Account platform.



6. Click **OK** to save the changes and return to the main System Configuration page.
7. Click **Options**, then select **Connection Components**; the connection component parameters that define multiple target addresses are displayed in the properties list.
8. Define the following parameters:
 - **EnableConnectAddressHistory** – Determines whether or not a list of addresses accessed with the selected account will be displayed in the Connect with Account window. The default value is **Yes**.
 - **MaxConnectHistory** – Defines the maximum number of remote machine addresses that can be displayed in the Connect with Account window. The address history is saved per account for each PVWA user. The default value is **7** addresses.
 - **MaxConnectAccountsNumber** – Defines the maximum number of accounts whose machine addresses history will be displayed in the Connect with Account window. The default value is **20** accounts.
9. In **Privileged Account Request**, define the following parameters:
 - **AddressSeparatorCharacter** – Defines the separator between addresses for remote connections.
 - **AnyAddressCharacter** – Defines the character that will represent "all addresses" in dual control requests.
10. Click **Apply** to apply the new Connection Component configurations, or,

Click **OK** to save the new Connection Component configurations and return to the System Configuration page.

Accessing PVWA from another Web Application

The following parameters, in the **Access Restriction** section of the Web Access Options, defines a URL in other web applications through which the PVWA can be accessed. These web applications must be installed on the same virtual directory. Access from applications that are installed on a different URL will be denied.

By default, an Allowed Referrer called /WebID/ is created with a BaseURL property. Change the value of this property or create one or more new Allowed Referrers.

To create a new Allowed Referrer

1. Right click on **Access Restriction**, then select **Add AllowedReferrer**; a new Allowed Referrer is created.
2. In the properties list, specify the URL from where users will be able to access PVWA. You can specify either of the following:
 - A portal URL which allows access from any page or subdirectory under the CompanyA/portal URL. For example, <https://CompanyA/portal/>.
 - The exact name of a URL which only allows access from the specified page. For example, <https://CompanyB/management/dashboard.php>.

Reports

The **Reports** section of the Web Access Options defines reports administration in the PVWA, and includes the following management features:

- **Report Scheduling UI**– These parameters define the configurations and default values for scheduling reports.
- **Report Definitions List** – These parameters define the **Report Definitions** tab in the **My Reports** page.
- **Report Grid Actions** – These parameters define the report actions that will be available from the reports list.
- **Report Display Columns** – These parameters define the properties that will appear in the reports list. The order of the properties determines how this information will be displayed.
- **Report Scheduling** – These parameters define the configurations for reports scheduling.
- **Report Collections** – These parameters define a list of assemblies that implement one or more reports. By default, all the reports contained in the assembly can be generated in the My Reports page. Customized reports can be configured by adding a Report node. Contact your CyberArk representative for further information.
- **Report Style** – These parameters define the style of exported reports, if the export format supports styling.

General Report Settings

The following parameters, in the **Reports** section of the Web Access Options, define general reports management in the PVWA.

- The **DefaultSafe** parameter specifies the name of the Safe that will be created for reports and where generated reports will be saved. The default Safe is **PVWAReports**. For more information about the Reports Safe, refer to *The Password Vault Web Access Built-in Environment* in the Privileged Account Security Installation Guide.
- The **ExecutionTimeout** parameter specifies the maximum time in minutes that a report can be generated. After this time has expired, a timeout message will be displayed. The default value is **60** minutes.
- The **ManageReportsGroup** parameter specifies the group that users who are authorized to generate reports must belong to. The default group is **PVWAMonitor**.
- The **RecordsFlushInterval** parameter specifies the number of records that will be written to the report output file in every internal interval. The default value is **100**.
- The **MaximumRecords** parameter specifies the maximum number of records that will be included in a report. The default value is **40,000**.
- The **MaxConcurrentReports** parameter specifies the maximum number of reports that can run concurrently. The default value is **1**.

The following parameters determine how the report output will be saved in a CSV file:

- The **CSVSeparator** parameter specifies the separator that will be used between the different columns in the CSV file. The default value is **,** (comma).
- The **CSVTextQualifier** parameter specifies the enclosing text of each column in the CSV file. The default value is **"** (quotations).
- The **DisplayFieldsHeader** parameter determines whether or not the header columns in the CSV file will be displayed in the report. The default value is **Yes**, which indicates that header columns will be displayed in the report.

Report Collections

The following parameters define each assembly that implements one or more reports:

- The **AssemblyName** parameter defines the name of an assembly.
- The **Name** parameter defines the name of the report collection.
- The **Parameters Lists** parameters specify a list of internal parameters that are used to generate reports. These parameters can be defined for each collection of reports and then overridden in the parameters lists for each specific report.

- The **Output Fields** parameters specify the list of fields that will be displayed in reports. These parameters can be defined for each collection of reports and then overridden in the output fields for each specific report.
 - **Privileged Accounts Inventory:**
 By default, the following fields are displayed: Safe, Folder, Name, Platform Id, DeviceType, Username, Address, Group, LastAccessedDate, LastAccessedBy, LastModifiedDate, LastModifiedBy, VerificationDate, CheckoutDate, CheckedOutBy, Age, ChangeFailure, VerificationFailure, MasterPassFolder, MasterPassName, DisabledBy, DisabledReason.
 The following fields can be added manually: CheckoutDate, CheckedOutBy, Age, ChangeFailure, VerificationFailure, MasterPassFolder, MasterPassName, DisabledBy, DisabledReason, any account property that has been specified for one or more accounts.
 - **Privileged Accounts Compliance Status:**
 By default, the following fields are displayed: Username, Address, Safe, Platform ID, ComplianceStatus, NonComplianceReason, ExpirationDue, PlannedPasswordChange, ChangeMode, OneTimePasswords, ExpirationPeriod, LastModifiedDate, LastAccessedBy, LastAccessedDate, LastAccessRequestTimeframe.
 The following fields can be added manually: ExclusiveAccess, DualControl, PerformPeriodicChange, AllowManualChange, HeadStart, AccessValidityPeriod, PasswordLength, PasswordComplexityPolicy, PasswordChangeReason, Disabled, DisabledReason, ChangeFailure, FailReason, PasswordAge, ExpirationDate, CreationDate, Deleted, LastModifiedBy, VerificationDate, Folder, Name, DeviceType, any account property that has been specified for one or more accounts.
 - **Entitlement report:**
 By default, the following fields are displayed: User, FullName, Group, GroupOwnership, Location, UserType, TargetPolicy, TargetSystem, TargetAccount, Safe, Read, Change, OtherPermissions
 - **Applications report:**
 By default, the following fields are displayed: ApplicationID, BusinessOwner, Location, AllowedMachines, OSUser, Path.
 The following fields can be added: ApplicationDescription, BusinessOwnerEmail, BusinessOwnerPhone, Disabled, Hash, AccessPermitted, ExpirationDate
 - **Activities log report:**
 By default, the following fields are displayed: Time, User, Action, Safe, Target (account), Target Platform, TargetAccount, NewTarget, Reason, Alert, RequestID, ClientID
- The **Reports** parameters define the list of available reports and their configuration parameters. The specified report parameter determines the information that defines the report.

Report Display Columns

The following parameters define the properties that will appear in the reports list:

- The **SortBy** parameter specifies the default column by which to sort the grid.
- Each parameter that is defined in this section specifies the title of the column, the width of the column in the grid, and the data type of the information. You can also specify whether or not the parameter will be included in the grid.

Report Grid Actions

The following parameters define all the report actions that can be displayed in the reports list to enable users to manage generated and scheduled reports:

- Each parameter that is defined in this section represents an action that can be performed in the reports list. The properties define the name of the action, and whether or not it will be displayed in the reports list and/or the action menu.

Report Definition List Settings

These parameters define how generated reports are displayed in the **Reports** tab in the **My Reports** page, and the activities that can be performed in this tab.

Report Display Columns

The following parameters define the properties that will appear in the reports list:

- The **SortBy** parameter specifies the default column by which to sort the grid.

Each parameter that is defined in this section specifies the title of the column, the width of the column in the grid, and the data type of the information. You can also specify whether or not the parameter will be included in the grid.

Report Grid Actions

The following parameters define the report actions that will be available from the reports list:

- Each parameter that is defined in this section represents an action that authorized users can performed in the reports list. The properties define the name of the action, and whether or not it will be displayed in the reports list and/or the action menu.

Report Scheduling Options

The following parameters define the general configurations and default values for scheduled reports:

- In the main Reports section, the **DefinitionsSafe** parameter specifies the name of the Safe where definitions for scheduled reports are saved. The default Safe is **PVWATaskDefinitions**.

Other default values can be configured in the **Reports Scheduling UI** parameters:

- The **GenerateReportRecurrence** parameter defines the default type of scheduling recurrence. Possible values are described in the table below:

Value	Description
Generate	Generate the report. This is the default value.
Generate and save report definitions	Generate the report and save the report definitions to use again.
Schedule	Schedule the report to run according to the following parameters.

- The **WeeklyRecurrence** parameter determines how many times the report will be generated each week. The default value is **1**.
- The **MonthlyRecurrence** parameter determines how many times the report will be generated each month. The default value is **1**.
- The **ScheduleStartTime** parameter defines the default time when scheduled reports will be generated. The time, specified using the **HH:MM** format, changes according to the user's time zone.
- The **FailureNotificationToMe** parameter determines whether or not a notification will be sent to the subscriber who scheduled the report if it cannot be generated. The default value is **No**.
- The **SuccessNotificationDefault** parameter determines whether or not a notification will be sent to subscribers after a report has been generated successfully.
- The **ScheduleTypeDefault** parameter defines whether the default scheduling type is Weekly or Monthly. The default value is **Weekly**.
- The **Search Users Display Columns** parameters define the columns that will be displayed in the Search for Subscribers window.
- The **SortBy** parameter specifies the default column by which to sort the grid.

Each parameter that is defined in this section specifies the name of the file category that will be displayed as a column, the display name of the column, the width of the column in the grid, and the data type of the information. You can also specify whether or not the column will be included in the grid.

Report Scheduling Settings

- The **DefinitionsSafe** parameter specifies the name of the Safe where definitions for scheduled reports are saved. The default Safe is **PVWATaskDefinitions**.
- The **MaximumTimeout** parameter specifies the maximum number of days to wait between report generations.

Report Generation Failure Notification Groups

The following parameters define the groups that will receive notifications when a report generation fails:

- The **FailureNotificationGroups** specifies the groups that receive notification when a report generation fails. By default, the Vault Admins group is specified.

Report Styles

The following parameters define the style of exported reports, if the export format supports styling, using a CSS-style RGB value between #000000 and #FFFFFF.

- The **ColumnTitleBackgroundColor** parameter defines the background color of column headers. The default value is #919aab.
- The **ColumnTitleTextColor** parameter defines the text color of column headers. The default value is #ffffff.
- The **OddGroupTitleBackgroundColor** parameter defines the background color of odd group headers. The default value is #f8b277.
- The **EvenGroupTitleBackgroundColor** parameter defines the background color of even group headers. The default value is #dce3ed.
- The **GroupTitleTextColor** parameter defines the text color of group headers. The default value is #1c2329.
- The **OddReportRecordBackgroundColor** parameter defines the background color of odd report records. The default value is #eeede8.
- The **EvenReportRecordBackgroundColor** parameter defines the background color of even report records. The default value is #f7f6f4.
- The **ReportRecordTextColor** parameter defines the text color of report records. The default value is #f7f6f4.

Logging

The following parameters, in the **Logging** section of the Web Access Options, define log files management for the PWWA.

- The **DebugLevel** parameter specifies the level of debug messages that will be logged. In this way, you can control the amount of debug messages written to the log files. Valid values are None, High, Low, and Profiling.

Note: **High** and **Profiling** create a large volume of logs and should only be enabled for troubleshooting.

- The **InformationLevel** parameter controls the amount of informational messages written to the log files. Valid values are None, High, and Low.
- The **LogFilesRetentionPeriod** parameter specifies the number of days for which all the log files will be stored. After the specified number of days, the log files will be deleted.

Caching

The following parameters, in the **Caching** section of the Web Access Options, define caching preferences.

Caching enables the Password Vault Web Access to store information from the Password Vault and display information from that store when requests are received. The cache is refreshed at predetermined intervals. This is recommended in large scale implementations to improve Password Vault Web Access performance.

- The **EnableCache** parameter determines whether or not the Password Vault Web Access will cache Vault information and display the contents of the cache.

Configuring the Account Retrieval Form

The Account Retrieval Form enables you to specify information that is required before you can access accounts that meet the following requirements:

- A reason for accessing the account must be specified.
- A ticketing system is integrated with the PWWA and users must specify the ticketing system information.
- Authorized users must confirm users' requests to access accounts.
- Connection details are required in order to log onto a remote device transparently with Privileged SSO.

The following parameters, in the **Privileged Account Request** section of the Web Access Options, define the account retrieval form.

- **Reason** – The following parameter configures the Reason pane:
 - The **ReasonFieldCaption** parameter specifies the customized text for the access reason prompt. The Reason pane will only be displayed in the Safe is configured for 'Require access reason' or if dual control is configured.

- **Ticketing system integration** – The following parameters configure the Ticketing system pane. This pane will only be displayed if the platform associated with the account requires users to specify ticketing information.
 - The **TicketingSystemFieldCaption** parameter specifies the customized text for the ticketing system prompt.
 - The **TicketingIDFieldCaption** parameter specifies the customized text for the ticketing ID prompt.

For more information about configuring parameters for integrating ticketing systems, refer to *Integrating with Ticketing Systems*, page 516.

- **Requests** – For information about configuring parameters that define dual control, refer to *Dual Control*, page 513.
- **Transparent Access** – For information about configuring parameters that define transparent access, refer to *Configuring Transparent Connections*, page 527.

Configuring Split Password Mode

The following platform parameters enable Split Password mode and determine which users will be able to see the first half of a password and which users will be able to see the second half.

To Configure Split Password Mode

1. In the PrivateArk Client, create the following user groups in the Vault:
 - A group for users who will be able to retrieve the **first** half of the passwords to view in split password mode. For example, Password First Half.
 - A group for users who will be able to retrieve the **second** half of the passwords to view in split password mode. For example, Password SecondHalf.
2. Add users to the groups as members, depending on the half of the password that they will be authorized to see.
3. In the PVWA, click **ADMINISTRATION** to display the **System Configuration** page, then click Platform Management to display a list of supported target account platforms.
4. Select the platform to configure for split password mode, then click **Edit**; the settings page for the selected platform appears.
5. In the general parameters for this platform, specify the following parameters:
 - **EnableSplitPassword** – Set this parameter to **Yes**. This enables the Split Password mode for this platform.
 - **PasswordFirstHalfGroup** – Specify the name of the group whose members will be able to view the first half of passwords assigned to this platform.
 - **PasswordSecondHalfGroup** – Specify the name of the group whose members will be able to view the second half of passwords assigned to this platform.

Users who belong to both groups can see the entire password. Users who do not belong to either group will not be able to see any passwords.

Adding Accounts

The following parameters, in the platform's UI & Workflows parameters, determine whether or not password management activities will be performed as soon as an account assigned to this platform is added.

Passwords can be either changed or verified as soon as they are added to the Password Vault, as follows:

- **Changing passwords when they are added** – A password change process can be initiated as soon as accounts marked with the **AutoChangeOnAdd** parameter are added to the EPV. The password is automatically generated by the CPM and is immediately synchronized with the corresponding password on the remote machine.
- **Verifying passwords when they are added** – Password verification processes can be initiated as soon as accounts marked with the **AutoVerifyOnAdd** parameter are added to the EPV. This ensures that new passwords in the Vault are immediately synchronized with the corresponding password on the remote machine.

Adding Safes

Authorized users can add Safes in the Vault through the PVWA. These Safes are created according to predefined parameters in the Web Access Safe Templates parameters. This template enables Vault administrators to maintain a strict structure for all the Safes that are created through the PVWA.

In order to add Safes, users require the following authorization in the Vault:

- Add Safes

All Safes that are added in the PVWA are created in the same location as the user's account in the Vault hierarchy.

For more information about the Safe template used to create new Safes, refer to the Privileged Account Security Reference Guide.

Configuring the PVWA Interface

Different features of the PVWA interface can be configured in the web.config file that is in the Inetpub\wwwroot>PasswordVault folder on the IIS.

Logo

The Password Vault Web Access interface can be customized, so that your company's logo is displayed on the top right side of the screen.

- In the PVWA installation folder, in the images subfolder, replace the **login_CustomLogo.png** file with your own logo, using the same file name.

The optimum size of the logo is 300 pixels wide and 60 pixels in height. Logos that are larger than the specified size will be resized automatically.

Multiple Languages

The interface language is determined according to the language preference in the internet browser which can be modified by the end-user. Usually, these settings are configured automatically during Windows installation. Currently, the PVWA supports the following languages:

- English
 - French
 - Spanish
 - German
 - Russian
 - Japanese
 - Korean
 - Simplified Chinese
 - Traditional Chinese
 - Brazilian Portuguese
- The **MultiLingualSupport** parameter in the web.config configuration file enables the PVWA to display the interface in the language configured in the internet browser. By default, the PVWA interface is English.

To Install the PVWA Interface for Languages other than English

1. Copy the **relevant language** folder from the CyberArk installation package to the 'bin' subfolder of the 'PasswordVault' folder.
2. In the web.config configuration file on the IIS server, set the **MultiLingualSupport** parameter to **Yes**, as shown in the following example.

```
<appSettings>
.
.
.
  <add key="MultiLingualSupport" value="Yes" />
</appSettings>
```

3. Save the web.config file and close it.

Embedding PVWA in an IFrame

PVWA can be embedded in an iFrame element using the following parameter:

- The **AllowFraming** parameter in the General PVWA system configuration parameters determines whether or not PVWA can be embedded in an IFrame. The default value is **Yes**.

Configuring the Mobile Password Vault Web Access

The Mobile PVWA can be configured to provide several functions that enable users to access critical accounts from mobile devices. All configurations can be specified by authorized users in the System Configuration page. For more information, refer to *Configuring the System through PVWA*, page 918.

Configuring Authentication to the Mobile PVWA

Users can authenticate to the Password Vault through the Password Vault Web Access using any of the following authentication methods:

- Cyberark
- Radius
- RSA SecurID
- LDAP

The default authentication method can be specified during installation, after which it is configured in the Web Access Options in the Authentication Methods section. You can change the default authentication method after installation with the following parameters in the GeneralSettings section of the Authentication Methods:

- **MobileDefaultMethod** – The identifier of the default authentication method for the mobile PVWA. This also determines the direct URL through which the end-user can access the mobile PVWA authentication page.

General Configurations

The following parameter, in the **General** parameters, defines general Mobile PVWA application settings.

- The **AuthenticationSourceAddress** parameter determines whether the source address used to authenticate users to the Vault will be taken from the PVWA's server or the end users' mobile IP address. This affects network areas authentication. The default value is **Client Address**, indicating the end user's PVWA address is used to authenticate to the Vault.

Configuring Mobile Devices

The **Supported Devices** parameters define the supported mobile devices that will automatically be redirected to the mobile version of the PVWA. Each supported device must be defined.

Displaying Application Links

The **Application Links** parameters define the application links that appear at the bottom of the Mobile PVWA screen.

Displaying Accounts Search Results

The following parameters in the **Accounts Search Results** section define the Search Accounts page in the Mobile PVWA:

- The **Search Options** parameters define the accounts search and how the results will be displayed.
 - The **ExecutionMaxDuration** parameter defines the maximum number of seconds that a search process will last.
 - The **MaxDisplayedRecords** defines the maximum number of accounts that will be displayed in the Search Results page for accounts. The default value is **500** records.
 - The **PageSize** defines the maximum number of accounts to display in the grid on each page. The default value is **18** accounts.
 - The **AutoOpenObjectDetails** determines whether the Account Details page for an account will be displayed if a search results in that single account. The default value is **No**.
- The **Displayed Properties** parameters define the account properties that will be displayed in the search results.

Displaying Account Details

The following parameters in the **Account Details** section define the Account Details page in the Mobile PVWA:

- The **Toolbar Actions** parameters define the buttons that will be displayed on the toolbar.
- The **Account Properties** parameters define the account properties that will be displayed in the Account Details page for each account.

Displaying Request Details

The following parameters in the **Request Details** section define the Request Details page in the Mobile PVWA:

- The **Request Properties** parameters define the request properties that will be displayed.
- The **Account Properties** parameters define the account properties that will be displayed in the Request Details page.

Defining Password Details

The following parameters in the **Password Details** section define password related parameters:

- The **ReasonFieldCaption** property enables you to specify an alternative caption for the reason field to replace the default caption.
- The **TicketingSystemFieldCaption** property enables you to specify an alternative caption for the ticketing system field to replace the default caption.
- The **TicketIdFieldCaption** property enables you to specify an alternative caption for the ticket id field to replace the default caption.

- The **PasswordRevealingTime** property defines the number of seconds that the password will be displayed on the Mobile PVWA screen. The default value is **30** seconds.
- The **TimeframeDateFormat** property defines the date format that will be displayed in the access timeframe section. The default date format is **MM/DD/YY**.
- The **PasswordPhoneticDisplay** parameters define the phonetic display that will be displayed for passwords. Each character is configured with a letter and a representative word.

Configuring the PVWA Mode

The PVWA mode determines how users will access the PVWA. You can configure the PVWA so that users can only access either the full version of the PVWA or the mobile version, or so that users can access both versions.

Managing the Full Version of the PVWA

The full version of the PVWA can be configured to determine whether or not users will be able to access the full version of the PVWA.

The following parameter in the **web.config** file manages the full version of the PVWA:

- The **FullVersionEnabled** parameter determines whether or not users will be able to access the PVWA through the full version of the PVWA. By default, this parameter is set to **Yes**.

To Disable the Full Version of the PVWA

- In the web.config file, set the **FullVersionEnabled** parameter to **No**. Alternatively, leave this parameter empty.

To Reenable the Full Version of the PVWA

- In the web.config file, set the **FullVersionEnabled** parameter to **Yes**.

Managing the Mobile PVWA

The Mobile PVWA can be configured to determine whether or not users can access it. This also determines whether or not the link in the full PVWA to the Mobile PVWA is displayed.

The following parameter in the web.config file manages the Mobile PVWA:

- The **MobileVersionEnabled** parameter determines whether or not users will be able to access the PVWA through the Mobile PVWA. By default, this parameter is set to **Yes**.

To Disable the Mobile PVWA

- In the web.config file, set the **MobileVersionEnabled** parameter to **No**. Alternatively, leave this parameter empty.

To Reenable the Mobile PVWA

- In the web.config file, set the **MobileVersionEnabled** parameter to **Yes**.

Direct Access to PVWA Pages

Users can access PVWA pages through direct URLs. These links can be sent in emails to authorized users to enable them to access specific pages quickly. Users are required to authenticate to the PVWA, after which the specific page is displayed.

Note: The direct URL used to access PVWA pages is case-sensitive. The first time a URL is used to access the PVWA, it is adjusted automatically to the correct case. Subsequently, users must alter the URL manually.

Accessing the Account Details Page

Users can access Account Details pages through a direct URL. This circumvents the need to search for a specific account and enables authorized users to access an account quickly.

To Access the Account Details Page

- Use the following URL, which specifies the name of the Safe and folder where the account is stored, as well as the name of the account itself:

```
http://<hostname>/PasswordVault/directaccess.aspx?objectdetails.aspx&safe=
<Safe name>&folder=<folder name>&object=<password name>
```

Accessing Accounts Search Results

Users can access Account pages that display the results of a search, enabling them to view specific accounts immediately.

To Access the Accounts Search Page

- Use the following URL, which specifies the search criteria. Separate multiple keywords for the search by spaces.

```
http://<hostname>/PasswordVault/directaccess.aspx?myaccounts.aspx&searchp
attern=[search term]
```

For example, to display the results of a search for all Unix root accounts, use the following URL:

```
http://<hostname>/PasswordVault/directaccess.aspx?myaccounts.aspx&searchpatter
n=unix root
```

To Access the Accounts Search Page for a Specific Safe

- Use the following URL, which specifies the name of the Safe in which the search will be carried out.

For example, to display the results of a search for all accounts in the Finance Safe, use the following URL:

```
http://<hostname>/PasswordVault/directaccess.aspx?myaccounts.aspx&SearchInSaf
e=Finance
```

To Access the Accounts Search Page for Specific Criteria in a Safe

- Use the following URL, which specifies the name of the Safe and the search criteria. Separate multiple keywords for the search by spaces.

`http://<hostname>/PasswordVault/directaccess.aspx?myaccounts.aspx&SearchInSafe=[Safe name]&searchpattern=[search term]`

For example, to display the results of a search for all Unix root accounts in the Finance Safe, use the following URL:

`http://<hostname>/PasswordVault/directaccess.aspx?myaccounts.aspx&SearchInSafe=Finance&searchpattern=unix root`

Accessing Request Pages

Users can access Request pages through a direct URL. These links can be sent in emails to authorized users to indicate that other users are waiting for confirmation so that they can access accounts, or to users who are waiting for their request to be confirmed.

To Access the My Requests Page

- Use the following URL which specifies the 'My Requests' page:
`http://<host name>/PasswordVault/directaccess.aspx?requestslst.aspx&Type=my`

To Access the Incoming Requests Page

- Use the following URL which specifies the 'Incoming Requests' page:
`http://<host name>/PasswordVault/directaccess.aspx?requestslst.aspx&Type=incoming`

Accessing Privileged Session Recordings

Users can generate direct URL links to PSM, PSMP and OPM session recordings from activity logs and third party log systems, such as SIEM, by adding the ID of a session recording that appears in the audit log to a dynamic link that connects authorized users to PSM events. These recordings can be accessed directly by authorized users and circumvents the need to search for a specific session recording event.

To Create a Direct URL to a Privileged Session Recording

- Open the audit log that contains the ID of the session recording to include in the direct link.
- Copy the ID of the session recording and paste it at the end of the following URL:
`http://localhost/PasswordVault/directaccess.aspx?recordings.aspx&searchpattern =`

For example, if the ID of a session recording in an audit log is 30ba1095-1419-4849-b19e-dedb2b80c069, you can paste it at the end of the direct access link as follows to create a direct link to the specific recording:

`http://localhost/PasswordVault/directaccess.aspx?recordings.aspx&searchpattern =30ba1095-1419-4849-b19e-dedb2b80c069`

If the URL is specified without the 'searchpatten' keyword, a list of all the recordings stored in the Safe will be displayed.

To Access a Privileged Session Recording Directly

- Use the direct URL that specifies the ID of the privileged session recording to access the session recording directly.

Accessing PSM Live Sessions

Users can generate direct URL links to live PSM sessions from activity logs and third party log systems, such as SIEM, by adding the ID of a live session that appears in the audit log to a dynamic link that connects authorized users to PSM events. These sessions can be accessed directly by authorized users and circumvents the need to search for a specific live session.

To Create a Direct URL to a Live PSM Session

1. Open the audit log that contains the ID of the live PSM session to include in the direct link,
or,
In the Live Session Details page of the, display the Advanced tab.
2. Copy the ID of the session recording and paste it at the end of the following URL:
<PVWA HOST ADDRESS>/DirectAccess.aspx?LivesessionDetails.aspx&SearchPattern=

For example, if the ID of a live session in an audit log is e8c51d81-fc60-49a9-a323-86f5e06d116e.session, you can paste it at the end of the PSM direct access link as follows to create a direct link to the specific live session:

<http://localhost/PasswordVault/DirectAccess.aspx?LivesessionDetails.aspx&SearchPattern=e8c51d81-fc60-49a9-a323-86f5e06d116e.session>

To Access a Live PSM Session Directly

- Use the direct URL that specifies the ID of the live PSM session to access the session directly.

Logging

Logging enables you to track all the activities carried out by the Password Vault and to identify problems, if they occur.

The following three log files contain the activities of the PVWA:

- CyberArk.WebApplication.log
- CyberArk.WebConsole.log
- CyberArk.WebSession<sessionId>.log

These log files are created by the Password Vault Web Access and stored on the Web server in the location specified in the **LogFolder** parameter in the web.config file. If this location is not specified, the log files will be stored in the Temp folder on the Web server. By default, this folder is **%windir%\temp**.

Note: If you change this parameter, make sure the following IIS process user has full permissions on the folder where the log files will be stored:

- Windows 2003 – “Network service” user
- Windows 2008 and Windows 2008 R2 – “IIS AppPool\PasswordVaultWebAccessPool” user

Each time a user logs onto the Password Vault Web Access, a new log file is created. Both debug and information details are written to the same log file, according to the **DebugLevel** and **InformationLevel** parameters in the Web Access Options in the System Configuration page. For more information, refer to *Configuring the System through PVWA*, page 918.

As the Vault’s performance may be slightly slower when logging is activated, the reduced level of logging is recommended unless a problem occurs.

All log files are stored for the number of days specified in the **LogFilesRetentionPeriod** parameter. After the specified number of days, the log files will be deleted.

In addition, a PVWA console log contains informational messages and errors that refer to PVWA function. This log is meant for the system administrator who needs to monitor the status of the PVWA.

Disaster Recovery

The PWWA benefits from Disaster Recovery features which provide seamless productivity during a failover. For more information about Disaster Recovery in the Vault, refer to *CyberArk Disaster Recovery Vault*, page 885.

Transparent Failover

As soon as the PWWA cannot reach the Production Vault, the failover process begins in the DR Vault transparently, and no human intervention is required.

The IP address of both the Vault and the DR Vault can be specified in the Vault.ini configuration file. When the PWWA cannot reach the Vault specified by the first IP address, it transfers automatically to the Vault specified by the second IP address, which is the DR Vault.

To Configure Transparent Failover

1. In the Vault.ini file, in the **Address** parameter, specify the IP addresses of the Vault and the DR Vault, separated by commas, as shown in the following example:

```
Address=1.1.1.102,1.1.1.232
```

The above example indicates that the IP address of the Production Vault is 1.1.1.102 and the IP address of the DR Vault is 1.1.1.232.

2. Add the **SwitchVaultAddressTimeOut** parameter.

This parameter specifies the number of seconds that the PWWA will try to access additional Vault IP addresses after the initial timeout to the current Vault, specified in the **Timeout** parameter, expires.

If this parameter is not added, the default value of three seconds will be applied.

3. Save the Vault.ini file and close it.

Replicating PWWA Users' Passwords

As each PWWA user requires a credential file to authenticate to the Vault, it is essential that the credential files in the DR Vault are always identical to those on the Production Vault. At regular intervals, the PWWA automatically initiates a password change in the Vault and in the corresponding credential file for the PWWA. In order for the PWWA user in the DR Vault to access the Vault and continue working seamlessly in a Disaster Recovery situation, the user's new credentials must be replicated to the DR Vault whenever they are changed.

This is configured by the following parameter in the CreateCredFile utility:

- **DisableSyncPasswordToDR** – Whether or not passwords in user credential files will be replicated to all DR sites before they are replaced. The default value of this parameter is 'No', which makes sure that user credential files on all DR sites (if they exist) are synchronized with the Production Vault and that users will be able to continue working with the Vault seamlessly after a failover. If this parameter is changed to 'Yes', passwords will be replaced in credential files regardless of whether or not they have been replicated to all DR sites.

Privileged Session Manager

The Privileged Session Manager (PSM) enables organizations to secure, control and monitor privileged access to network devices by using the Vault technology to manage privileged accounts and create detailed session audits and video recordings of all IT administrator privileged sessions on remote machines.

This chapter describes how to configure PSM and how to begin working with it.

This chapter comprises the following sections:

- Administrating the Privileged Session Manager
- Architecture
- Enabling Privileged Session Management
- Configuring Privileged Session Management
- Accessing PSM Recording Sessions Directly
- Disaster Recovery
- Auditing in PSM

Administering the Privileged Session Manager

Managing the Privileged Session Manager

The PSM is installed on a Windows system as an automatic system service called CyberArk Privileged Session Manager.

It can be stopped and started through the standard Windows service management tools.

To Stop the CyberArk Privileged Session Manager Service

1. From the **Start** menu, select **Settings**, then **Control Panel**.
2. From the list of Control Panel options, select **Administrative Tools**, then **Services**; the Services window appears.
3. Right-click **CyberArk Privileged Session Manager**, and select **Stop**.

To Start the CyberArk Privileged Session Manager Service

1. From the **Start** menu, select **Settings**, then **Control Panel**.
2. From the list of Control Panel options, select **Administrative Tools**, then **Services**; the Services window appears.
3. Right-click **CyberArk Privileged Session Manager**, and select **Start**.

PSM Activity Logs

All activities that are carried out by the PSM are written in a log file and stored in the Log subfolder of the PSM installation folder from where they can be uploaded into the Vault. The maximum size of the log file is specified in the PSM configuration.

Note: Some PSM errors are only written in the event viewer.

The following log files contain the activities of the PSM:

- **PSMConsole.log** – This file contains informational messages and errors that refer to PSM function. This log is meant for the system administrator who needs to monitor the status of the PSM.
- **<SessionID>.Recorder.log** – This file contains errors and trace messages related to the PSM Recorder that can be used for troubleshooting. The types of messages that are included depend on the debug levels that are specified in the Recorder settings of the PSM configuration.
- **<SessionID>.<connection component>.log** – This file contains errors and trace messages related to the connection component that can be used for troubleshooting. The types of messages that are included depend on the debug levels that are specified in the Connection Client settings of the PSM configuration.

History Log Files

New log files are created when they reach the size specified in the **LogRotationSize** parameter in the PSM Server settings parameters. When log files reach the specified size in MB, they are timestamped and moved to the \old subfolder of the folder where they are created, and a new log file is created.

- PSM Server log files are created in the PSMLogs folder and are moved to the PSMLogs\old subfolder.
- PSM Recorder and Connection client log files are created in the PSMLogs\Components folder and are moved to the PSMLogs\Components\old subfolder.

The file is marked with a time stamp and renamed as follows:

<filename> (<date>-<time>)

For example, log files that were created in the PSMLogs folder on February 10th, 2009, at 11.30am, will be renamed as follows:

- PSMTrace.log will be renamed to PSMTrace.log.2009-02-10__11-30-00
- PSMConsole.log will be renamed to PSMConsole.log.2009-02-10__11-30-00

After they have been renamed, they will be moved to the PSMLogs\old folder.

Recording PSM Activities in the Event Viewer

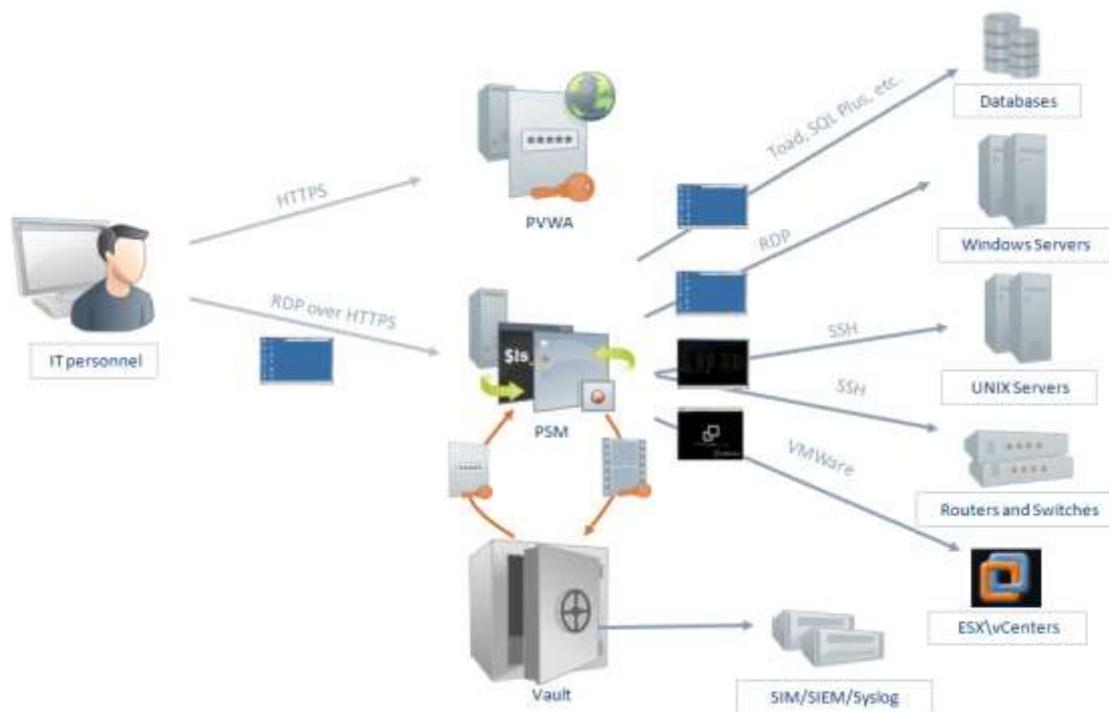
To enable standard monitoring tools to monitor the PSM by, errors are always written in the PSM machine Event Log in addition to the above log files.

In order to identify PSM components that performed activities, the following prefix is added to messages in the Event log:

- CyberArk PSM

Architecture

The PSM enables users to log onto remote machines through a proxy machine via the PVWA, using a secure protocol. The user begins the logon process by logging onto the PVWA and identifying the password to use to log onto the remote machine. When the user tries to connect transparently to the remote machine, the PVWA redirects the user to the PSM server that fetches the credentials from the Vault and connects to the remote machine using the native protocol. The activities that are performed in the privileged session are recorded by the PSM and uploaded to the Vault, where they can be accessed and viewed by auditors and other authorized users.



PSM Workflow and Architecture

By default, the user connects to the PSM proxy machine through port 3389, using RDP protocol. This is required to facilitate remote access, although this port is not usually opened in the corporate firewall, and in some cases it is not permitted.

PSM can be configured to work with the Microsoft Remote Desktop Gateway (TSGateway) which tunnels the RDP session between the user and the PSM proxy machine using HTTPS protocol (port 443), providing a secure connection without needing to open the firewall. All information that is transferred between the user and the PSM proxy machine is encrypted and protected by the HTTPS protocol, which enables secure cross-network and remote access. For more information about Microsoft Remote Desktop Gateway, refer to <http://technet.microsoft.com/en-us/library/cc731264.aspx>.

Enabling Privileged Session Management

Configuring PSM Users

All users who will use PSM connections to connect to remote machines must have the relevant permissions in Safes where the accounts that will be used are stored.

Give Users the Appropriate Safe Permissions

1. In the Safes page, open the Safe where the passwords that will be used to connect to remote devices transparently are stored.
2. Give users who will use passwords the appropriate permissions in the Safe(s) where the passwords are stored:

To view passwords and connect to remote devices:

- Use accounts
- Retrieve accounts

To connect to remote devices without viewing passwords:

- Use accounts

Setting PSM in the Master Policy

1. Click POLICIES to display the Master Policy.
2. In Session Management, select **Require privileged session monitoring and isolation**.
3. In the Rule Preview pane, click the **Edit** icon; the following options appear:
 - **Active** – This rule will be applied at Master Policy level to all platforms, unless an Exception overrides it.
 - **Inactive** – The rule will not be applied at all.

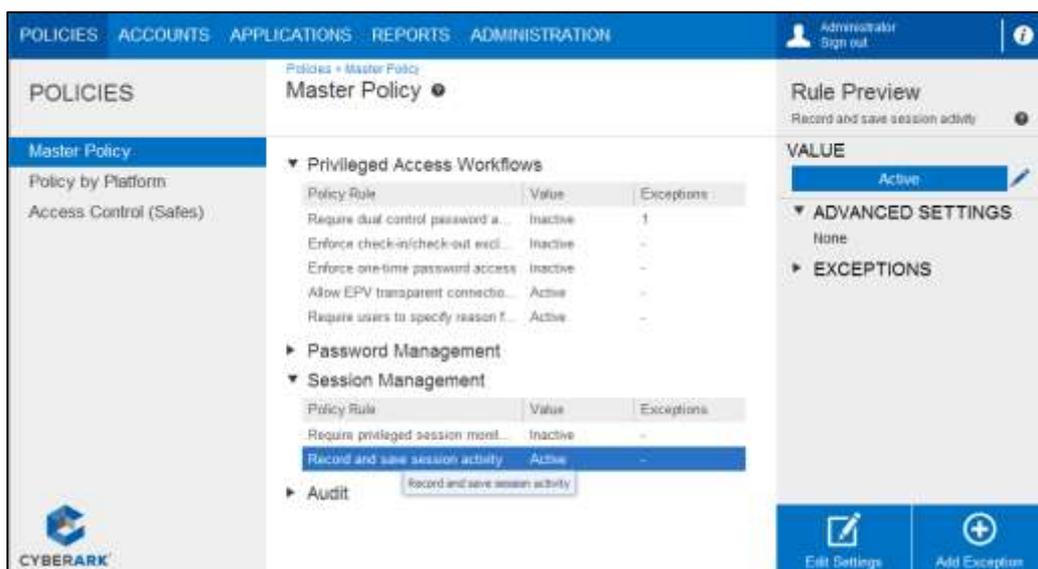
The screenshot shows the CyberArk console interface for configuring the Master Policy. The navigation menu at the top includes POLICIES, ACCOUNTS, APPLICATIONS, REPORTS, and ADMINISTRATION. The current view is 'Policies > Master Policy'. The left sidebar shows 'Master Policy' and 'Access Control (Safes)'. The main content area displays a table of policy rules under 'Session Management':

Policy Rule	Value	Exceptions
Require dual control password a...	Inactive	1
Enforce check-in/check-out excl...	Inactive	-
Enforce one-time password access	Inactive	-
Allow EPV transparent connectio...	Active	-
Require users to specify reason f...	Active	-
Require privileged session monit...	Inactive	-
Record and save session activity	Active	-

The right-hand pane shows the 'Rule Preview' for 'Require privileged session monitoring and isolation'. The current 'VALUE' is 'Inactive'. Below this are sections for 'ADVANCED SETTINGS' (set to 'None') and 'EXCEPTIONS'. At the bottom right, there are buttons for 'Edit Settings' and 'Add Exception'.

4. Select **Active**, then click the **Save** icon to save the new rule status.
5. To record privileged sessions, select **Record and save sessions activity**.

6. In the Rule Preview pane, click the **Edit** icon, then select **Active**.



7. Click the **Save** icon to save the new rule status.

Customizing PSM for Specific Platforms

After setting Session Management at Master Policy level, you can customize PSM for specific platforms, according to your enterprise needs. This enables you to control privileged SSO, secure remote access for privileged accounts that are managed by that platform, and more.

Note: To enable transparent connections, the end user's machine must have Microsoft RDP Client 5.2 or higher installed.

To Enable PSM for Platforms

1. Click **ADMINISTRATION** to display the **System Configuration** page, then click Platform Management to display a list of supported target account platforms.
2. Select the platform to configure, then click **Edit**; the settings page for the selected platform appears.
3. Expand **UI & Workflows**, and then select **Privileged Session Management**; the PSM parameters are displayed with their default values.
4. If multiple PSM servers are installed, specify the relevant PSM server ID in the ID field.
5. To enable the PSM to retrieve accounts that are required to initiate the PSM connection without requiring confirmation, even if the Safes are configured for Dual Control, change the value of **DisableDualControlForPSMConnections** to **Yes**.

6. Click **Apply** to save and apply these configurations and stay in the platform settings page,
or,
Click **Save** to save these configurations and return to the System Configuration page.

These changes will be applied the next time the PSM refreshes the configuration, according to the value of the **ConfigurationRefreshInterval** parameter in the Privileged Session Management configuration.

Enabling Session Recordings for Specific Users and Groups

The PSM can be configured at platform level to record specific users and groups. In addition, certain users and groups can be excluded from that list. For example, in an implementation where all external users' sessions are recorded, the PSM can be configured not to record a specific user, such as the *external_admin* user.

The following sections in the Privileged Session Management parameters of each platform configured for PSM define which users and groups will be recorded and which will not:

- **Recorded Users and Groups** – This section defines the users and groups whose sessions will be recorded by the Privileged Session Manager. These users and groups will only be recorded if the **Record and save session activity rule** is set in the Master Policy, and if these users and groups do not appear in the **Exclude Recorded Users and Groups** section. By default, all users and groups are recorded.
- **Exclude Recorded Users and Groups** – This section defines the users and groups whose sessions will not be recorded by the Privileged Session Manager, even when the **Record and save session activity rule** is set in the Master Policy.

To Enable the PSM for Specific Users and Groups

1. Click **ADMINISTRATION** to display the **System Configuration** page, then click Platform Management to display a list of supported target account platforms.
2. Select the platform to configure, then click **Edit**; the settings page for the selected platform appears.
3. Expand **UI & Workflows**, and then right-click **Privileged Session Management**.
4. From the drop-down menu, select **Add Recorded Users and Groups**; a new section is added to the Privileged Session Management parameters.
5. Expand the Recorded Users and Groups section and select **User or Group**.
To add additional users and groups to the list, right-click **Recorded Users and Groups**, and select **Add User or Group**.
6. In the **Properties** list, specify the name of the user or group that will be recorded when they connect to a remote device with an account associated with this platform.

7. Right-click **Privileged Session Management**, then from the drop-down menu, select **Add Exclude Recorded Users and Groups**; a new section is added to the Privileged Session Management parameters.
8. Expand the Exclude Recorded Users and Groups section and select **User or Group**.
9. In the **Properties** list, specify the name of the user or group to exclude from the Recorded Users and Groups list.
10. Click **Apply** to save and apply these configurations and stay in the platform settings page,

or,

Click **Save** to save these configurations and return to the System Configuration page.

These changes will be applied the next time the PSM refreshes the configuration, according to the value of the **ConfigurationRefreshInterval** parameter in the Privileged Session Management configuration.

Configuring Secure Connect

The PSM enables users to connect securely to remote machines through the PSM from their own workstations using all types of accounts, including accounts that are not managed in the CyberArk Vault.

Note: SSH keys cannot be used with unmanaged accounts using the secure connect capability.

Configuring Multiple Secure Connections

The PSM enables you to configure multiple secure connections so that users can access different platforms and networks from a central point. You can also customize each secure connection with its own specific settings, such as the recording Safe and other session settings.

To Configure Multiple Secure Connections

1. Click **ADMINISTRATION** to display the **System Configuration** page, then click Platform Management to display a list of supported target account platforms.
2. Select the **PSM Secure Connect** platform then click **Duplicate**; the Duplicate Platform window appears.

Note: If you have configured multiple secure connect platforms, duplicate the platform that is closest in configuration.
3. Type the name and a description of the new platform.
4. Click **Save & Close** to create the new platform.
5. Select the new target account platform, and then click **Edit**; the configuration page for the selected platform appears.

6. Change any parameter values and/or add new values to define the new platform. For example, specify the PSM server that will be used when connection to a remote device through this platform.
7. Click **Apply** to save the new configurations and apply them immediately.

Enabling Secure Connect for Specific Users and Groups

By default, all users and groups can use Secure Connect. However, you can customize it so that only specific users and groups can use it. After you specify at least one user or group, only those users can use Secure Connect and no others.

To Enable Secure Connect for Specific Users and Groups

1. Click ADMINISTRATION, and in the System Configuration page, click **Options**; the Web Access Options page appears.
2. Expand **Privileged Session Management**, then expand the **General Settings**; the configured General PSM settings are displayed.
3. Expand **Server Settings**, then expand **Secure Connect Settings**; the Secure Connect Settings are displayed.
4. Select **Secure Connect Settings**; the Secure Connect Setting properties are displayed in the Properties list.
5. Expand **Secure Connect Users and Groups**, then select **User or Group**.
 - To add additional users and groups to the list, right-click **Secure Connect Users and Groups**, and select **Add User or Group**.
6. In the **Properties** list, specify the name of the user or group that will be able to connect to remote machines using Secure Connect.
7. Click **Apply** to save and apply these configurations and stay in the Web Access Options page,
or,
Click **Save** to save these configurations and return to the System Configuration page. The changes will be applied after the period of time specified in the **ConfigurationRefresh Interval** parameter.

Customizing Secure Connect

Secure Connect platforms enable you to control PSM settings for secure connect sessions, by overriding the general PSM settings.

To Customize PSM settings for Secure Connections

1. Click **ADMINISTRATION** to display the **System Configuration** page, then click Platform Management to display a list of supported target account platforms.
2. Select **PSMSecureConnect**, then click **Edit**; the settings page for this platform appears.

Note: If you have configured multiple Secure Connect platforms, select the secure connect platform to configure.

3. Expand **UI & Workflows**, and then select **Privileged Session Management**; the Privileged Session Management properties are displayed.
4. Modify any of the properties in the PSMSecureConnect platform to determine how the PSM secure connect sessions will run.

For example, set the **ShowRecordedSessionNotification** to **Yes** to make sure that a notification is displayed on the remote machine console whenever a secure connect session begins.

5. Click **Apply** to save the new values and stay in the platform settings page, or,
Click **OK** to save them and return to the System Configuration page. The changes will be applied after the period of time specified in the **ConfigurationRefresh Interval** parameter.

Customizing Connection Components for Secure Connections

You can add and remove specific connection clients to the list of available clients or customize any internal settings in the following ways:

- **Add or remove clients from the list of secure connect clients** – Add new clients to the list of available clients in the Secure Connect page or remove existing clients from the list.
- **Customize existing secure connect clients** – Customize existing connection components to override parameters set at system level.

To Add New Connection Components to the List of Secure Connect Clients

1. Click **ADMINISTRATION** to display the **System Configuration** page, then click Platform Management to display a list of supported target account platforms.
2. Select the **PSMSecureConnect** platform, then click **Edit**; the settings page for this platform appears.

Note: If you have configured multiple Secure Connect platforms, select the secure connect platform to configure.

3. Expand **UI & Workflows**, and then **Connection Components**; all the connection components that can be used to connect to remote machines with Secure Connect are listed.
4. Right-click **Connection Components**, then from the pop-up menu select **Add Connection Component**; a new connection component is added to the current list of connection components that are configured for this platform.

5. In the new Connection Component, specify the following properties:
 - **Id** – The unique ID that identifies the connection component you created.
 - **Enable** – Whether or not this connection component will be enabled for the PSMSecureConnect platform. Specify **Yes**.
6. To define a subsection of override parameters, right-click on the component to configure, then select **Add Override User Parameters**; a new section for these override parameters is created. These parameters will override corresponding user parameters that were defined at system level. For more information about overriding configurations set at system level, refer to *Configuring Platform Parameters*, page 623.
7. To add parameters to these sections, right-click the name of the section, then select **Add Parameter**; a new parameter is added.
8. Specify the name and value of the new parameter. You can add as many new parameters as you require.
9. Click **Apply** to save the new parameter values and stay in the platform settings page,
or,
Click **OK** to save them and return to the System Configuration page. The changes will be applied after the period of time specified in the **ConfigurationRefresh Interval** parameter.

For more information about configuring audit records, commands, and other features for Secure Connect, refer to *Configuring Privileged Session Management*, page 677.

To Remove Connection Components from the List of Secure Connect Clients

You can disable connection components in the PSMSecureConnect platform and remove them from the list of secure connect clients in the Secure Connect page. Any time you wish to display them again in the list, just enable them.

1. In the **PSMSecureConnect** platform, expand **Connection Components**; a list of connection components that are configured for this platform is displayed.
Note: If you have configured multiple Secure Connect platforms, select the secure connect platform to configure.
2. Select the connection component to disable; the main connection component properties are displayed.
3. Set **Enable** to **No** to disable the connection component for this platform.
4. Click **Apply** to save the new parameter values and stay in the platform settings page,
or,
Click **OK** to save them and return to the System Configuration page. The changes will be applied after the period of time specified in the **ConfigurationRefresh Interval** parameter.

Customizing Existing Connection Components

You can customize connection components for the PSMSecureConnect platform by overriding certain features of connection components at platform level, overriding the general configuration. For example, you can override the port that will be used to connect to a remote machine with a particular client, or the protocol that will be used.

To Customize Existing Connection Components

1. In the **PSMSecureConnect** platform, expand **Connection Components**; all the connection components that can be used to connect to remote machines with Secure Connect are listed.
2. Select a connection component to modify.
3. Select a connection component to modify, then expand the list of default override parameters to display the predefined parameters.
4. Select the override parameter to change then, in the Properties list, change any of the available values. For example, you can determine whether or not a parameter is required, whether it's visible, and the default value that is displayed in the Secure Connect page when a particular client is selected.
5. Click **Apply** to save the new parameter values and stay in the platform settings page,
or,
Click **OK** to save them and return to the System Configuration page. The changes will be applied after the period of time specified in the **ConfigurationRefresh Interval** parameter.

Configuring Privileged Session Management

The PSM can be configured to provide a variety of session management activities. This section describes how to configure and implement the following functions:

- Configuring Recordings and Audits
- Configuring the Privileged Session Management Interface
- Configuring Live Session Monitoring
- Configuring PSM Connections
- Configuring the Privileged Session Manager

All configurations can be specified by authorized users in the System Configuration page. For more information, refer to *Configuring the System through PVWA*, page 918.

Configuring Recordings and Audits

The PSM records privileged sessions and stores them in the Vault where they can be viewed at any time by authorized users. It provides the following recording and audit options:

- Recordings:
 - **Video recordings** – The PSM can create video recordings for all supported connection components. By default, all these recordings are enabled.
 - **Text recordings** – The PSM can create text recordings for all supported connection components. By default, all these recordings are enabled.
Note: The PSM only captures text recordings for PSM-RDP connections in environments where single language support is configured. For more information, refer to *Configuring Universal Keystrokes for Windows Connections when an Additional Language is Used*, page 603.

These recordings are automatically configured and enabled at system level and can be overridden at platform level, enabling you to customize recordings for platforms.

- Audits:
 - **Audit records** – The PSM can create audit records for each command and event that is executed or keystrokes that are typed during privileged sessions for all supported connection components.

This section describes how to configure recordings and audits, and includes the following:

- Customizing Recordings
- Separating Keystroke Records
- Configuring Detailed Audit
- Configuring Universal Keystrokes for Windows Connections when an Additional Language is Used
- Filtering SQL Command Audits
- Hiding Passwords during Recordings
- Configuring the Recordings Safe

Customizing Recordings

Video and text recordings for PSM connections are automatically enabled at Master Policy level and are configured at PSM general level (in Web Access Options). These instructions describe how to customize these recordings at platform level, which overrides the general level.

You can customize settings for the following text recorders:

- **SQL text recorder** – The PSM can record all the commands that are executed during privileged sessions on SQL connections. This type of recording is supported for the following connection components:
 - PSM-Toad
 - PSM-SQLPlus
- **SSH text recorder** – The PSM can record all the keystrokes that are typed during privileged sessions on SSH connections. This type of recording is supported for the following connection components:
 - PSM-SSH
 - PSM-TelnetSample
- **Windows events text recorder** – The PSM can record all the Windows titles that were accessed during privileged sessions on Windows connections. This type of recording is supported for the following connection component:
 - PSM-RDP

Note: This configuration also affects SSH text recordings in PSMP.

Notes:

- Universal keystrokes recording and Windows events recording cannot be configured for the same PSM-RDP connection. Windows events text recording is enabled for PSM-RDP connections by default.
- Windows events text recording is not supported when connecting with local administrators (except for the built-in Administrator user) to systems that are UAC enabled.

Before enabling the Windows events text recorder, refer to *Configuring your PAS Environment to support Windows Events Text Recording and Windows Events Auditing*, page 600.

- **Universal keystrokes text recorder** – The PSM can record all the keystrokes that are typed during privileged sessions on all supported connections. This type of recording is supported for all connection components.

Notes:

- Universal keystroke recording and Windows events recordings cannot be configured for the same PSM-RDP connection. Windows events text recording is enabled for PSM-RDP connections by default. To enable universal keystrokes text recording, first disable Windows events text recording. For more information, refer to the relevant steps in the following procedure.
- The PSM only captures text recordings for PSM-RDP connections in environments where single language support is configured. For more information, refer to *Configuring Universal Keystrokes for Windows Connections when an Additional Language is Used*, page 603.

Before enabling the Universal keystrokes text recorder, refer to *Configuring your PAS Environment to support Universal Keystrokes Text Recording and Universal Keystrokes Auditing*, page 602.

To Customize Recordings

1. Click **ADMINISTRATION** to display the **System Configuration** page, then click **Platform Management** to display a list of supported target account platforms.
2. Select the platform to configure, then click **Edit**; the settings page for the selected platform appears.
3. Expand **UI & Workflows**, then right-click **Privileged Session Management**, a pop-up menu displays the parameter sets that you can add and customize to manage your PSM recordings.
4. From the pop-up menu, select **Add Recorder Settings**; a new set of parameters called Recorder Settings is added.
5. Disable or customize video recordings for this platform:
 - i. Expand **Recorder Settings** and select **Video Recorder**.
 - ii. By default, video recordings are enabled. To disable video recordings, set the value of **Enabled** to **No**.
6. Disable or customize text recordings for this platform:
 - To disable or customize SSH text recordings:

Note: These settings affect SSH text recordings for SSH connections through PSM as well as through PSMP.

 - i. Right-click **Recorder Settings**, then select **Add SSH Text Recorder**; a new set of parameters called SSH Text Recorder is added.
 - ii. By default, SSH text recordings for SSH connections are enabled. To disable these recordings for this platform, set the value of **Enabled** to **No**.
 - iii. Define the channels that will be recorded during the session. By default, the following channels are recorded for SSH connections:

Property	Default Value	Description
In	Yes	Whether or not the terminal's STDIN stream will be recorded.
Out	Yes	Whether or not the terminal's STDOUT and STDERR streams will be recorded.

Property	Default Value	Description
Keystrokes	Yes	Whether or not all the keystrokes logged by the user from the start of the line until the user presses Enter will be recorded. Note: Control characters are not recorded.

To disable recordings on any of these channels, set the value of the channel property to **No**.

- To disable or customize SQL text recordings:
 - i. Right-click **Recorder Settings**, then select **Add SQL Text Recorder**; a new set of parameters called SQL Text Recorder is added.
 - ii. By default, SQL text recordings for SQL connections are enabled. To disable these recordings for this platform, set the value of **Enabled** to **No**.
 - iii. Define the channels that will be recorded during the session. By default, the following channels are recorded for Oracle Database connections:

Property	Default Value	Description
In	Yes	Whether or not SQL commands will be recorded.

As this is the only channel that is recorded for SQL text recordings, this channel must be enabled in order for sessions to be recorded.

- To disable or customize Windows events text recordings:
 - i. Right-click **Recorder Settings**, then select **Add Windows Events Text Recorder**; a new set of parameters called Windows Events Text Recorder is added.
 - ii. By default, Windows events text recordings for Windows connections are enabled. To disable these recordings for this platform, set the value of **Enabled** to **No**.
 - iii. Define the channels that will be recorded during the session. By default, the following channels are recorded for Windows connections:

Property	Default Value	Description
WindowTitles	Yes	Whether or not window titles will be recorded in a text file.

As this is the only channel that is recorded for Windows Events text recordings, this channel must be enabled in order for sessions to be recorded.

- To disable or customize Universal Keystrokes text recordings:
 - i. Right-click **Recorder Settings**, then select **Add Keystrokes Text Recorder**; a new set of parameters called Keystrokes Text Recorder is added.
 - ii. By default, universal keystrokes text recording is enabled for the supported connection components except PSM-RDP.
 - To disable auditing for any component, in the Properties list, set the value of **Enable** to **No**.

- To enable these recordings for other platforms, set the value of **Enabled** to **Yes**.

Note: Text recordings for PSM-RDP connections can only be enabled in environments where single language support is configured. For more information, refer to *Configuring Universal Keystrokes for Windows Connections when an Additional Language is Used*, page 603.

- iii. Define the channels that will be recorded during the session. By default, the following channels are recorded for Keystrokes Text auditing:

Property	Default Value	Description
In	Yes	Whether or not the PSM will include each individual keystroke that was typed by the user in the text recording file.
Keystrokes	Yes	Whether or not all the keystrokes logged by the user from the start of the line until the user presses Enter will be recorded. Note: You can change the characters that define when a keystrokes audit record ends.

To disable recordings on any of these channels, set the value of the channel property to **No**.

7. Click **Apply** to save the new parameter values and stay in the platform settings page,
or,
Click **OK** to save them and return to the System Configuration page. The changes will be applied after the period of time specified in the **ConfigurationRefresh Interval** parameter.

Separating Keystroke Records

You can define a list of keys that indicate when a keystrokes audit record ends. This list can be defined for each connection component and can be overridden for a specific platform. Whenever the user strikes a key on the keyboard from this list, the PSM creates a new audit record that contains the group of keys that were typed up to this point.

Note: In an environment where support for multiple languages is configured, in PSM-RDP connections the only separator key is Enter and it cannot be changed. For more information, refer to *Configuring Universal Keystrokes for Windows Connections when an Additional Language is Used*, page 603.

To override the separator keys for a connection component in your platform

1. Click **ADMINISTRATION** to display the **System Configuration** page, then click **Platform Management** to display a list of supported target account platforms.
2. Select the platform to configure, then click **Edit**; the settings page for the selected platform appears.
3. Expand **UI & Workflows**, and then expand **Connection Components**.
4. Right-click the connection component to configure then, from the pop-up menu, select **Add Override target settings**; a new set of Override target settings are added to the Connection Component.

5. Expand **Override target Settings**, then right-click **Client Specific** parameters, and select **Add Multiline Parameter**; a new parameter is added.
6. In the Properties list, in the Name property, specify the name of the multiline property. Specify **KeystrokesRecordSeparator**.
7. Click the **Value** property; an edit box appears to enable you to specify the list of keys that indicate when a keystrokes audit record ends. As this is a multiline parameter, each line represents a single key. Any key can be specified in this list, although special characters must be enclosed with parenthesis and are case sensitive. For example, [Tab] or [RCtrl]. The default value is the **Enter** key.

Specify any regular character or any of the following special characters: [RAlt] [LAlt] [LShift] [RShift] [LCtrl] [RCtrl] [F1] [F2] [F3] [F4] [F5] [F6] [F7] [F8] [F9] [F10] [F11] [F12] [Esc] [Home] [Delete] [Insert] [End] [PageUp] [PageDown] [Pause/Break] [LWinKey] [RWinKey] [Menu] [Tab] [LeftArrow] [RightArrow] [UpArrow] [DownArrow] [Backspace] [CapsLock] [NumLock] [ScrollLock] [Enter]
8. Click **OK**; the list of keys that indicate when a keystrokes audit record ends is displayed in the Value property as one line.
9. Click **Apply** to apply the new configurations,
or,
Click **OK** to save the new configurations and return to the System Configuration page.

Configuring Detailed Audit

By default, the PSM records all the activities that take place during privileged sessions and provides audits for the following events:

- **SQL commands** – The PSM can record all the commands that were executed during privileged SQL sessions. This type of auditing is supported for the following connection components:
 - PSM-Toad
 - PSM-SQLPlus
- **SSH keystrokes** – The PSM can record all the keystrokes that are carried out during privileged SSH sessions. This type of auditing is supported for the following connection component:
 - PSM-SSH

Note: This configuration also affects SSH keystrokes audit in PSMP.
- **Window titles** – The PSM can record the titles of the windows that are displayed during privileged Windows sessions. This type of auditing is supported for the following connection component:
 - PSM-RDP

Notes:

 - Universal keystrokes recording and Windows events recording cannot be configured for the same PSM-RDP connection. Windows events recording is enabled for PSM-RDP connections by default.
 - Windows events audit is not supported when connecting with local administrators (except for the built-in Administrator user) to systems with UAC enabled.

Before enabling the Windows events audit, refer to *Configuring your PAS Environment to support Windows Events Text Recording and Windows Events Auditing*, page 600.

- **Universal keystrokes** – The PSM can record all the keystrokes that are carried out during all privileged sessions. This type of auditing is supported for all connection components.

Note: Universal keystroke recording and Windows events recording cannot be configured for the same PSM-RDP connection. Windows events recording is enabled for PSM-RDP connections by default. To enable universal keystrokes recording, first disable Windows events recording. For more information, refer to the relevant steps in the following procedure.

Before enabling the Universal keystrokes audit, refer to *Configuring your PAS Environment to support Universal Keystrokes Text Recording and Universal Keystrokes Auditing*, page 602.

In environments where single language support is configured, you can benefit from Universal keystrokes for PSM-RDP connections without any extra configuration. In environments where additional language support is configured, specific prerequisites are required. For more information, refer to *Configuring Universal Keystrokes for Windows Connections when an Additional Language is Used*, page 603.

Detailed audit is automatically configured and enabled at system level and can be overridden at platform level, enabling you to customize detailed audit for platforms.

To Configure Detailed Audit

1. Click **ADMINISTRATION** to display the **System Configuration** page, then click **Platform Management** to display a list of supported target account platforms.
2. Select the platform to configure, then click **Edit**; the settings page for the selected platform appears.
3. Expand **UI & Workflows**, and then right-click **Privileged Session Management**.
4. From the pop-up menu, select **Add Audit Settings**; a new parameter is added to the Privileged Session Management settings.
5. Select the Audit Settings, then from the pop-up menu, select one of the following options, depending on the audit settings to disable or customize:
 - To disable or customize SQL Level Audits for PSM-Toad and PSM-SQLPlus connection components using this platform:
 - i. Right-click **Audit Settings**, then from the pop-up menu, select **Add SQL Level Audit**.
 - ii. By default, SQL level auditing is enabled for the supported connection components. To disable auditing for these components, in the Properties list, set the value of **Enable** to **No**.
 - iii. Configure advanced properties to determine how the PSM will manage audit records. For more information about these properties, refer the Privileged Account Security Reference Guide.
 - To disable or customize SSH Keystrokes Audits for PSM-SSH and/or PSMP-SSH connection components using this platform:
 - i. Right-click **Audit Settings**, then from the pop-up menu, select **Add SSH Keystrokes Audit**.
 - ii. By default, SSH keystrokes auditing is enabled for the supported connection component. To disable auditing for this component, in the Properties list, set the value of **Enable** to **No**.

- iii. Configure advanced properties to determine how the PSM will manage audit records. For more information about these properties, refer to the Privileged Account Security Reference Guide.
- Note:** This configuration affects SSH Keystrokes Audits in both PSM and PSMP.
- To disable or customize Windows Events Audits for PSM-RDP connection components using this policy:
 - i. Right-click **Audit Settings**, then from the pop-up menu, select **Add Windows Events Audit**.
 - ii. By default, Windows events auditing is enabled for the supported connection component. To disable auditing for this component, in the Properties list, set the value of **Enable** to **No**.
 - iii. Configure additional properties to determine how the PSM will manage audit records. For more information about these properties, refer to the Privileged Account Security Reference Guide.
 - To disable or customize universal keystrokes audit for all connection components using this platform:
 - i. Right-click **Audit Settings**, then from the pop-up menu, select **Add Keystrokes Audit**.
 - ii. By default, universal keystrokes audit is enabled for the supported connection components except PSM-RDP.
 - To disable auditing for any component, in the Properties list, set the value of **Enable** to **No**.
 - To enable these recordings for other platforms, set the value of **Enabled** to **Yes**.
 - iii. Configure advanced properties to determine how the PSM will manage audit records. For more information about these properties, refer to the Privileged Account Security Reference Guide.
6. Click **Apply** to save the new parameter values and stay in the platform settings page,
or,
Click **OK** to save them and return to the System Configuration page. The changes will be applied after the period of time specified in the **ConfigurationRefreshInterval** parameter.

Configuring your PAS Environment to support Windows Events Text Recording and Windows Events Auditing

Before enabling Windows events text recording or Windows events auditing, configure your PAS environment, as described below:

- CyberArk component compatibility:
 - All PSM servers in your environment must be V8.2 or above.
 - All PSM SSH-Proxy servers in your environment must be V7.2.9 or above.
 - The Vault and the PVWA components must be V8.2 or above.
- Target server prerequisites:
 - A share called "admin" must be available on the target server.
 - Make sure the "SERVER" Windows service is running.
 - In the firewall, open TCP port 445.
 - The account used to access the target machine must belong to the Administrators Group.

Note: To enable Detailed Session Auditing, PSM installs a service on the target machine. The service starts when a new session is initiated, and stops immediately after the session is established.

1. Log on to the PrivateArk Client as an administrative user and open the PVWAConfig Safe.
2. Right-click the **PVConfiguration.xml** file and retrieve it for editing.

3. In the **PVConfiguration.xml** file, make the following changes:
- i. Under the PSM-RDP node, locate the **TargetSettings** node and add the **Capabilities** node as its last child node, as shown in bold text in the following example:

```
<ConnectionComponent Id="PSM-RDP" Height="768" Width="1024"
FullScreen="no"
Type="CyberArk.PasswordVault.TransparentConnection.PSM.PSMConne
ctionComponent,
CyberArk.PasswordVault.TransparentConnection.PSM">
  <ComponentParameters />
  <UserParameters>
    ...
  </UserParameters>
  <TargetSettings Protocol="RDP" ClientApp="mstsc.exe"
ClientDispatcher="PSMRdpClient.exe">
    <ClientSpecific>
      <Parameter Name="port" Value="3389" />
    </ClientSpecific>
    <LockAppWindow Enable="No" />
    <Capabilities>
      <Capability Id="WindowsEventsTextRecorder" />
      <Capability Id="WindowsEventsAudit" />
    </Capabilities>
  </TargetSettings>
</ConnectionComponent>
```

- ii. Under the **ConnectionClientSettings** node, locate the **Capabilities** node and add the **WindowsEventsTextRecorder** and **WindowsEventsAudit** nodes as the last child nodes, as shown in bold text in the following example:

```
<ConnectionClientSettings>
  <Capabilities>
    ...
    <WindowsEventsTextRecorder Id="WindowsEventsTextRecorder"
Description="Windows events text recorder" Type="TextRecorder"
IntegrationType="Embedded" Format="WIN">
      <WindowsEventsTextRecorder>
        <Channels />
      </WindowsEventsTextRecorder>
    </WindowsEventsTextRecorder>
    <WindowsEventsAudit Id="WindowsEventsAudit"
Description="Windows events audit" Type="Auditer"
IntegrationType="Embedded" Format="WIN">
      <WindowsEventsAudit>
        <Channels />
      </WindowsEventsAudit>
    </WindowsEventsAudit>
    ...
  </Capabilities>
</ConnectionClientSettings>
```

4. Save the changes and return the **PVConfiguration.xml** file to the **PVWACConfig** Safe. The changes will be applied after the period of time specified in the **ConfigurationRefreshInterval** parameter.

Configuring your PAS Environment to support Universal Keystrokes Text Recording and Universal Keystrokes Auditing

Before enabling Universal keystrokes text recording or Universal keystrokes auditing, configure your PAS environment, as described below:

- CyberArk Component Compatibility:
 - All PSM servers in your environment must be V8.6 or above.
 - All PSM SSH-Proxy servers in your environment must be V7.2.12 or above.
 - The Vault and the PVWA components must be V8.6 or above.
1. Log on to the PrivateArk Client as an administrative user and open the **PVWAConfig** Safe.
 2. Right-click the **PVConfiguration.xml** file and retrieve it for editing.
 3. In the **PVConfiguration.xml** file, make the following changes:
 - i. Under the **ConnectionClientSettings** node, locate the **Capabilities** node and add the **KeystrokesTextRecorder** and **KeystrokesAudit** nodes as the last child nodes, as shown in bold text in the following example:

```

<ConnectionClientSettings>
  <Capabilities>
    ...
    <KeystrokesTextRecorder Id="KeystrokesTextRecorder"
      Description="Keystrokes text recorder" Type="TextRecorder"
      IntegrationType="Embedded" Format="Keystrokes">
      <KeystrokesTextRecorder>
      <Channels />
    </KeystrokesTextRecorder>
  </KeystrokesTextRecorder>
  <KeystrokesAudit Id="KeystrokesAudit" Description="Keystrokes
  audit" Type="Auditer" IntegrationType="Embedded"
  Format="Keystrokes">
  <KeystrokesAudit>
  <Channels />
  </KeystrokesAudit>
  </KeystrokesAudit>
  ...
</Capabilities>
</ConnectionClientSettings>

```

- ii. For every connection component in which you want to add the universal keystrokes features, locate the **TargetSettings** node and add the **Capabilities** node as its last child node.

Note: If the **Capabilities** node already exists, add the new **KeystrokesTextRecorder** and **KeystrokesAudit** nodes beneath it.

The bold text in the example below shows how to add the universal keystrokes features to the **PSM-SQLServerMgmtStudio** connection component. To configure other connection components, add the same text in their configuration.

```

<ConnectionComponent Id="PSM-SQLServerMgmtStudio"
Type="CyberArk.PasswordVault.TransparentConnection.PSM.PSMConne
ctionComponent,
CyberArk.PasswordVault.TransparentConnection.PSM">
<ComponentParameters />
<UserParameters>
...
</UserParameters>
<TargetSettings Protocol="SQLNet" ClientApp="&quot;C:\Program
Files (x86)\Microsoft SQL
Server\100\Tools\Binn\VSShell\Common7\IDE\Ssms.exe&quot; -S
&quot;{Address}&quot; -U &quot;{UserName}&quot; -P
&quot;{Password}&quot;" ClientDispatcher="NA"
ClientInvokeType="CommandLine">
<ClientSpecific>
...
</ClientSpecific>
<LockAppWindow Enable="Yes"
MainWindowClass="wndclass_desked_gsk" Timeout="800000"
SearchWindowWaitTimeout="30" MainWindowTitle="Microsoft SQL
Server Management Studio" />
<Capabilities>
<Capability Id="KeystrokesAudit" />
<Capability Id="KeystrokesTextRecorder" />
</Capabilities>
</TargetSettings>
</ConnectionComponent>

```

4. Save the changes and return the **PVConfiguration.xml** file to the **PVWAConfig Safe**. The changes will be applied after the period of time specified in the **ConfigurationRefreshInterval** parameter.

Configuring Universal Keystrokes for Windows Connections when an Additional Language is Used

Universal keystrokes recording is configured by default to support Windows sessions in which a single language is used.

If you use an additional language in your Windows sessions (for example, if you use both English and French keyboards), configure the Universal keystrokes as described in this section.

Prerequisites and Limitations when Additional Language Support is Enabled

- On the target machine:
 - The PSM requires the following prerequisites:
 - A share called "admin" must be available on the target server.
 - Make sure the "SERVER" Windows service is running.
 - In the firewall, open TCP port 445.
 - The account used to access the target machine must belong to the Administrators Group.

Note: To enable Universal Keystrokes for Windows sessions when additional language support is enabled, PSM installs a service on the target machine. The service starts when a new session is initiated, and stops immediately after the session is established.

- Add the additional language as an extra keyboard for the target account user on the target machine.
- When Windows Keystrokes additional language support is enabled, only connections to Win2008R2 or Win2012R2 target systems are supported.
- On the PSM Server:
 - Set the system locale to the additional language.

To Configure Universal Keystrokes to Support an Additional Language

By default, single language support for capturing Universal Keystrokes for Windows sessions is configured at system level. This setting can be overridden at system or platform level, enabling you to customize additional language support according to your preferences.

1. Click **ADMINISTRATION** to display the **System Configuration** page, then click **Platform Management** to display a list of supported target account platforms.
2. Select the platform to configure, then click **Edit**; the settings page for the selected platform appears.
3. Expand **UI & Workflows**, and then expand **Connection Components**.
4. Right-click the Windows connection component to configure. By default, the Windows connection component is PSM-RDP.
5. From the Connection Component pop-up menu, select **Add Override target settings**; a new set of Override target settings are added to the Connection Component.
6. Expand **Override target Settings**, then right-click **Client Specific** parameters, and select **Add Parameter**; a new parameter is added.
7. In the Properties list, in the Name property, specify **WindowsKeystrokesSingleLanguage**.
8. In the Properties list, in the **Value** property, specify **No**.
Note: To revert to single language support, change this Value to Yes.
9. Click **Apply** to apply the new configurations,
or,
Click **OK** to save the new configurations and return to the System Configuration page.

Filtering SQL Command Audits

The PSM can filter SQL command audits that are recorded during PSM-Toad and PSM-SQLPlus connections to minimize unwanted audit records, reducing the number of audit records stored in the Vault and increasing server performance. Filters can be created at system level to apply to all SQL commands issued through PSM connections, or at platform level to apply to SQL commands issued through connections that are linked to a specific platform.

You can define lists to filter commands that are recorded according to the following criteria:

- **Commands to audit** – A **whitelist** is a list of SQL commands that will be included in the command audit records. All other commands will not be included. By default, all commands that are issued during privileged sessions are audited. However, after you create a whitelist, only the listed commands will be audited, if they do not appear in the blacklist.
- **Commands not to audit** – A **blacklist** is a list of SQL commands that will be excluded from audit records. All other commands will be included.

By defining blacklists and whitelists, you assert granular control over audit records in the Vault and determine exactly which commands will be audited. These lists are created in audit filter rules as regular expressions which define specific commands. You can create as many rules as you require for blacklists as well as whitelists, as well as lists that combine them both.

Example 1 – Blacklist:

By default, the PSM includes a single blacklist that excludes the multiple commands that are issued automatically at the start of each Toad session. These commands are predetermined as part of the Toad setup, and are not relevant to the privileged session, other than to start it. This blacklist excludes these commands from the session audit, and reduces the number of audit records stored in the Vault.

Example 2 – Whitelist:

The following example describes an example of when you would require a whitelist: You wish to audit all DDL queries such as 'update', 'insert', and 'delete' so that you know who issues these commands, when, and from which station. However, you don't need to audit other commands that are issued. You can create a whitelist that contains these commands, ensuring that every time these specific commands are issued during the privileged session, they are audited.

To Enable/Disable the SQL Command Audit Filter

1. Click ADMINISTRATION, then in the System Configuration page click **Options**; the Web Access Options are displayed.
2. Expand the **Audit Filters** parameters, then select **SQLLevelAudit**; the following properties of the SQL Level Audit filter are displayed in the Properties list:
 - **Id** – The unique ID of the audit filter.
 - **Description** – A description of the audit filter.
3. Expand the **SQLLevelAudit** filter to display the predefined audit filter rules. Each rule is configured for the system, and can be overridden at platform level.

4. Select an audit filter rule to display the rule's Properties list, which includes the following:
 - **Id** – The unique ID of the audit filter rule.
 - **Type** – Whether this rule is a blacklist (exclude) or a whitelist (include).
 - **EnableForReports** – Whether or not this rule is enabled by default for reports. This property is for future use.
 - **EnableForAudit** – Whether or not this rule is enabled by default for auditing.
 - **Description** – A description of the audit rule.
5. Enable/disable the audit filter rule:
 - To enable the audit filter rule – Set **EnableForAudit** to **Yes**; the audit filter rule will be applied to all commands issued during PSM-Toad and PSM-SQLPlus connections, regardless of the platform that is used. For more information about enabling audit filters for a specific platform, refer to *To Apply SQL Command Audit Filters to Specific Platforms*, page 607.

Note: By default, before a whitelist is enabled, all commands are audited. After enabling the first whitelist, only the commands specified in this whitelist will be audited. To audit more commands, create and enable additional whitelists.
 - To disable the audit filter rule – Set **EnableForAudit** to **No**; the audit filter rule will be cancelled and the filter rule will not be applied to commands issued during PSM-Toad and PSM-SQLPlus connections.
6. Click **Apply** to save the new parameter values and stay in the Web Access Options page,
or,
Click **OK** to save them and return to the System Configuration page.

These changes will be applied the next time the PSM refreshes the configuration, according to the value of the **ConfigurationRefreshInterval** parameter in the Privileged Session Management configuration.

To Create an SQL Command Audit Filter

1. Click ADMINISTRATION, then in the System Configuration page, click **Options**; the Web Access Options are displayed.
2. Expand the **Audit Filters** parameters, then right-click **SQLLevelAudit**.
3. From the pop-up menu, select **Add Audit Filter Rule**; a new audit filter rule is added to the list of audit filters and the properties of the new rule are displayed.
4. Specify the following properties for the new audit filter rule:
 - **Id** – The unique ID of the audit filter rule.
 - **Type** – Whether this rule is a blacklist or a whitelist.
 - To create a **blacklist**, specify **Exclude**.
 - To create a **whitelist**, specify **Include**.
 - **EnableForReports** – Whether or not this rule is enabled by default for reports. This property is for future use.
 - **EnableForAudit** – Whether or not this rule is enabled by default for auditing. Specify **Yes** to enable this audit filter rule.
 - **Description** – A description of the audit rule.

5. Right-click **Audit Filter Rule**, then from the pop-up menu, select **Add Regular Expression**; a new parameter is created in which you can specify the regular expression that defines a single audit filter.
6. In the Properties list, in the **RegExp** property, specify the regular expression to filter. Repeat this step to list all the commands that will be filtered during recorded privileged sessions.
 - **Blacklist** – This list specifies the commands that **will not** be included in audits of the privileged session.
 - **Whitelist** – This list specifies the commands that **will** be included in audits of the privileged session. No other commands will be audited.
7. Click **Apply** to save the new parameter values and stay in the Web Access Options page,
or,
Click **OK** to save them and return to the System Configuration page.

These changes will be applied the next time the PSM refreshes the configuration, according to the value of the **ConfigurationRefreshInterval** parameter in the Privileged Session Management configuration.

To Apply SQL Command Audit Filters to Specific Platforms

1. Click **ADMINISTRATION** to display the **System Configuration** page, then click Platform Management to display a list of supported target account platforms.
2. Select the platform to configure, then click **Edit**; the settings page for the selected platform appears.

Note: This is only relevant to platforms that use the following connection components:

 - PSM-Toad
 - PSM-SQL Plus
3. Expand **UI & Workflows**, and then right-click **Privileged Session Management**, then from the pop-up menu, select **Add Audit Settings**; a new set of parameters is created for Audit Settings.
4. Right-click **Audit Settings**, then from the pop-up menu, select **Add Audit Filters Override**; a new set of parameters is created, in which you can add additional rule parameters that will override the audit filters rule that is currently set at system level.
5. Right-click **Audit Filters Override**, then from the pop-up menu, select **Add Audit Filter Rule Override**; a new parameter is added with the following property:
 - **AuditFilterId** – The unique ID of the audit filter to override at platform level. This ID is specified in the Audit Filters rules in Web Access Options. For more information about locating this property, refer to *To Enable/Disable the SQL Command Audit Filter*, page 605.
6. Right-click **Audit Filter Rule Override**, then from the pop-up menu, select **Add Rule**; a new parameter is added with the following properties:
 - **Id** – The unique ID of the rule to override. This ID is specified in the Audit Filters rules in Web Access Options. For more information about locating this property, refer to *To Enable/Disable the SQL Command Audit Filter*, page 605.
 - **EnableForAudit** – Whether or not this rule is enabled by default for auditing. This property overrides the same property at system level for this platform only.

7. Click **Apply** to save the new parameter values and stay in the platform settings page,
or,

Click **OK** to save them and return to the System Configuration page.

These changes will be applied the next time the PSM refreshes the configuration, according to the value of the **ConfigurationRefreshInterval** parameter in the Privileged Session Management configuration.

Hiding Passwords during Recordings

The PSM identifies passwords that are typed by users during SSH and Telnet sessions by looking for password prompts. By default, the prompts that the PSM looks for include common prompts for Unix platforms or for Vault passwords. Customize this list to include all password prompts that are received in your environment. When users type a character that cannot be included in a password, such as a space, or when they press Enter, the PSM resumes the audit and recording. You can update this list of characters too.

This can be configured at platform level, overriding the general configuration.

Note: This configuration affects both PSM and PSMP.

To Hide Passwords during Recordings

1. Click **ADMINISTRATION** to display the **System Configuration** page, then click Platform Management to display a list of supported target account platforms.
2. Select the platform to configure, then click **Edit**; the settings page for the selected platform appears.
3. Expand **UI & Workflows**, then right-click **Privileged Session Management**: a pop-up menu displays the parameter sets that you can add and customize to manage your PSM recordings.
4. From the pop-up menu, select **Internal Capability Settings**; a new set of parameters called Internal Capability Settings is added.
5. Right-click **Internal Capability Settings**, then from the pop-up menu, select **Add SSH Password Hiding**; a new capability parameter is added.
6. Select **SSH Password Hiding**, then specify the following properties:
 - **Enabled** – Determines whether or not passwords will be recorded during PSMP sessions. The default value is **Yes**, indicating that this feature is enabled and passwords will not be recorded.
 - **PasswordPrompts** – This is a regular expression that is used to identify password prompts. When the system finds a match to this regular expression, it omits the password from the PSM session recording.
 - **InvalidPasswordChars** – Defines characters that cannot be included in passwords. When the user specifies one of these characters, the PSM resumes auditing and recording each keystroke. The default values are spaces and tabs.
7. Click **Apply** to save the new parameter values and stay in the Web Access Options page,
or,

Click **OK** to save them and return to the System Configuration page. The changes will be applied after the period of time specified in the **ConfigurationRefreshInterval** parameter.

Configuring the Recordings Safe

In each platform, you can specify the Safe where recordings of activities for accounts associated with that platform will be stored. You can set a different recordings Safe for each platform, depending on your access control policy.

To Configure the Recordings Safe for a Platform

1. Log onto the Password Vault Web Access as a user with permission to configure platforms.
2. Click **ADMINISTRATION** to display the **System Configuration** page, then click Platform Management to display a list of supported target account platforms.
3. Select the platform to configure, then click **Edit**; the settings page for the selected platform appears.
4. Expand **UI & Workflows**, and then select **Privileged Session Management**; the PSM parameters are displayed with their default values.
5. In the **SessionRecorderSafe** property, specify the name of the Safe where recordings of activities for accounts associated with the platform will be stored.

Specify either of the following:

- **Safe name** – A Safe name that will be created exactly as you specified.
- **Dynamic Safe name** – Define the location of session recordings and create separate recording Safes for different accounts without needing to create a new PVWA platform for it, by specifying a Safe name that comprises a dynamic reference to the Safe name of the account that was used to initiate the privileged session. Specify a partial Safe name and then {AccountSafeName} to create a Safe whose name includes the name of the Safe where the account used to initiate the session is stored. For example, if an account that is stored in a Safe called 'Windows' is used, PSM-{AccountSafeName} will create a Safe called PSM-Windows.

This Safe will be created when the first recording is uploaded to it.

6. Click **Apply** to save the new parameter values and stay in the same page,
or,
Click **OK** to save them and return to the System Configuration page.

To Create a Recordings Safe in Advance

You can create PSM recordings Safes before initiating sessions, in order to define specific permissions for them and not let the PSM create them automatically and allocate default permissions.

- Add the PSMAppUsers group as a member of this Safe with the following permissions:
 - Retrieve accounts/files
 - List accounts/files
 - View audit

Configuring the Privileged Session Management Interface

The **Privileged Session Management UI** parameters of the Web Access Options determine how PSM-related items will be displayed in the PVWA, as well as the user experience during PSM sessions. This section describes how to configure the following:

- Configuring the PSM Session User Experience
- Downloading Session Recordings
- Searching for Session Recordings
- Viewing Recording Details
- Viewing Account Recordings
- Displaying Live Sessions
- Direct Playback in the PVWA
- Displaying Notifications when Sessions are Recorded

Configuring the PSM Session User Experience

The following general Privileged Session Management UI parameters configure the user experience for PSM sessions and define which method will be used to establish the connection:

- The **NonIERemoteDesktopAccess** parameter determines whether the external tool or the built-in solution via an RDP file, will be used to establish PSM RDP connections when the connection is not made with Microsoft RDP ActiveX. The default is RDPFile.
- The **ConnectPSMWithRDPActiveX** parameter determines whether PSM will connect with Microsoft RDP ActiveX or with the non-IE access method that was configured in the **NonIERemote DesktopAccess** parameter. Possible values are:
 - **Always** – The RDP ActiveX will always be used to establish PSM connections, preventing connections from non-IE browsers, such as Firefox. Neither the external tool nor RDP files will be used to establish connections.
 - **ByBrowser** – Depending on the setting of the **NonIERemoteDesktopAccess** parameter, either the external tool or an RDP file will be used to establish PSM connections from non-IE browsers, such as Firefox. Microsoft RDP ActiveX will be used to establish connections through IE.
 - **Never** – Depending on the setting of the **NonIERemoteDesktopAccess** parameter, either the external tool or an RDP file will be used to establish PSM connections from both IE and non-IE browsers. Microsoft RDP ActiveX will not be used to establish connections. This is the default setting.
- The **UseRemoteApp** parameter determines whether or not PSM sessions are displayed in a standard client window, facilitating an intuitive user experience. This is only relevant when PSM connections are established using an RDP file and the PSM is installed on Windows 2012R2. By default, during PVWA installation this parameter is set to **No**, and PSM installation on Windows 2012R2 automatically changes it to **Yes**.

The following table shows the user experience according to how the above parameters are set.

NonIERemote Desktop Access Connect PSMWith RDPActiveX	RDPFile			External Tool (HOB)		
		Connection Method	RemoteApp experience		Connection Method	RemoteApp experience
Always	IE	ActiveX	✘	IE	ActiveX	✘
	Non-IE	Not supported	Not supported	Non-IE	Not supported	Not supported
		Connection Method	RemoteApp experience		Connection Method	RemoteApp experience
By Browser	IE	ActiveX	✘	IE	ActiveX	✘
	Non-IE	RDP file	✓	Non-IE	External tool	✘
		Connection Method	RemoteApp experience		Connection Method	RemoteApp experience
Never	IE	RDP file	✓	IE	External tool	✘
	Non-IE	RDP file	✓	Non-IE	External tool	✘
		Connection Method	RemoteApp experience		Connection Method	RemoteApp experience
Requirements	<ul style="list-style-type: none"> Requires PSM 9.2 or higher and Vault/PVWA 9.2 or higher. RemoteApp user experience requires: <ul style="list-style-type: none"> PSM must be installed on Windows 2012R2. RDP client v6.1.7601 or above (RDP protocol version v7.1 or above) on end user machines. 			<ul style="list-style-type: none"> Requires HOB installation and special CyberArk license. 		

Notes:

- Connections from Unix/Linux environments are only supported with an external tool.
- Connections when NLA is enabled on the PSM server are only supported with an external tool.
- Connections with RD Gateway are only supported when connecting with ActiveX.

Downloading Session Recordings

The following general Privileged Session Management UI parameter configures how recordings are downloaded.

- The **RedirectToRecordingDetailsOnPlay** parameter determines whether or not the Play button will download recordings immediately or will redirect the page to the Record Details page and then download the file. If the recording is downloaded immediately, the page might be refreshed once due to browser security restrictions.

Searching for Session Recordings

The **Search Session Recordings** parameters, in the **Privileged Session Management UI** parameters, configure the Search for Sessions page.

The following **General** parameters configure the search criteria that users will be able to specify in order to locate session recordings.

- The **DefaultFilterByDates** parameter determines whether or not the 'Filter by dates' section will be enabled by default. By default, this parameter is set to 'No'.
- The **DefaultFromTime** parameter specifies the default filter time for the search if a 'From date' is specified. By default, searches include recordings that occurred after 08:00.
- The **DefaultToTime** parameter specifies the default filter time for the search if a 'To date' is specified. By default, searches include recordings that occurred before 23:45.
- The **DisplaySafeInSearch** parameter determines whether or not an additional text box will be displayed in the search bar to enable users to specify a Safe pattern. This will be used to filter Safes during a search. By default, this parameter is set to 'No'.
- The **OptimizeRecordingsSearch** parameter specifies whether or not recording searches will be optimized. Specify 'Yes' to enable faster searches, with the following limitations:
 - The searched values will be matched as prefixes. For example, a search for "admin" will match "administrator" but will not match "123admin".
 - Only the first 50 characters of each search keyword are considered when trying to match them against the session properties.

The following **Recordings Displayed Columns** parameters define the columns displayed in the list of recordings as a result of the search process.

- The **SortBy** parameter specifies the name of the column by which to sort the recordings displayed in the search results. By default, the recordings are sorted by the Safe column.

The specified columns are properties of the password or recording. By default, the following columns can be specified to locate session recordings:

- | | | | |
|--------------------|---------------------|-----------------------|-----------------|
| ▪ File | ▪ PIMSuCWD | ▪ AccountUsernam
e | ▪ Duration |
| ▪ Safe | ▪ RemoteMachin
e | ▪ AccountAddress | ▪ VideoSiz
e |
| ▪ Folder | ▪ ClientApp | ▪ AccountPolicyID | ▪ TextSize |
| ▪ User | ▪ Protocol | ▪ Start | ▪ LockedBy |
| ▪ FromIP | ▪ AccountDetails | ▪ End | ▪ TicketID |
| ▪ PIMSuComman
d | | | |

The following parameters define each column that will be displayed:

- The **Name** parameter specifies the name of the property that will be displayed in this column.
- The **DisplayName** parameter specifies the title of the column that will be displayed. If this is not specified, the default property name will be displayed.
- The **Width** parameter specifies the width of the column in pixels.
- The **DataType** parameter specifies the type of information that will be displayed in the column. The data type can be a string, date, or image.
- The **Visible** parameter determines whether or not the column will be visible.

Viewing Recording Details

The **Recording Details** parameters, in the **Privileged Session Management UI** parameters, configure the Recording Details page.

- The **Toolbar Actions** parameters define the buttons that appear on the toolbar in the Recording Details page. You can specify the name of each button and whether or not it will be displayed.
- The **Recording Descriptor Properties** parameters define the recording properties that will comprise the display name of the PSM recordings. You can specify the name of each property and whether or not it will be displayed.
- The **Recording Details Properties** parameters define the recording properties that are displayed in the Recording Details page. You can specify the name of each property, the display name, and whether or not it will be displayed.
- The **Recording Details Password Properties** parameters define the properties of the password that was used during the recording session that will be displayed. You can specify the name of each property, the display name, and whether or not it will be displayed.
- The **Recording Details Tabs** parameters define the tabs that are displayed in the Recording Details page. For each tab, you can specify the following:
 - The **General** parameter, **ReportPeriod**, specifies the default number of days that will be included in the list.
 - The **Displayed Columns** parameters define the columns that will be displayed in the tab.
 - The **SortBy** parameter specifies the name of the column by which to sort the recordings in the tab. By default, the recordings are sorted by the Time column.

You can specify the following parameters for each column:

- The **Name** parameter specifies the name of the property that will be displayed in this column.
- The **DisplayName** parameter specifies the title of the column that will be displayed. If this is not specified, the default property name will be displayed.
- The **Width** parameter specifies the width of the column in pixels.
- The **DataType** parameter specifies the type of information that will be displayed in the column. The data type can be a string, date, or image.
- The **Visible** parameter determines whether or not the column will be visible.

Viewing Account Recordings

The **Account Details Session Recordings** parameters, in the **Privileged Session Management UI** parameters, define the columns that will be displayed in the Recordings tab in the Account Details page.

- The **SortBy** parameter specifies the name of the column by which to sort the recordings in the tab. By default, the recordings are sorted by the Safe column.

You can specify the following parameters for each column:

- The **Name** parameter specifies the name of the property that will be displayed in this column.
- The **DisplayName** parameter specifies the title of the column that will be displayed. If this is not specified, the default property name will be displayed.
- The **Width** parameter specifies the width of the column in pixels.
- The **DataType** parameter specifies the type of information that will be displayed in the column. The data type can be a string, date, or image.
- The **Visible** parameter determines whether or not the column will be visible.

Displaying Live Sessions

The following **Live Sessions Displayed Columns** parameters define the columns displayed in the list of live sessions.

- The **SortBy** parameter specifies the name of the column by which to sort the sessions. By default, the sessions are sorted by the Safe column.

The specified columns are properties of the password or sessions. By default, the following columns can be specified to locate live sessions:

- | | | | |
|----------|-----------------|-------------------|------------|
| ▪ File | ▪ FromIP | ▪ AccountDetails | ▪ Start |
| ▪ Safe | ▪ RemoteMachine | ▪ AccountUsername | ▪ TicketID |
| ▪ Folder | ▪ ClientApp | ▪ AccountAddress | |
| ▪ User | ▪ Protocol | ▪ AccountPolicyID | |

The following parameters define each column that will be displayed:

- The **Name** parameter specifies the name of the property that will be displayed in this column.
- The **DisplayName** parameter specifies the title of the column that will be displayed. If this is not specified, the default property name will be displayed.
- The **Width** parameter specifies the width of the column in pixels.
- The **DataType** parameter specifies the type of information that will be displayed in the column. The data type can be a string, date, or image.
- The **Visible** parameter determines whether or not the column will be visible.

Direct Playback in the PVWA

The **JumpOffset** parameter in the **Commands** parameters in the **Privileged Session Management UI** parameters, defines the time (in seconds) prior to the location of a selected command that the recording will begin to play.

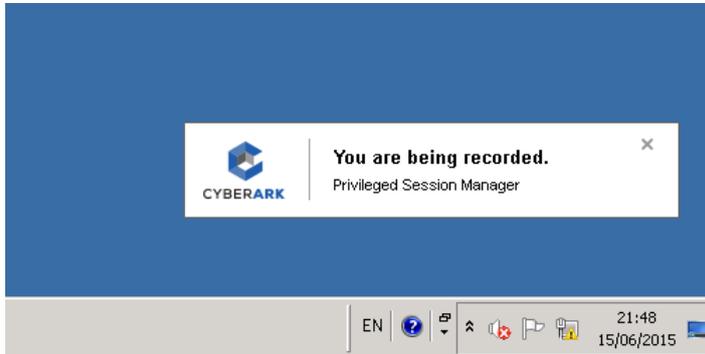
The **Streaming** parameters, in the **Privileged Session Management UI** parameters, define the embedded video player that is used to play PSM recordings directly in the PVWA.

Note: The embedded video player requires that Adobe Flash player 10.0 browser add-on or higher is installed on the end users' browser. For more information, refer to the Privileged Account Security System Requirements.

- The **Enabled** parameter determines whether or not authorized users can play recordings directly in the PVWA. If this parameter is set to **Yes**, users will be able to play recordings using an embedded video player. If this parameter is set to **No**, recordings will be downloaded and played using the default media player. The default value is **Yes**.
- The **Width** parameter determines the width of the embedded video player that is displayed. The default value is **800** pixels.
- The **Height** parameter determines the height of the embedded video player that is displayed. The default value is **600** pixels.
- The **AutoPlay** parameter determines whether direct playback will start automatically after selecting the recording to play, or whether the embedded video player will be displayed and the user will be able to start playback by clicking the Play button. The default value is **Yes**, indicating that the direct playback will start playing automatically.
- The **AllowFullScreen** parameter determines whether or not users will be able to expand the video display area to utilize the entire screen. The default value is **Yes**.
- The **AllowDownload** parameter determines whether or not users will still be able to download recordings as files when streaming is enabled. The default value is **Yes**.
- The **BufferSize** parameter specifies the size, in megabytes, that the video player will request from the server on each data request. The default value is **1** (one) megabyte.

Displaying Notifications when Sessions are Recorded

A notification can be displayed when a remote session is opened and the PSM starts recording it, and ensures that users know that their session is being recorded. This notification is displayed at the bottom right corner of the remote session window.



This notification can be configured with the following parameters in the Privileged Session Management parameters of each platform configured for PSM:

- **ShowRecordedSessionNotification** – This parameter determines whether or not a notification will be displayed when the PSM starts recording a remote session. The default value is **Yes**.
- **RecordedSessionNotificationDisplayTime** – This parameter determines the number of seconds that the recorded session notification is displayed. The default value is **5** seconds. If **0** (zero) is specified, the notification will not be closed automatically and will be displayed until the user closes it.

To Display a Notification when a Session is Recorded

1. Click **ADMINISTRATION** to display the **System Configuration** page, then click Platform Management to display a list of supported target account platforms.
2. Select the platform to configure, then click **Edit**; the settings page for the selected platform appears.
3. Expand **UI & Workflows**, and then right-click **Privileged Session Management**.
4. In the **Properties** list, specify a value for the **ShowRecordedSessionNotification** property.
5. Specify a value for the **RecordedSessionNotificationDisplayTime** property.
6. Click **Apply** to save and apply these configurations and stay in the platform settings page,

or,

Click **Save** to save these configurations and return to the System Configuration page.

These changes will be applied the next time the PSM refreshes the configuration, according to the value of the **ConfigurationRefreshInterval** parameter in the Privileged Session Management configuration.

Configuring Live Session Monitoring

The PSM enables authorized users to monitor live sessions from their own workstation, take part in controlling these sessions, and even terminate them.

The Live Session Monitoring Settings determine how users can monitor live privileged sessions and the types of activities that they can perform. In order to monitor live sessions, users require the following settings:

- **Monitoring Live Sessions**
 - Users require ownership of the Safe where the PSM recordings are stored (by default, in the **PSMRecordings** Safe) with the following permission:
 - Retrieve files/passwords
 - In the PWA system configuration, the **Live Sessions Monitoring** settings must specify the following:
 - **Monitoring** – The **AllowMonitor** property must be set to **Yes**
 - **Monitor level** – The **MonitoringLevel** property determines whether users can **view** or **control** live sessions.
- **Terminating Live Sessions**
 - Users require ownership of the **PSMRecordings** Safe with the following permission:
 - Retrieve files/passwords
 - Users require membership in one of the following groups:
 - **PSMLiveSessionTerminators** – The default group whose members can terminate live sessions.
 - Membership in one of the groups listed in the **TerminatingLiveSessions UsersAndGroups** parameter in the Live Sessions Monitoring settings.
 - In the PWA system configuration, the **Live Sessions Monitoring** settings must specify the following:
 - The **AllowTerminate** property must be set to **Yes**

To Configure Live Session Monitoring

1. Click ADMINISTRATION, then in the **System Configuration** page click **Options**; the Web Access Options are displayed.
2. Expand the **Privileged Session Management** parameters, then expand **General Settings**.
3. Expand the **Server Settings**, then expand the **Live Sessions Monitoring Settings**; the general Live Sessions Monitoring Settings properties are displayed in the Properties list.

4. Specify the following properties:
 - **AllowMonitor** – Permits authorized users to monitor live sessions. The exact monitoring task is specified in the **MonitoringLevel** parameter.
 - **MonitoringLevel** – Specifies the monitoring task that authorized users can perform. The following options are available:
 - **View** – Users can view live sessions from their own workstation, but cannot participate in the session.
 - **Control** – Users can participate in live sessions and can control them in the same way as the original user.
 - **AllowTerminate** – Permits authorized users to terminate live sessions.
5. Expand **Live Sessions Monitoring Settings**, and then expand **Terminating Live Sessions Users and Groups**; a list of Vault users and groups that are authorized to terminate live sessions is displayed.

By default, all members of the Vault group called **PSMLiveSessionTerminators** are authorized to terminate live sessions.

- Make sure that the users who will terminate live sessions belong to this group, or,
 - Create a new user or group that will be able to terminate live sessions:
 - i. Right-click **Terminating Live Sessions Users and Groups**, then from the pop-up menu select **Add User or Group**; a new User or Group parameter is added.
 - ii. In the new **User or Group** parameter, then in the Properties list specify the name of the user or group to authorize.
6. Click **Apply** to save the new parameter values and stay in the Web Access Options page, or,

Click **OK** to save them and return to the System Configuration page. The changes will be applied after the period of time specified in the **ConfigurationRefresh Interval** parameter.

Disabling and Enabling Live Session Monitoring

By default, live session monitoring is enabled at system level for all authorized users, and can be disabled at platform level. Live session monitoring can also be disabled at system level, but when it is disabled, it cannot be enabled at platform level.

To Enable\Disable Live Session Monitoring at System Level

1. Click ADMINISTRATION, then in the **System Configuration** page click **Options**; the Web Access Options are displayed.
2. Expand the **Privileged Session Management** parameters, then expand **General Settings**.
3. Expand the **Server Settings**, then expand the **Live Sessions Monitoring Settings**; the general Live Sessions Monitoring Settings properties are displayed in the Properties list.

4. In the Properties list, set the **Enable** parameter.
 - Set **Enable** to **No** to disable live session monitoring at system level. This cannot be overridden at platform level.
 - Set **Enable** to **Yes** to re-enable live session monitoring at system level.
5. Click **Apply** to save the new parameter values and stay in the Web Access Options page,
or,
Click **OK** to save them and return to the System Configuration page. The changes will be applied after the period of time specified in the **ConfigurationRefresh Interval** parameter.

Customizing Live Sessions at Platform Level

You can override live sessions monitoring settings in individual platforms, enabling you to determine whether or not authorized users can or cannot monitor live sessions during privileged sessions that use accounts managed by specific platforms, regardless of the general live sessions monitoring settings.

In order to monitor live sessions at platform level, users require the Safe ownership and permissions listed above in *Configuring Live Session Monitoring*, page 617.

To Manage Live Sessions at Platform Level

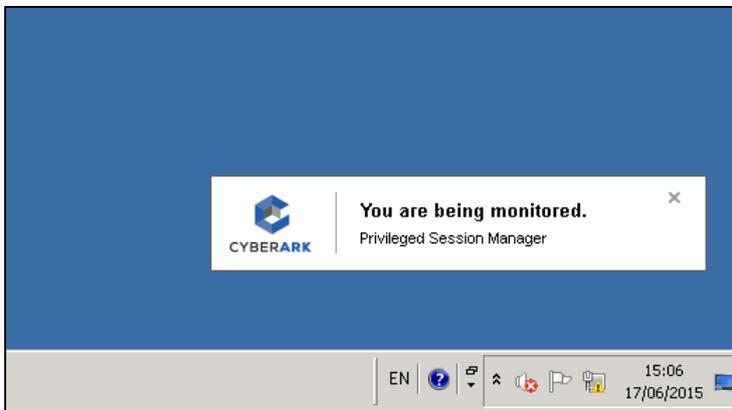
1. In the System Configuration page, click **Options**, then configure the general **Live Sessions Monitoring Settings**.
 2. Click **ADMINISTRATION** to display the **System Configuration** page, then click Platform Management to display a list of supported target account platforms.
 3. Select the platform to configure, then click **Edit**; the settings page for the selected platform appears.
 4. Expand **UI & Workflows**, and then right-click **Privileged Session Management**, then from the pop-up menu select **Add Override Live Sessions Monitoring Settings**; a set of parameters is added to the PSM parameters. These parameters enable you to set Live Session Monitoring settings for this platform which will override the general Live Session Monitoring settings.
 5. Select **Override Live Sessions Monitoring Settings** and set the following properties:
 - **AllowMonitor** – Whether or not authorized users can view and/or control live sessions that use accounts managed by this platform. The monitoring task level (View/Control) is taken from the general live sessions monitoring settings.
 - **AllowTerminate** – Whether or not authorized users can terminate live sessions that use accounts managed by this platform.
- Note:** When live session monitoring is disabled at system level, it cannot be enabled at platform level.

6. Click **Apply** to save the new parameter values and stay in the platform settings page,
or,
Click **OK** to save them and return to the System Configuration page. The changes will be applied after the period of time specified in the **ConfigurationRefresh Interval** parameter.

Displaying a Live Monitoring Notification

When authorized users start monitoring a live session, a notification can be displayed on the screen to indicate that the session is being monitored in real-time. This is configured separately for each platform.

This notification is displayed at the bottom right corner of the remote live session window.



To Display a Live Monitoring Notification

1. Click **ADMINISTRATION** to display the **System Configuration** page, then click Platform Management to display a list of supported target account platforms.
2. Select the platform to configure, then click **Edit**; the settings page for the selected platform appears.
3. Expand **UI & Workflows**, and then select **Privileged Session Management**.
4. In the Properties list, set the following properties:
 - **ShowLiveMonitoringNotification** – Whether or not to display an alert during a privileged session indicating that this session is being monitored live. The default value is **Yes**.
 - **LiveMonitoringNotificationDisplayTime** – Time in seconds to display the alert during live sessions, indicating that this session is being monitored. Specify '0' (zero) to display it indefinitely. The default value is **5** seconds.
5. Click **Apply** to save the new values and stay in the platform settings page,
or,
Click **OK** to save them and return to the System Configuration page. The changes will be applied after the period of time specified in the **ConfigurationRefresh Interval** parameter.

Configuring PSM Connections

The connections for privileged sessions performed through the PSM can be configured in the PVWA. Each connection is defined by the following types of parameters:

- **User parameters** – Parameters that determine the information that users will be required to supply while initiating a PSM connection.
- **Component parameters** – Parameters that define PSM connections and how each connection will be performed. These parameters can be configured at different levels to affect connections at system, platform, or account level. For more information, refer to *Configuring PSM Components*, below.
- **Target settings** – Parameters that define target machines for each connection.

During PSM installation, a series of supported PSM connections are created. You can use these connections with the default settings, or you can customize them by changing existing parameters or adding additional parameters manually.

Configuring PSM Components

Connection components can be configured at the following levels:

- **General** – Parameters set at general PSM connection level are applied to every PSM connection. For more information, refer to *Configuring General PSM System Parameters*, page 622.

In addition to the general parameters that are applied to all PSM connections, you can configure general parameters for specific PSM components. These parameters enable you to customize connections according to your enterprise needs and standards. For more information about the available features and parameters, refer to the relevant section:

- For details about configuring PSM for server connections, refer to *Configuring PSM for Server Connections*, page 624.
- For details about configuring PSM for database connections, refer to *Configuring PSM Connections for Database Connections*, page 664.
- For details about configuring PSM for virtualization connections, refer to *Configuring PSM Connections for Virtualization Connections*, page 666.
- **Platform** – Some of the PSM connection parameters can be set at platform level, overriding some of the general connection settings and enabling you to customize connections that are associated with a specific platform. For more information and a list of parameters that can be configured at platform level, refer to *Configuring Platform Parameters*, page 623.
- **Account** – Some of the PSM connection parameters can be set at account level, overriding some of the general connection settings as well as platform level settings. This enables you to customize PSM connections for specific accounts. For more information, refer to *Configuring Account Parameters*, page 624.

Before Configuring PSM Connections to Windows Machines

- Configure the remote machine to allow remote connections.

Before Configuring PSM Connections for VMWare vCenter Personal Accounts

- Make sure that end users cannot access the vCenter directly, but are forced to access it through the PSM.

Configuring General PSM System Parameters

The general PSM system parameters in Web Access Options define PSM connections, and determine how each connection will be performed. By default, all the connection components are configured. You can view them in the Web Access Options.

1. Click ADMINISTRATION, then in the System Configuration page click **Options**; the Web Access Options are displayed.
2. Click **Connection Components**, and expand the connection component to configure.
3. Click **User parameters** to display parameters that prompt users for more information.
4. Click **Target Settings** to display parameters that define specific target machine settings.
5. Some parameters are defined automatically during installation and others can be added manually. Target Settings can be specified in the Client Specific section.
 - i. Right-click on **User parameters** or **Client Specific**, then select **Add Parameter**; a new parameter is added.
 - ii. Specify the name and value of each property.

The general Connection Components parameters that are common to all PSM connection components are described in the following table:

Parameter	Description
Connection Component (root level)	
ID	A unique ID that identifies the connection component. The values of these settings are configured during installation.
FullScreen	Whether or not the remote desktop window will be opened in full screen mode. The full screen mode opens a new window with an additional window for logon. You can toggle between screen modes with Alt+Ctrl+Break. Default value: No.
Height	The height in pixels of the desktop resolution on the remote machine. The height of the window that is opened on the remote desktop is calculated from this parameter. Default value: 768 pixels.
Width	The width in pixels of the desktop resolution on the remote machine. The width of the window that is opened on the remote desktop is calculated from this parameter. Default value: 1024 pixels.

Parameter	Description
Type	<p>Specifies the interface that is used for the connection.</p> <ul style="list-style-type: none"> The default value for privileged SSO connections is: <code>CyberArk.PasswordVault.TransparentConnection.PSM.PSMConnectionComponent, CyberArk.PasswordVault.TransparentConnection.PSM</code> The default value for windows transparent connections is: <code>CyberArk.PasswordVault.TransparentConnection.PSM.PSMConnectionComponent, CyberArk.PasswordVault.TransparentConnection.PSM.PSM</code>
Display Name	Defines the display name of the connection component.

- Click **Apply** to save the new configurations.

Configuring Platform Parameters

The platform parameters can be configured to override transparent connection configurations set at general system level (in the previous step).

- Click **ADMINISTRATION** to display the **System Configuration** page, then click Platform Management to display a list of supported target account platforms.
- Select the platform to configure, then click **Edit**; the settings page for the selected platform appears.
- Expand **UI & Workflows**, and then expand **Connection Components**.
- Select a connection component to display the parameters for the connection component to configure.
- Click **Override Component Parameters** to display parameters that override the general component parameters.
- To define a subsection of override parameters, right-click on the component to configure, then select **Add Override Component Parameters** or **Add Override target settings**; a new section for these override parameters is created.
- To add parameters to these sections, right-click the name of the section, then select **Add Parameter**; a new parameter is added.
- Specify the name and value of each parameter.

The general platform Connection Components parameters are described in the tables below.

- General parameters** – The following parameters can be specified in each platform for connection components:

Parameter	Description	Override at account level
Component Parameters		
AllowTransparent Connection	Whether or not the user will be able to connect transparently to a remote machine.	x

Parameter	Description	Override at account level
AllowViewing Passwords	Whether or not the user will be able to view and copy the password. Default value: Yes. Users who have the “Manage Safe” authorization can view the password, regardless of this parameter setting. Note: If AllowTransparentConnection is set to ‘No’, this parameter will be set to ‘Yes’ automatically.	✘
PSMConnection Default	Defines the default connection component for connections via PSM.	✘
EnforceTransparent ConnectionInDual Control	Whether or not transparent connections will be limited in dual control.	✘

9. Click **Apply** to save the new configurations.

Configuring Account Parameters

1. Add the account that will be used to log onto the remote device transparently, or,
In the Accounts Details page of the account that will be used to log onto the remote device transparently, click **Edit**.
2. In the Optional Properties section, specify the parameters that are relevant to this connection component. A detailed list and explanations of all the available parameters are in the tables on the following pages.

Configuring PSM for Server Connections

The following PSM connections for servers can be customized to meet your specific requirements:

- **PSM for Windows sessions (PSM-RDP)** – These connections can be configured to provide the following features:
 - **Securing connections to target machines with SSL** – PSM-RDP connections can verify the target machine before connecting to it and encrypt the session, using an SSL connection. For more information, refer to *Securing RDP Connections to Target Machines with SSL*, below.
 - **Running Specific Commands on RDP Connections** – PSM-RDP connections can start by launching a dedicated program on the target machine. For more information, refer to *Running Specific Commands on RDP Connections*, page 629.

- **PSM for Windows and Unix sessions (PSM-RDP and PSM-SSH)** – These connections can be configured to provide the following features:
 - **Configuring Multiple Target Addresses** – The PVWA can be configured to display multiple target addresses for users to select from when they create a request or connect to a remote machine transparently. For more information, refer to *Configuring Multiple Target Addresses*, page 634.
- **PSM for Unix and Telnet sessions (PSM-SSH and PSM-Telnet)** – These connections can be configured to provide the following features:
 - **Configure logon accounts that elevate users to the role of privileged user** – When the root user is forbidden from logging on directly, a logon account in the Vault can be associated with a Unix session to logon to the remote machine and then elevate itself to the role of the privileged user. For more information, refer to *Using Logon Accounts for PSM-SSH and PSM-Telnet Connection Components*, page 647.
- **PSM for Telnet sessions (PSM-Telnet)** – These connections can be configured to provide the following features:
 - **Automatic login sequence** – The PVWA can be configured to initiate an automatic login sequence using dynamic values to log on to a remote machine using Telnet. For more information, refer to *Configuring PSM-Telnet Connection Components*, page 645.
- **PSM for Unix File Transfer (WinSCP) sessions (PSM-WinSCP)** – These connections can be configured to provide the following features:
 - **Connection through a CLI** – PSM-WinSCP connections can be initiated through a CLI to integrate with your enterprise workflow. For more information, refer to *Using WinSCP through a CLI*, page 654.
- **PSM for AS400 (iSeries) sessions (PSM-AS400)** – These connections can be configured to provide the following feature:
 - **Customizing AS400 (iSeries) emulation parameters** – PSM-AS400 connections can be customized to support command line arguments for the wc3270 emulation that is used by the PSM. For more information, refer to *Customizing AS400 (iSeries) and OS/390 (Z/OS) Emulation Parameters*, page 635.
- **PSM for OS390 (Z/OS) sessions (PSM-OS390)** – These connections can be configured to provide the following feature:
 - **Customizing OS390 (Z/OS) emulation parameters** – PSM-OS390 connections utilize the same wc3270 emulation as PSM-AS400 connections and are configured in the same way. For more information, refer to *Customizing AS400 (iSeries) and OS/390 (Z/OS) Emulation Parameters*, page 635.

Securing RDP Connections to Target Machines with SSL

Users can configure secure PSM-RDP connections to target machines by verifying the target machine before connecting to it and encrypting the session, using an SSL connection. To facilitate this type of connection, the target machine must have its own certificate. The PSM server machine must trust the CA that signed the certificate used by the target machine.

▪ Before Configuring Secure RDP Connections with SSL

- Import the CA Certificate that signed the certificate used by the target machine into the Windows certificate store on the PSM server machine:
 - Certificates (Local Computer)/Trusted Root Certification Authorities
 By storing the certificate in this location, all users will be able to access the remote machine using an authenticated connection.

▪ To Configure Secure RDP Connections with SSL

1. In the System Configuration page, in the Web Access section, click **Options**, then select **Connection Components**; the connection component parameters that define target addresses are displayed in the properties list.
2. Expand the PSM-RDP connection component, and then expand the **Target Settings**.
3. Right-click **Client Specific**, then in the pop-up menu select **Add Parameter**; a new parameter is added to the list of client specific parameters.
4. In the parameter properties, specify the following:
 - **Name** – The name of the client specific parameter. Specify **AuthenticationLevel**.
 - **Value** – The authentication level that will be used for this connection. Specify any of the following values:

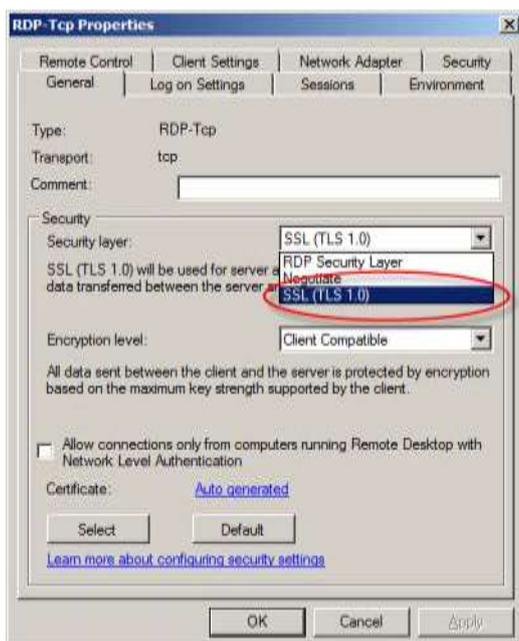
Value	Description
0	The PSM server is not required to authenticate the target machine before connecting to it.
1	The PSM server will authenticate the target machine before connecting to it.
2	The PSM server will authenticate the target machine before connecting to it. If the authentication fails, the user will be able to cancel the connection or to initiate a connection without authentication.

5. Click **Apply** to apply the new Connection Component configurations, or, Click **OK** to save the new Connection Component configurations and return to the System Configuration page.

Securing RDP Connections to the PSM Machine with SSL

Users can configure secure RDP connections to the PSM machine using an SSL connection.

1. On the PSM server, set the security layer:
 - i. On the PSM server, open the Remote Desktop Session Host Configuration.
 - ii. In the Connections list, double-click the **RDP-Tcp** connection; the RDP-Tcp Properties window is displayed.
 - iii. From the security layer drop-down box, select **SSL (TLS 1.0)**.



- iv. Click **OK**.
2. In the PVWA, update all the active connection components to enable RDP over SSL connections to the PSM machine. For example, for PSM SSH connections, update PSM-SSH.

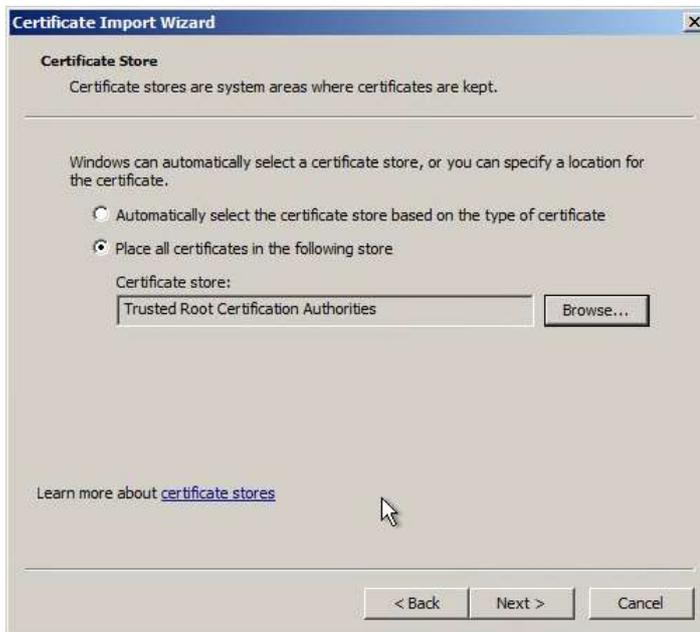
Note: To support Live Session connections, update the PSM-MonitorSession connection component.

- i. Log onto the PVWA as an administrative user.
- ii. In the System Configurations page, click **Options**, then expand the **Connection Components**.
- iii. In each active connection component, add a new **Component Parameter**.
- iv. In the Component Parameter properties, add a new parameter with the following values:
 - **Name** – The name of the component parameter.
 - For connections with ActiveX or an external tool, specify **AdvancedSettings4.AuthenticationLevel**.
 - For connections with RDP files, specify **authentication level:i**.
 Add both parameters to use both methods.
 - **Value** – The value of this parameter name. Specify 1.
- v. Click **Apply** to apply the new configurations and stay in the Options page.

3. Connections to the PSM require a certificate on the PSM machine. By default, Windows generates a self-signed certificate, but you can use a certificate that is supplied by your enterprise.

In the Privileged Session Management parameters, make sure that the PSM address specifies the exact common name of the certificate.

- i. Expand the **Privileged Session Management** parameters and then expand **Configured PSM Servers**.
 - ii. Expand **Connection Details**, and select **Server**; the Server Properties are displayed.
 - iii. In the Address property, specify the certificate common name.
 - iv. Click **Apply** to apply the new configurations,
or,
Click **OK** to save the new configurations and return to the System Configuration page.
4. On the Client machines, make sure that the PSM machine certificate is signed by a trusted CA.



Running Specific Commands on RDP Connections

Users can configure a PSM-RDP connection component to start the connection by launching a dedicated program on the target machine. When this dedicated program is closed the RDP session is closed as well.

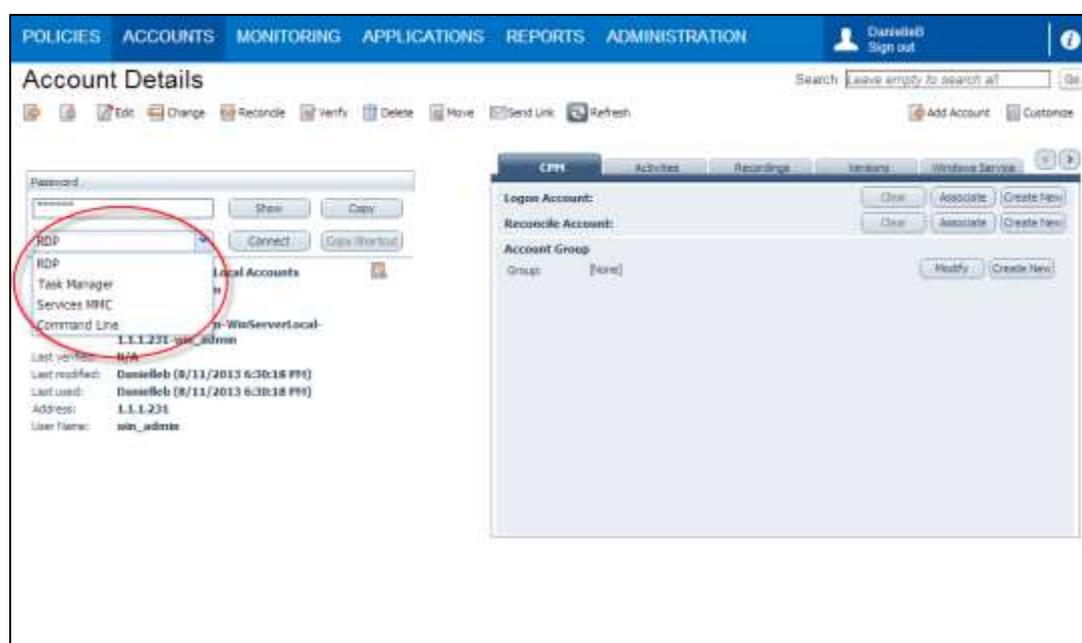
This feature is stored on the following server platforms:

- Windows 2003
- Windows 2008/2008R2 with Remote Desktop Services (Terminal Services)

Note: Make sure that the Remote Desktop Session Host role is installed.

Make sure you have the required number of RDS CALs to enable you to access the RDS server. For more information, refer to *Microsoft Remote Desktop Services (Terminal Services) License* in the Privileged Account Security Installation Guide.

The following example shows a drop-down list of connections have been configured to enable accounts to initiate three specific commands on a remote machine.



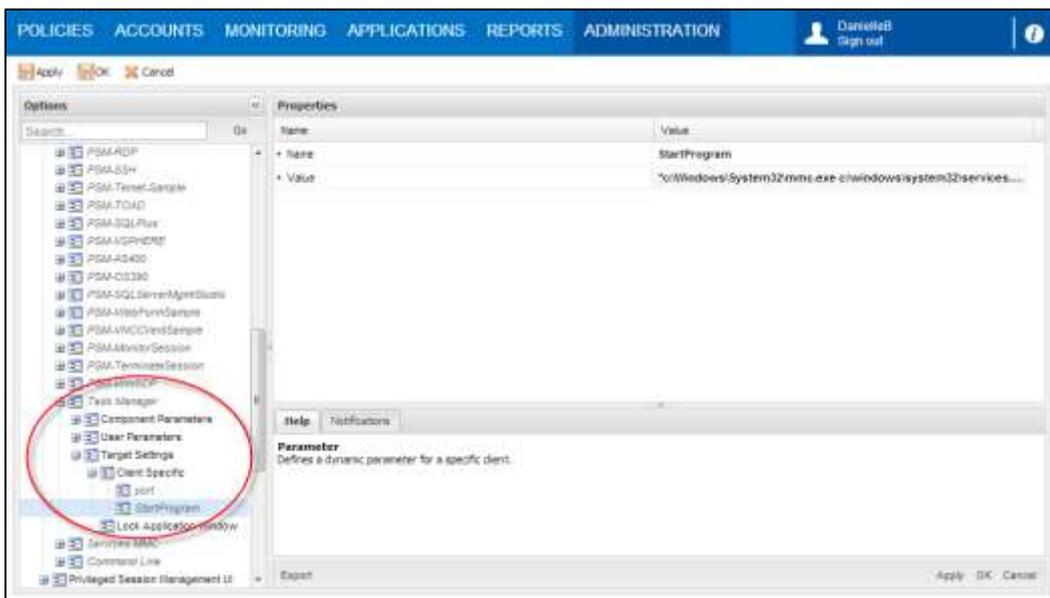
To Configure RDP Connections to Start with a Specific Command

1. In the System Configuration page, in the **Component Settings** section, click **Options**, then select **Connection Components**; the connection component parameters that define target addresses are displayed in the properties list.
2. Right-click the **PSM-RDP** connection component, then from the pop-up menu select **Copy**.
3. Right-click **Connection Components**, then from the pop-up menu select **Paste Connection Component**; a copy of the PSM-RDP connection component is added to the list of configured connection components with its original ID (PSM-RDP).

4. Configure the new connection component to run a specific command:
 - i. In the list of properties, specify a new ID that reflects the purpose of the connection. This ID will be displayed in the drop-down list of available connections in the Accounts Details page and must be self-explanatory.
 - ii. Expand the connection component, then expand the **Target Settings**.
 - iii. Right-click **Client Specific**, then in the pop-up menu select **Add Parameter**; a new parameter is added to the list of client specific parameters.
 - iv. In the parameter properties, specify the following:
 - **Name** – The name of the client specific parameter. Specify **StartProgram**.
 - **Value** – The full path of the executable program that will be started when the connection is initiated.

To specify a file that is not an executable, specify the executable that is used in order to execute this application. For example, to specify 'msc', you would specify "**c:\Windows\System32\mmc.exe c:\windows\system32\services.msc**".

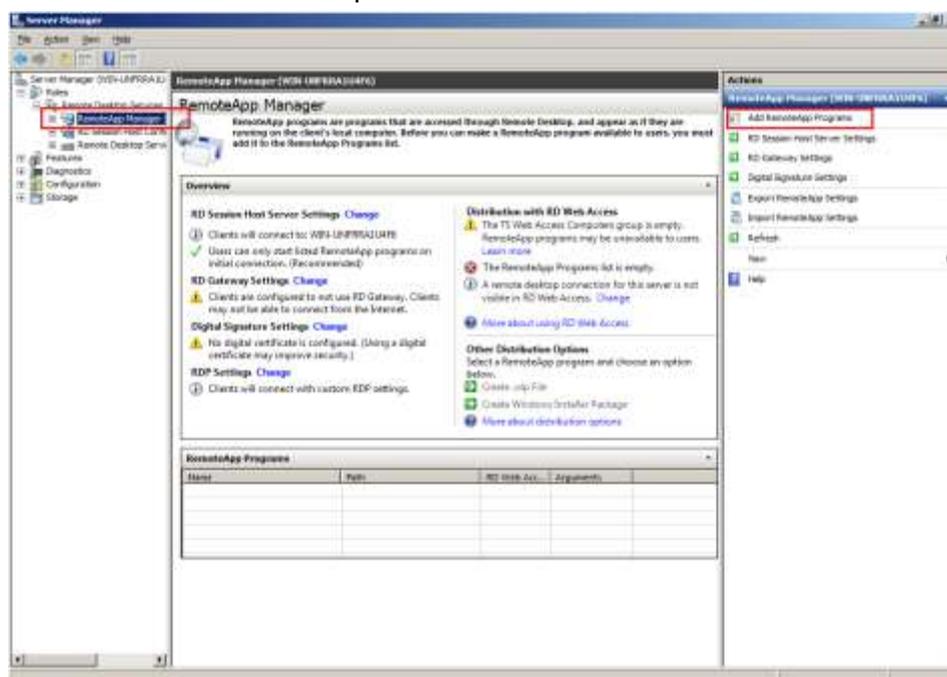
The following example shows the configuration for a connection component that will start the task manager on the remote machine when the connection is made. This connection component specifies the full path of the task manager executable.



5. To specify a working directory for the selected application, usually required to ensure the correct resolution of any relative filenames, create another new Client Specific parameter and specify the following:
 - **Name** – The name of the client specific parameter. Specify **WorkDir**.
 - **Value** – The full path of the working directory for the program specified in the StartProgram parameter. If this property is not specified, the default working directory will be used. The default working directory is **C:\Users*current user***.

Note: Do not include quotation marks or a trailing slash.
6. Click **OK** to save the new Connection Component configurations and return to the System Configuration page.

7. Click **ADMINISTRATION** to display the **System Configuration** page, then click Platform Management to display a list of supported target account platforms.
8. Select the platform to configure, then click **Edit**; the configuration page for the selected platform appears.
9. Expand **UI & Workflows**, and right-click **Connection Components**, then from the pop-up menu select **Add Connection Component**; a new connection component is added to the current list of connection components that are configured for this platform.
10. In the new Connection Component, specify the following properties:
 - **Id** – The unique ID that identifies the connection component you created previously for the specific command.
 - **Enable** – Whether or not this connection component will be enabled for this platform. Specify **Yes**.
11. Click **Apply** to apply the new platform configurations, or, Click **OK** to save the new configurations and return to the System Configuration page.
12. On the target machine, make sure of the following:
 - The application that will be opened by the connection is included in the RemoteApp Manager Programs List.
 - The target machine must be configured to allow an initial startup program by the Remote Desktop Services.



Configuring Domain Accounts

You can use the PSM to access target machines using Windows Domain accounts or UNIX Domain/NIS accounts.

Note: In SSH protocol, there is no foolproof way to ensure the identity of the target machine, which could potentially lead to a security risk. Please take this into consideration when using this feature.

Configuring Domain Platforms

Create a platform that determines how domain accounts will be managed.

- **Windows Domain Platforms** – Use the predefined Windows Domain platform. For general information about managing platforms, refer to *Managing Target/Service Account Platforms*, page 108.
- **UNIX Domain/NIS Platforms** – There is no predefined platform to manage UNIX Domain/NIS accounts, and it must be configured manually as described below.
 1. Click **ADMINISTRATION** to display the **System Configuration** page, then click **Platform Management** to display a list of supported target account platforms.
 2. Select an existing SSH platform that is similar to the new target account platform. For example, Unix via SSH.
 3. Click **Duplicate**; the Duplicate Platform window appears.
 4. Type the name and a description of the new platform, then click **Save & Close** to create the new platform.
 5. Select the new target account platform, and then click **Edit**; the configuration page for the selected platform appears.
 6. Expand **UI & Workflows**, and then expand **Connection Components**; the Connection Components parameters are displayed with their default values.
 7. Right-click **PSM-SSH** and select **Add Override User Parameters**; a new set of parameters is added.
 8. Right-click **Add Override User Parameters** and select **Add Parameter**; a new parameter is added.
 9. Select the new parameter then, in the Properties list, set the following properties.

Property	Description
Name	The name of the parameter. Specify PSMRemoteMachine .
Visible	Whether or not the user will be prompted for this parameter before the connection is established. Specify Yes .
Type	The type that will be used to modify the appearance or behavior of a parameter UI field. Specify the following: CyberArk.PasswordVault.Web.TransparentConnection.RemoteMachineUserParameter, CyberArk.PasswordVault.Web Note: Do not change this value.
Required	Whether or not the user is required to provide this information for the remote connection to be activated. Specify Yes .
EnforceInDualControlRequest	Whether or not the user will be required to provide information in order to create a dual control request. Specify No .

10. Change any additional parameter values and/or add new values to define the new platform.
11. Click **Apply** to save the new configurations and apply them immediately,
or,
Click **OK** to save the new configuration and return to the System Configuration page.

Adding a Windows Domain or UNIX Domain/NIS Account

1. In the Add Account page, add the domain account that will be used to access the target account. Specify the following account properties:
 - **Platform Name** – Select the platform that you created for the domain account in the previous section.
 - **Address** – Specify the IP address or DNS of the domain server in the domain where the target machine resides.
2. Specify additional required and optional account properties.

For more information about adding accounts, refer to *Adding Accounts*, page 122.

For specific information about the information required to add Windows Domain accounts, refer to *Windows Domain Accounts*, page 141.

For specific information about the information required to add Unix Domain/NIS accounts, refer to *Unix Domain/NIS Accounts*, page 140.

Customizing Connection History to Target Machines

You can configure connection history to target machines by customizing connection component settings for PSM sessions. You can either use the predefined settings or customize them to meet your specific requirements.

1. Click **Options** to display the Web Access Options parameters, then select **Connection Components** to set the connection history.
2. Define the following parameters:
 - **EnableConnectAddressHistory** – Determines whether or not a list of addresses accessed with the selected account will be displayed in the Connect with Account window. The default value is **Yes**.
 - **MaxConnectHistory** – Defines the maximum number of remote machine addresses that can be displayed in the Connect with Account window. The address history is saved per account for each PWWA user. The default value is **7** addresses.
 - **MaxConnectAccountsNumber** – Defines the maximum number of accounts whose machine addresses history will be displayed in the Connect with Account window. The default value is **20** accounts.

3. In the **Privileged Account Request** parameters, define the following parameters:
 - **AddressSeparatorCharacter** – Defines the separator between addresses for remote connections.
 - **AnyAddressCharacter** – Defines the character that will represent “all addresses” in dual control requests.
4. Click **Apply** to apply the new Connection Component configurations,
or,
Click **OK** to save the new Connection Component configurations and return to the System Configuration page.

Configuring Multiple Target Addresses

For scenarios where you would like to access multiple targets using the same account, without using a domain account, you can configure the PWWA to display multiple target addresses for users to select from when they create a request or connect to a remote machine transparently.

Note: This can be configured for Windows and Unix accounts.

To Configure a Platform for Multiple Target Addresses

1. Click **ADMINISTRATION** to display the System Configuration page, then click **Platform Management** to display a list of supported target account platforms.
2. Select the platform that will manage accounts used to access the remote machines that will be displayed in the multiple targets list, then click **Edit**; the settings page for the selected platform appears.
3. Expand **UI & Workflows**, and then expand **Connection Components** and select the component to configure.
4. Display the component's **Override User Parameters**.
5. In the **PSMRemoteMachine** parameters specify the following text in the **Type** property:

```
CyberArk.PasswordVault.Web.TransparentConnection.RemoteMachineUser  
Parameter, CyberArk.PasswordVault.Web
```

6. Click **OK** to save the changes and return to the main System Configuration page.

Customizing AS400 (iSeries) and OS/390 (Z/OS) Emulation Parameters

Users can customize PSM-AS400 and PSM-OS390 connections by specifying command line arguments for the WC3270 emulation that is used by the PSM.

- **To Customize Connections for WC3270 Command Options**
 1. In the System Configuration page, click **Options** to display the Web Access Options parameters, then select **Connection Components**; the connection component parameters that define target addresses are displayed in the properties list.
 2. Expand the connection component to customize, and then expand the **Target Settings**.
 3. Right-click **Client Specific**, then in the pop-up menu select **Add Parameter**; a new parameter is added to the list of client specific parameters.
 4. In the parameter properties, specify the following:
 - **Name** – The name of the client specific parameter. Specify **CommandLineArguments**.
 - **Value** – The command line arguments that will be sent to the wc3270 client at the start of the connection session.
When you specify the value of the CommandLineArguments parameter, use the following guidelines:
 - Parameters that contain spaces must be enclosed in quotation marks.
 - To specify an option that creates files on the file system, make sure that the user running the PSM connection has permissions for the specified folder. For example, a command that creates a new file.
 - To run a command that manages the session file, specify the full path of the session file. In addition, do not specify the hostnames in the session file as the PSM server uses the hostname specified in the managed account.

The following table lists the command options that are supported by the PSM. For more information about the wc3270 options, refer to the wc3270 manual at <http://x3270.bgp.nu/wc3270-man.html>.

Command option	Description
-allbold	Forces all characters to be displayed using the 'bold' colors (colors 8 through 15, rather than colors 0 through 7). This helps with PC console windows in which colors 0 through 7 are unreadably dim. All-bold mode is the default for color (3279) emulation, but not for monochrome (3278) emulation.
-charset name	Specifies an EBCDIC host character set. For more information about the available character sets, refer to in the wc3270 manual.
-clear toggle	Sets the initial value of toggle to false. For a full list of available toggle names, refer to the wc3270 manual.
-model name	The model of 3270 display to be emulated. The model name is in two parts, either of which may be omitted: <ul style="list-style-type: none"> ▪ The first part is the base model, which is either 3278 or 3279. 3278 specifies a monochrome (green on black) 3270 display; 3279 specifies a color 3270 display.

- The second part is the model number, which specifies the number of rows and columns. Model 4 is the default.

Model Number	Columns	Rows
2	80	24
3	80	32
4	80	43
5	132	27

Note: Technically, there is no such 3270 display as a 3279-4 or 3279-5, but most hosts seem to work with them anyway.

The default model is 3278-4.

-mono	Prevents wc3270 from using color, ignoring any color capabilities reported by the terminal.
-noprompt	Disables command-prompt mode.
-oversize colsxrows	Makes the screen larger than the default for the chosen model number. This option has effect only in combination with extended data stream support (controlled by the "wc3270.extended" resource), and only if the host supports the Query Reply structured field. The number of columns multiplied by the number of rows must not exceed 16383 (3fff hex), the limit of 14-bit 3270 buffer addressing. It can also be specified as auto, which causes wc3270 to fill the entire terminal or console window.
-port n	Specifies a different TCP port to connect to. N can be a name from /etc/services like telnet, or a number. This option changes the default port number used for all connections. (The positional parameter affects only the initial connection.)
-proxy type:host[:port]	Causes wc3270 to connect via the specified proxy, instead of using a direct connection. The host can be an IP address, DNS, or hostname. The optional port can be a number or a service name. For a list of supported proxy types, refer to the wc3270 manual.
-reconnect	Causes wc3270 to automatically reconnect to the host if it ever disconnects. This option has effect only if a hostname is specified on the command line.
-S	Runs wc3270 in auto-shortcut mode. Wc3270 will create a temporary shortcut (.LNK file) that matches the parameters in the session file (model number, character set, etc.) and re-run itself from the shortcut.
+S	Disables auto-shortcut mode. It is generally a good idea to put this option on the command lines of all shortcuts, to avoid infinite looping.
-scriptport port	Causes wc3270 to listen for scripting connections on local TCP port port.
-set toggle	Sets the initial value of toggle to true. For a complete list of toggle names, refer to the wc3270 manual.

	The -p option of x3270if causes it to use this socket, instead of pipes specified by environment variables.
-title text	Sets the console window title to text, overriding the automatic setting of the hostname and the string wc3270.
-tn name	Specifies the terminal name to be transmitted over the telnet connection. The default name is IBM-model_name-E, for example, IBM-3278-4-E. Some hosts are confused by the -E suffix on the terminal name, and will ignore the extra screen area on models 3, 4 and 5. Prepending an s: on the hostname, or setting the "wc3270.extended" resource to "false", removes the -E from the terminal name when connecting to such hosts. The name can also be specified with the "wc3270.termName" resource.
-trace	Turns on data stream and event tracing at startup. The default trace file name is x3trc.process_id.txt in the wc3270 Application Data directory.
-tracefile file	Specifies a file to save data stream and event traces into.
-tracefilesize size	Places a limit on the size of a trace file. If this option is not specified, or is specified as 0 or none, the trace file will be unlimited. If specified, the trace file cannot already exist, and the (silently enforced) minimum size is 64 Kbytes. The value of size can have a K or M suffix, indicating kilobytes or megabytes respectively.
-v	Display the version and build options for wc3270 and exit.
-xrm "wc3270.resource: value"	Sets the value of the named resource to value. Resources control less common wc3270 options. For a full list of resources, refer to the wc3270 manual.

- Click **Apply** to apply the new Connection Component configurations, or, Click **OK** to save the new Connection Component configurations and return to the System Configuration page.

Changing Keyboard Mapping

Connections that are configured to support Command Line Arguments can use a keymap file that enables you to specify your own keymaps and customize your keyboard functions.

The **-keymap** option or the **wc3270.keymap** resource allows a custom keymap to be specified. If the option **--keymap xxx** is given (or the **wc3270.keymap** resource has the value **xxx**), **wc3270** will look for a resource named **wc3270.keymap.xxx**. If no resource definition is found, it will look for a file named **xxx.wc3270km**.

Multiple keymaps may be specified by separating their names with commas. Definitions in later keymaps supercede those in earlier keymaps.

In addition, separate keymaps may be defined that apply only in 3270 mode or NVT mode. For example, the resource definition **wc3270.keymap.xxx.nvt** or the file **xxx.nvt.wc3270km** will augment the definition of keymap **xxx** in NVT mode. Similarly, the resource definition **wc3270.keymap.xxx.3270** or the file **xxx.3270.wc3270km** will augment the definition of keymap **xxx** in 3270 mode.

Each line (rule) in a keymap specifies actions to perform when a particular key or sequence of keys is pressed. Keymap rules have the following syntax:

```
[modifier...]<Key>key...: action[(param[,...])] ...
```

The optional **Shift**, **Alt** or **Ctrl** modifiers specify that the **Shift**, **Alt** and **Ctrl** keys are pressed along with the specified **key**, respectively. The **LeftCtrl**, **RightCtrl**, **LeftAlt**, and **RightAlt** modifiers specify a particular **Ctrl** or **Alt** key. The **Enhanced** modifier is also available; **Enhanced <Key>ENTER** is the keypad **Enter** key. **Key** is either an ISO 8859-1 symbol name, such as **equal** for '=' and **a** for 'a', or a symbolic Windows key name, such as **UP**. More than one **key** can be specified, indicating that a sequence of keys must be pressed in order for the rule to be matched. The **action** is an action from the actions list. More than one **action** may be specified; they will be executed in order. For more a full list of actions, refer to <http://x3270.bgp.nu/wc3270-man.html#Actions>.

Keymap entries are case-sensitive and modifier-specific. This means that a keymap for the **b** key will match only a lowercase **b**. Actions for uppercase **B**, or for **Alt-b** or **Control-B**, must be specified separately.

Available symbolic key names are: **ADD**, **ALT**, **APPS**, **BACK** (Backspace), **CLEAR**, **CTRL**, **DECIMAL**, **DELETE**, **DIVIDE**, **DOWN**, **END**, **Enter** (alias for RETURN), **ESCAPE**, **EXECUTE**, **F1**, **F2**, **F3**, **F4**, **F5**, **F6**, **F7**, **F8**, **F9**, **F10**, **F11**, **F12**, **F13**, **F14**, **F15**, **F16**, **F17**, **F18**, **F19**, **F20**, **F21**, **F22**, **F23**, **F24**, **HELP**, **HOME**, **INSERT**, **LEFT**, **LMENU**, **LWIN** (Left Windows key), **MULTIPLY**, **NEXT** (Page Down), **NUMLOCK**, **NUMPAD0**, **NUMPAD1**, **NUMPAD2**, **NUMPAD3**, **NUMPAD4**, **NUMPAD5**, **NUMPAD6**, **NUMPAD7**, **NUMPAD8**, **NUMPAD9**, **PageUp** (alias for PRIOR), **PageDown** (alias for Next), **PAUSE**, **PRINT**, **PRIOR** (Page Up), **RETURN** (Enter), **RIGHT**, **RMENU**, **RWIN** (Right Windows key), **SCROLL**, **SELECT**, **SEPARATOR**, **SHIFT**, **SLEEP**, **SNAPSHOT**, **SUBTRACT**, **TAB** and **UP**.

CyberArk supplies a keymap file that is required for wc3270 on AS400 (iSeries) machines. Copy this file to the PSM server machine, and specify the name of this keymap file in the CommandLineArguments. For example **-keymap "PSMCustomFnKeyMap"**, as shown in the example above. For more information about the keymap file, refer to the wc3270 manual at <http://x3270.bgp.nu/wc3270-man.html#Keymaps>.

Configuring Connection Component Parameters

- **Windows sessions (PSM-RDP)** – The following parameters are specific to Windows RDP connection components. These are in addition to the general parameters that are common to all connection components:

Parameter	Description	Override at platform level	Override at account level
Connection Component (root level)			
EnableWindowScrollbar	Whether or not scrollbars will be added to the transparent connection window. Default value: No.	x	x
Component Parameters			
<ul style="list-style-type: none"> ▪ For RDP File connections – redirectclipboard:i ▪ For ActiveX connections – AdvancedSettings.RedirectClipboard 	Whether or not users will be able to redirect the clipboard from their local machine to the remote server. Possible values: <ul style="list-style-type: none"> ▪ 0 – Users will not be able to redirect the clipboard. ▪ 1 – Users will be able to redirect the clipboard. This is the default value. 	✓	x
<ul style="list-style-type: none"> ▪ For RDP File connections – redirectprinters:i ▪ For ActiveX connections – AdvancedSettings.RedirectPrinters 	Whether or not users will be able to redirect printers from their local machine to the remote server. Possible values: <ul style="list-style-type: none"> ▪ 0 – Users will not be able to redirect printers. ▪ 1 – Users will be able to redirect printers. This is the default value. Note: To redirect printers, the AllowMappingLocalDrives parameter must be enabled.	✓	x
Target Settings			
Protocol	Defines the target connection protocol	✓	x
ClientApp	Defines the application to open in the target machine\connection	✓	x
ClientDispatcher	Defines the internal client that will open the target connection.	✓	x
ClientInvokeType	The type of the connection client that will be used to connect to the device. Valid types are Internal (clients developed by CyberArk) and command line.	✓	x

Parameter	Description	Override at platform level	Override at account level
Client Specific	Defines a dynamic list of parameters for a specific client.		
Port	The port used to connect to the remote device. The default port for Windows transparent connections is 3389 .	✓	✓
AuthenticationLevel	The authentication level that will be used for this connection. Optional values are: <ul style="list-style-type: none"> ▪ 0 – The PSM server is not required to authenticate the target machine before connecting to it. ▪ 1 – The PSM server will authenticate the target machine before connecting to it. ▪ 2 – The PSM server will authenticate the target machine before connecting to it. If the authentication fails, the user will be able to cancel the connection or to initiate a connection without authentication. 	✓	✓
StartProgram	The full path of the program that will be started when the PSM-RDP connection is initiated.	✓	✓
WorkDir	The full path of the working directory for the program specified in the StartProgram parameter. If this property is not specified, the default working directory will be used. The default working directory is C:\Users\<current user> .	✓	✓
TerminateOnWinAuditInitFailure	Whether or not the PSM RDP session will stop when the Windows Events Audit or Universal keystrokes audit cannot be initialized. Acceptable values: Yes/No Default value: No	✓	✗

Parameter	Description	Override at platform level	Override at account level
TerminateOnWinAuditTimeout	Whether or not the PSM RDP session will stop when the Windows Events Audit or Universal keystrokes audit is not working. Acceptable values: Yes/No. Default value: Yes.	✓	✗
WindowsEventsSampleRate	How often the PSM will check for new windows that were accessed on the target machine. Default value: 0.05 seconds.	✓	✗
WindowsEventsKeepAlive	How long a session will be kept alive when the Windows Events Audit or Universal keystrokes audit is not active. Default value: 1 minute. When the specified amount of time has passed, the PSM will decide whether or not to terminate the session according to the value specified in the TerminateOnWinAuditTimeout parameter.	✓	✗
EnableTargetLogging	Whether or not trace logging to the Event Viewer on the target machine is enabled. Acceptable values: Yes/No Default value: No	✓	✗
WindowsKeystrokesSingleLanguage	Whether or not universal keystrokes recording for Windows connections will be supported for a single or additional languages during privileged sessions. Acceptable values: Yes/No Default value: Yes	✓	✗
RedirectDrivesRetries	The number of times that the PSM will try to map local drives on the client computer to the remote machine. The default value is 6 .		
RedirectDrivesRetryInterval	The number of milliseconds between PSM efforts to map local drives on the client computer to the remote machine, as defined in RedirectDrivesRetries. The default value is 5000 milliseconds.		

Parameter	Description	Override at platform level	Override at account level
User Parameters			
AllowMappingLocalDrives	Whether or not users will be allowed to redirect their local hard drives to the remote server. Note: This is not supported for remote devices that run on Windows 2000.	✓	✗
AllowConnectToConsole	Whether or not users will be allowed to connect through the PVWA to the administrative console of the remote machine.	✓	✗
RedirectSmartCards	Whether or not users will be allowed to redirect their Smart Card so that the certificate stored on the end user's card can be accessed on the target. To enable this feature, the Smart Card driver must be installed on the PSM machine. In load-balanced implementations, the driver must be installed on all load balanced PSMs.	✓	✗

- **Unix sessions (PSM-SSH)** – The following parameters are specific to Unix/SSH connection components. These are in addition to the general parameters that are common to all connection components:

Parameter	Description	Override at platform level	Override at account level
Component Parameters			
<ul style="list-style-type: none"> ▪ For RDP File connections – redirectclipboard:i ▪ For ActiveX connections – AdvancedSettings.RedirectClipboard 	Whether or not users will be able to redirect the clipboard from their local machine to the remote server. Possible values: <ul style="list-style-type: none"> ▪ 0 – Users will not be able to redirect the clipboard. ▪ 1 – Users will be able to redirect the clipboard. This is the default value. 	✓	✗

Parameter	Description	Overri de at platfor m level	Overri de at accou nt level
<ul style="list-style-type: none"> ▪ For RDP File connections – redirectprinters:i ▪ For ActiveX connections – AdvancedSettings.RedirectPrinters 	<p>Whether or not users will be able to redirect printers from their local machine to the remote server.</p> <p>Possible values:</p> <ul style="list-style-type: none"> ▪ 0 – Users will not be able to redirect printers. ▪ 1 – Users will be able to redirect printers. This is the default value. <p>Note: To redirect printers, the AllowMappingLocalDrives parameter must be enabled.</p>	✓	✗
Target Settings			
Client Specific	Defines a dynamic list of parameters for a specific client.		
Port	The port used to connect to the remote device. The default port for SSH connections is 22 .	✓	✓
AutoLogon SequenceWith LogonAccount	<p>A multiline sequence that defines an automatic sign-on process which uses a logon account to log onto a remote machine and then another account to elevate the user so that it can run sessions. The sequence uses regular expression prompts and responses with dynamic values based on the relevant account that can include one or more dynamic references. The PSM reads these references in the following order: account properties, user parameters, then client specific parameters.</p> <p>For more information, refer to <i>Using Logon Accounts for PSM-SSH and PSM-Telnet Connection Components</i>, page 647.</p>	✓	-
SendRateValue	A send rate value in milliseconds that overrides the default send rate delay value, which determines the speed at which the client will send the login sequence keystrokes.	✓	✓
PromptTimeout	A timeout value in milliseconds that overrides the default prompt timeout value, which determines how long the client will wait for the next prompt to be received before displaying an error message and closing the session.	✓	✓

- **Telnet sessions (PSM-Telnet)** – The following parameters are specific to Telnet connection components. These are in addition to the general parameters that are common to all connection components:

Parameter	Description	Override at platform level	Override at account level
Component Parameters			
<ul style="list-style-type: none"> ▪ For RDP File connections – redirectclipboard:i ▪ For ActiveX connections – AdvancedSettings.RedirectClipboard 	<p>Whether or not users will be able to redirect the clipboard from their local machine to the remote server.</p> <p>Possible values:</p> <ul style="list-style-type: none"> ▪ 0 – Users will not be able to redirect the clipboard. ▪ 1 – Users will be able to redirect the clipboard. This is the default value. 	✓	✗
<ul style="list-style-type: none"> ▪ For RDP File connections – redirectprinters:i ▪ For ActiveX connections – AdvancedSettings.RedirectPrinters 	<p>Whether or not users will be able to redirect printers from their local machine to the remote server.</p> <p>Possible values:</p> <ul style="list-style-type: none"> ▪ 0 – Users will not be able to redirect printers. ▪ 1 – Users will be able to redirect printers. This is the default value. <p>Note: To redirect printers, the AllowMappingLocalDrives parameter must be enabled.</p>	✓	✗
Target Settings			
Client Specific	Defines a dynamic list of parameters for a specific client.		
ClientProtocol	The protocol used to create the connection to the remote device. The default protocol is Telnet.	-	✓
AutoLogon Sequence	A multiline sequence that defines the automatic sign-on process using regular expression prompts and responses with placeholders for dynamic values that can include one or more dynamic references. The PSM reads these references in the following order: account properties, user parameters, then client specific parameters. For more information, refer to <i>Configuring PSM-Telnet Connection Components</i> , page 645.	✓	-

Parameter	Description	Override at platform level	Override at account level
AutoLogonSequenceWithLogonAccount	A multiline sequence that defines an automatic sign-on process which uses a logon account to log onto a remote machine and then another account to elevate the user so that it can run sessions. The sequence uses regular expression prompts and responses with dynamic values based on the relevant accounts that can include one or more dynamic references. The PSM reads these references in the following order: account properties, user parameters, then client specific parameters. For more information, refer to <i>Using Logon Accounts for PSM-SSH and PSM-Telnet Connection Components</i> , page 647.	✓	-
SendRateValue	A sent rate value in milliseconds that overrides the default send rate delay value, which determines the speed at which the client will send the login sequence keystrokes.	✓	✓
PromptTimeout	A timeout value in milliseconds that overrides the default prompt timeout value, which determines how long the client will wait for the next prompt to be received before displaying an error message and closing the session.	✓	✓

Configuring PSM-Telnet Connection Components

The **AutoLogonSequence** parameter defines a multiline sequence that is used by PSM during the automatic sign-on process for a Telnet connection to a remote device. It contains regular expression prompts and responses with dynamic values.

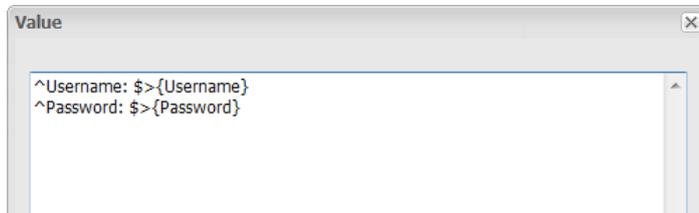
You can customize this parameter according to the logon process for each connection.

To Specify a Login Sequence for PSM-Telnet Connection Components

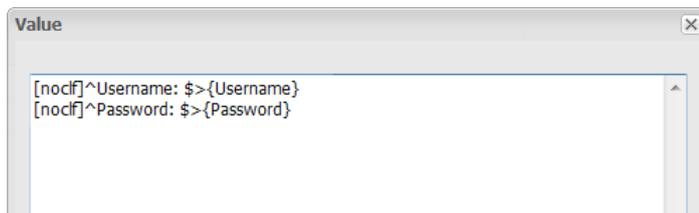
1. In the System Configuration page, click **Options**; the Web Access Options are displayed.
2. Click **Connection Components**; a list of all the configured connection components is displayed.
3. Right-click **PSM-Telnet-Sample** then, from the pop-up menu, select **Copy**.

4. Right-click **Connection Components** then, from the pop-up menu, select **Paste**; a new connection component is added to the bottom of the existing list.
5. Rename the new connection component.
 - Select the new connection component, then in the Properties list change the Id of the new connection component to **PSM-Telnet** or any other relevant name.
6. Expand **Target Settings**, and then expand **Client Specific**; a list of Client Specific parameters appears.
7. Select **AutoLogonSequence**.
8. In the Properties list, click the value of the **Value** property; the Value edit box appears.
9. Specify the logon process, as shown in the following examples.

The following example shows a simple logon process that includes a username and password then logs the user on.



To prevent the client from adding a CRLF character (new line) to the end of the response, specify (nocrlf) at the beginning of the prompt, as shown in the following example:



In each line, the text to the left of the '>' (parenthesis) represents the regular expression for the prompt on the remote machine. The text to the right of the '>' (parenthesis) represents the PSM response, including a dynamic reference to an account property.

This response can include one or more dynamic references. The PSM reads these references in the following order: account properties, user parameters, then client specific parameters.

To specify '>' as a character in the prompt, use the character code **\x3e**.

10. Click **OK**.
11. Click **Apply** to apply the new Connection Component configurations,
 - or,
 - Click **OK** to save the new Connection Component configurations and return to the System Configuration page.

Using Logon Accounts for PSM-SSH and PSM-Telnet Connection Components

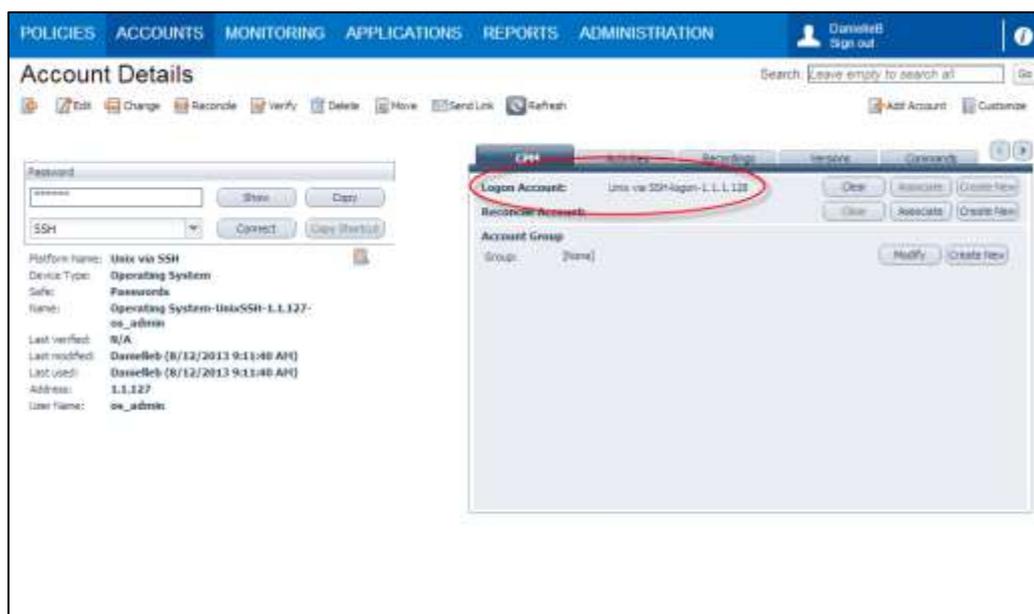
A logon account can be used to initiate sessions to machines that do not permit direct logon. This account can be used to log onto the remote machine and then elevate itself to the role of privileged user using credentials that are stored in the Vault.

The PSM uses a multiline sequence during the automatic sign-on process which contains regular expression prompts and responses with dynamic values that define the logon process and subsequent activities. As different types of machines require different logon prompts, you can override this sequence at platform level or create new connection components which you can customize.

Note: For SSH connections, the logon account can use either password or SSH key authentication. If the logon account uses SSH key authentication, the associated privileged account must use password authentication.

The following example shows the process that takes place using a logon account.

Step 1: Link a logon account to the account that cannot be used for direct logon, but will be used to run sessions on the remote machine. The following screen shows the Account Details page of the root account that will be used to run sessions on a remote machine. In this scenario, this account cannot be used to log onto the remote machine, so the **UNIX via SSH-logon-1.1.1.128** logon account has been associated with the account.



Step 2: The PSM connects to the remote machine automatically using the associated logon account and elevates the user to a privileged user. After the user clicks **Connect**, a session is opened in the remote machine and the logon account is used to log on. In this example, after successfully logging on, the current user issues the **su** command and elevates itself to the **root** user using the credentials in the main account managed in the Vault.

```

root@RHEL5Up6-64bit:~
Using username "logon".
Last login: Thu Jun 25 02:17:55 2015 from 10.10.103.14
[logon@RHEL5Up6-64bit ~]$ exec su - root
Password:
[root@RHEL5Up6-64bit ~]#

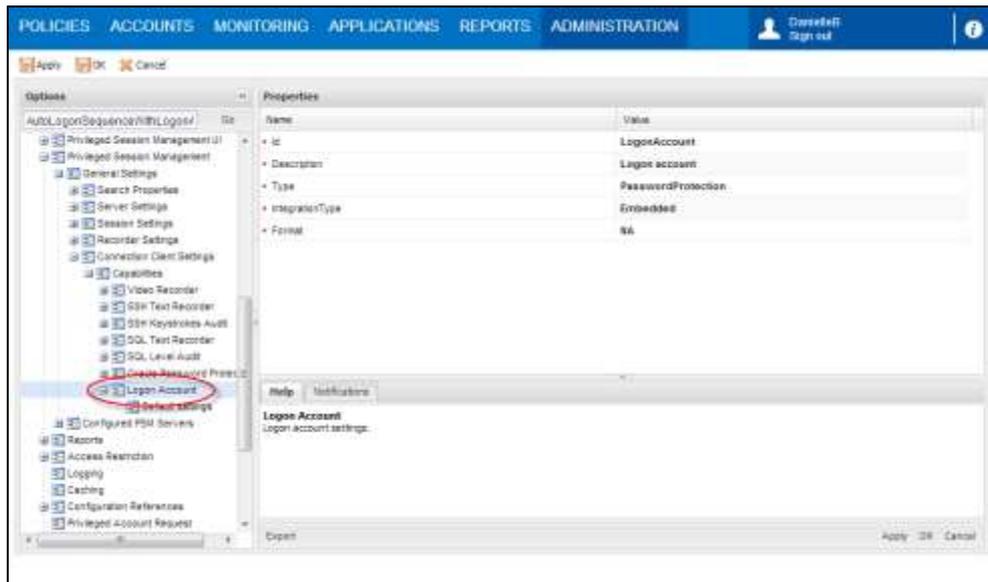
```

To Define an Automatic Logon Sequence with a Logon Account

1. Before associating a logon account, make sure that the Connection Client capabilities are configured for a logon account:
 - Note:** The logon account capability is added automatically by the PSM installation. If your first PSM installation is PSM v7.0, enable the logon account capability manually as described below.
 - i. In the System Configuration page, click **Options** to display the Web Access Options parameters.
 - ii. Select and expand **Privileged Session Management**, then expand **General Settings**.
 - iii. In **Connection Client Settings**, expand **Capabilities**.
 - iv. Right-click Capabilities, then from the pop-up menu, select **Add Logon Account**; a new Logon Account parameter is created.
 - v. In the Logon Account properties, make sure that the following property values are specified:

Property	Specifies
Id	LogonAccount
Description	LogonAccount
Type	PasswordProtection
IntegrationType	Embedded
Format	NA

These values are shown in the following window:



2. In the Account Details page of the account that will be used to run sessions on a remote machine, associate the account that will be used to log onto the remote machine.

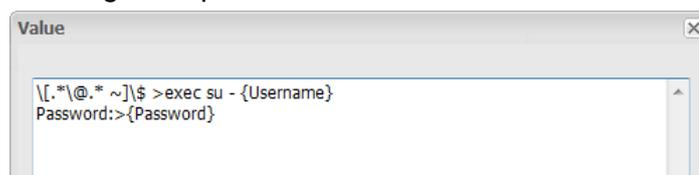
For more information about adding a linked account to new and existing accounts, refer to *Linked Accounts*, page 220.

3. Specify the automatic logon sequence with the logon account:
 - i. In the System Configuration page, display the **Web Access Options**.
 - ii. Expand **Connection Components**, then expand the connection component to configure.
 - iii. Expand **Target Settings** and then expand **Client Specific**; a list of Client Specific parameters appears.
 - iv. Select **AutoLogonSequenceWithLogonAccount**, then in the Properties list, click the value of the **Value** property; the Value edit box appears.
 - v. Specify the prompts and responses to include in the automatic logon process, using regular expressions and dynamic account properties to mimic the **exact** sequence that will be run on the remote machine.

As prompts differ according to machine, it is important to make sure that you write the prompt exactly as the machine requires.

- For PSM-SSH connections:

Specify the command that will elevate the logon user to the user who will run sessions on the remote machine. Use regular expression prompts and responses with dynamic values, as shown in the following example:

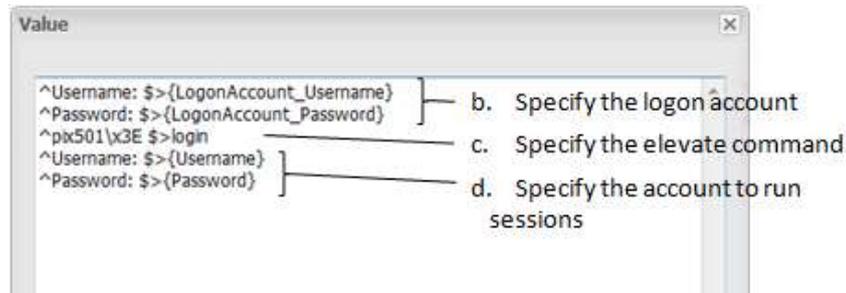


In each line, the text to the left of the '>' (parenthesis) represents the regular expression for the prompt on the remote machine. The text to the right of the '>' (parenthesis) represents the PSM response, including a dynamic reference to an account property.

This response can include one or more dynamic references. The PSM reads these references in the following order: account properties, user parameters, then client specific parameters.

To specify '>' as a character in the prompt, use the character code **\x3e**.

- For PSM-Telnet connections:
 - a. Create a new PSM-Telnet connection component, based on PSM-Telnet-Sample, as described in *Configuring PSM-Telnet Connection Components*, page 645.
 - b. Specify the logon command that enable the logon account to log onto the remote machine.
 - c. Specify the command that will elevate the logon user to the user who will run sessions on the remote machine.
 - d. Specify the username and password of the user who will run sessions on the remote machine.



In each line, the text to the left of the '>' (parenthesis) represents the regular expression for the prompt on the remote machine. The text to the right of the '>' (parenthesis) represents the PSM response, including a dynamic reference to an account property.

This response can include one or more dynamic references. The PSM reads these references in the following order: account properties, user parameters, then client specific parameters.

To specify '>' as a character in the prompt, use the character code **\x3e**.

- vi. Click **OK**; the logon sequence is displayed in the Value property as one line.
- vii. Click **Apply** to apply the new Connection Component configurations, or, Click **OK** to save the new Connection Component configurations and return to the System Configuration page.

Troubleshooting

The client 'skips' characters while imitating the login sequence

1. In the relevant connection component, add the **SendRateValue** parameter in the Client Specific target settings.
2. Set the parameter value to higher than 100 milliseconds.
3. Click **Apply** to apply the new Connection Component configurations, or, Click **OK** to save the new Connection Component configurations and return to the System Configuration page.

The following message appears: PSMSH059E Failed to execute login sequence: Incorrect sequence defined in configuration, or network timeout occurred

1. Make sure that the value of the **AutoLogonSequence** or the **AutoLogonSequenceWithLogonAccount** parameter is configured correctly.
 2. Compare the specified login sequence with the login sequence from the text recording file after a session fails.
 - i. From the Client Specific target settings, remove the **AutoLogonSequence** or the **AutoLogonSequenceWithLogonAccount** parameter.
 - ii. Run the logon sequence again to make the client text record the session from the beginning.
 - iii. After the session fails, copy the prompts for the login sequence from the text recording file.
 - iv. In the Client Specific target settings, add the **AutoLogonSequence** or the **AutoLogonSequenceWithLogonAccount** parameter again.
 3. If the specified login sequence is identical to the recorded text and the error message is still displayed, set the value of the **PromptTimeout** parameter to a much higher value. For example, **10000**.
- **WinSCP sessions (PSM-WinSCP)** – The following parameters are specific to WinSCP connection components. These are in addition to the general parameters that are common to all connection components:

Parameter	Description	Override at platform level	Override at account level
-----------	-------------	----------------------------	---------------------------

Component Parameters

<ul style="list-style-type: none"> ▪ For RDP File connections – redirectclipboard:i ▪ For ActiveX connections – AdvancedSettings.RedirectClipboard 	Whether or not users will be able to redirect the clipboard from their local machine to the remote server. Possible values: <ul style="list-style-type: none"> ▪ 0 – Users will not be able to redirect the clipboard. ▪ 1 – Users will be able to redirect the clipboard. This is the default value. 	✓	✗
<ul style="list-style-type: none"> ▪ For RDP File connections – redirectprinters:i ▪ For ActiveX connections – AdvancedSettings.RedirectPrinters 	Whether or not users will be able to redirect printers from their local machine to the remote server. Possible values: <ul style="list-style-type: none"> ▪ 0 – Users will not be able to redirect printers. ▪ 1 – Users will be able to redirect printers. This is the default value. Note: To redirect printers, the AllowMappingLocalDrives parameter must be enabled.	✓	✗

Parameter	Description	Override at platform level	Override at account level
-----------	-------------	----------------------------	---------------------------

Target Settings

Client Specific	Defines a dynamic list of parameters that are required to log onto a specific client.		
DispatcherParameters	<p>The parameter that defines the target server and the connection. This parameter uses the following syntax:</p> <pre>{Address} {Username} {Password} [{PSMClientApp}] [{Port}] [{FileTransferProtocol}] [{WindowTimeout}] [{RestrictiveMode}] [{AcceptHostKeyInCache}]</pre> <p>These parameters must be specified in the above order and on a different line.</p> <p>This syntax is explained below:</p> <ul style="list-style-type: none"> ▪ Address – Hostname/IP of the target server. ▪ Username – Username of the target account. ▪ Password – Password of the target account. ▪ WinSCP Executable Path – Location of the WinSCP exe file. If this is not specified, the default path is used – C:\Program Files (x86)\CyberArk\PSM\Components\WinSCP.exe. ▪ Port – Port used to connect to the remote device. If this is not specified, the default port is used – 22. ▪ FileProtocol – The protocol used to transfer files. Optional values are SCP and SFTP. If this is not specified, the default value is used – SFTP. ▪ WindowTimeout – Number of seconds to wait for each window. If this is not specified, the default value is used – 30 seconds. 	✓	-

Parameter	Description	Override at platform level	Override at account level
	<ul style="list-style-type: none"> ▪ RestrictiveMode – Whether or not to kill the process if an unexpected window appears during initialization and login. Specify Yes to end the process automatically or No to allow the user to handle the unexpected windows within the timeout limits. If this is not specified, the default value is used – No. ▪ AcceptAddingKeyToCache – Whether or not to dismiss the host key warning by adding the host key into the machine cache. Specify Yes to add the host key to the machine cache automatically or No to force the user to add the host key manually. If this is not specified, the default value is used – No. <p>Note: Do not specify a new line after the final parameter.</p>		
RedirectDrivesRetries	The number of times that the PSM will try to map local drives on the client computer to the remote machine. The default value is 3 .		
RedirectDrivesRetryInterval	The number of milliseconds between PSM efforts to map local drives on the client computer to the remote machine, as defined in RedirectDrivesRetries. The default value is 5000 milliseconds.		
User Parameters			
AllowMappingLocalDrives	Whether or not users will be allowed to redirect their local hard drives to the remote server.	✓	✗

Using WinSCP through a CLI

To use WinSCP through a CLI, create a new connection component as described below:

1. In the System Configuration page, click **Options**; the Web Access Options are displayed.
2. Expand **Connection Components**; the list of configured connection components is displayed.
3. Copy the original PSM-WinSCP component:
 - i. Right-click **PSM-WinSCP** then, from the pop-up menu, select **Copy**.
 - ii. Right-click **Connection Components** then, from the pop-up menu, select **Paste**; a copy of the connection component is added to the bottom of the existing list.
4. Rename the new connection component.
 - Select the new connection component, then in the Properties list change the Id of the new connection component to **WinSCP-CommandLine**.
5. Expand the new connection component and select Target Settings; the general target setting properties are displayed.
6. Change the values of the following properties:

Property	New value
ClientApp	C:\Program Files (x86)\CyberArk\PSM\Components\WinSCP.exe /console scp://{username}:{password}@{address}
ClientDispatcher	NA
ClientInvokeType	CommandLine

7. Right-click **Lock Application Window**, then from the pop-up menu, select **Detete**; the Lock Application Window parameter is removed from the target settings parameters.
8. Add a new Client Specific parameter:
 - i. Right-click **Client Specific**, then select **Add Parameter**; a new parameter is added.
 - ii. In the properties list, specify the following values:
 - **Name** – The name of the new parameter. Specify **CmdLineParmHideTimeout**.
 - **Value** – The time, in milliseconds, that the PSM waits for the command line parameters hiding process to finish its operation. Specify **50000**.
9. Click **OK** to save the new Connection Component configurations and return to the System Configuration page.
10. Click **ADMINISTRATION** to display the **System Configuration** page, then click Platform Management to display a list of supported target account platforms.
11. In the platform that will support WinSCP connections through a command line Create a platform, add the new connection component.
12. Click **Apply** to apply the new Connection Component configurations, or, Click **OK** to save the new Connection Component configurations and return to the System Configuration page.

- **OS/390 (Z/OS) sessions** – The following parameters are specific to OS/390 (Z/OS) connection components. These are in addition to the general parameters that are common to all connection components:

Parameter	Description	Override at platform level	Override at account level
Target Settings			
Client Specific	Defines a dynamic list of parameters for a specific client.		
SourceFileTemplate	A macro file that contains a list of commands to the client. These commands can be specified with placeholders (in parentheses {}), so that users can specify custom metadata. For a complete list of commands, refer to <i>Customizing AS400 (iSeries) and OS/390 (Z/OS) Emulation Parameters</i> , page 635. Note: The default source file template is a sample. Change this to specify the source file in your environment.	✓	✓
CommandLine Arguments	The wc3270 option that can be run during the PSM-OS390 connection session. For a complete list of commands, refer to <i>Customizing AS400 (iSeries) and OS/390 (Z/OS) Emulation Parameters</i> , page 635.	✓	✓
Lock Application Window	Defines the behavior of the Lock Application Window process.		
Enable	Whether or not the application window will be locked on the screen. Default value: No. Note: This parameter is ignored when the RemoteApp user experience is enabled. For more information, refer to the UseRemoteApp parameter in <i>Configuring the PSM Session User Experience</i> , page 610.	✓	✗
MainWindowTitle	Used to identify the main window.	✓	✗
MainWindowClass	Used to identify the main window.	✓	✗
Timeout	The time, in milliseconds, to wait for the application window to be displayed. Default value: 8000 milliseconds.	✓	✗
SearchWindow WaitTimeout	The time, in milliseconds, to wait between each iteration when searching for the application window. Default value: 30 milliseconds.	✓	✗

- **AS400 (iSeries) sessions** – The following parameters are specific to AS400 (iSeries) connection components. These are in addition to the general parameters that are common to all connection components:

Parameter	Description	Override at platform level	Override at account level
Component Parameters			
<ul style="list-style-type: none"> ▪ For RDP File connections – redirectclipboard:i ▪ For ActiveX connections – AdvancedSettings.RedirectClipboard 	<p>Whether or not users will be able to redirect the clipboard from their local machine to the remote server.</p> <p>Possible values:</p> <ul style="list-style-type: none"> ▪ 0 – Users will not be able to redirect the clipboard. ▪ 1 – Users will be able to redirect the clipboard. This is the default value. 	✓	✗
<ul style="list-style-type: none"> ▪ For RDP File connections – redirectprinters:i ▪ For ActiveX connections – AdvancedSettings.RedirectPrinters 	<p>Whether or not users will be able to redirect printers from their local machine to the remote server.</p> <p>Possible values:</p> <ul style="list-style-type: none"> ▪ 0 – Users will not be able to redirect printers. ▪ 1 – Users will be able to redirect printers. This is the default value. <p>Note: To redirect printers, the AllowMappingLocalDrives parameter must be enabled.</p>	✓	✗
Target Settings			
Client Specific	Defines a dynamic list of parameters for a specific client.		
SourceFileTemplate	<p>A macro file that contains a list of commands to the client. These commands can be specified with placeholders (in parentheses {}), so that users can specify custom metadata. For a complete list of commands, refer to <i>Customizing AS400 (iSeries) and OS/390 (Z/OS) Emulation Parameters</i>, page 635.</p> <p>Note: The default source file template is a sample. Change this to specify the source file in your environment.</p>	✓	✓

Parameter	Description	Override at platform level	Override at account level
CommandLine Arguments	The list of WC3270 options that can be run during the PSM-AS400 connection session. You can specify multiple options, separated by a comma. For a complete list of commands, refer to <i>Customizing AS400 (iSeries) and OS/390 (Z/OS) Emulation Parameters</i> , page 635.	✓	✓
Lock Application Window	Defines the behavior of the Lock Application Window process.		
Enable	Whether or not the application window will be locked on the screen. Default value: No. Note: This parameter is ignored when the RemoteApp user experience is enabled. For more information, refer to the UseRemoteApp parameter in <i>Configuring the PSM Session User Experience</i> , page 610.	✓	✗
MainWindowTitle	Used to identify the main window.	✓	✗
MainWindowClass	Used to identify the main window.	✓	✗
Timeout	The time, in milliseconds, to wait for the application window to be displayed. Default value: 8000 milliseconds.	✓	✗
SearchWindow WaitTimeout	The time, in milliseconds, to wait between each iteration when searching for the application window. Default value: 30 milliseconds.	✓	✗

Configuring Platform Level Parameters

- **Windows sessions (PSM-RDP)** – The following parameters are specific to the PSM-RDP connection components in Windows platforms.

Parameter	Description	Override at account level
User Parameters		
AllowMappingLocalDrives	Whether or not users will be allowed to redirect the local hard drives to the remote server. Note: This is not supported for remote devices that run on Windows 2000.	✗
AllowConnectToConsole	Whether or not users will be allowed to connect through the PVWA to the administrative console of the remote machine.	✗
LogonDomain	The Windows account property whose value will be used to resolve the logon domain. Note: If this parameter is specified in the account, it will not be displayed in the system tab.	✓
PSMRemoteMachine	The name of the remote device. This can be configured for Windows Domain accounts and UNIX domain/NIS.	✗
RedirectSmartCards	Whether or not Smart Card redirection is enabled, so that the certificate stored on the end user's card can be accessed on the target. To enable this feature, the Smart Card driver must be installed on the PSM machine. In load-balanced implementations, the driver must be installed on all load balanced PSMs.	✗
Each parameter has the following settings:		
Name	The name of the parameter.	
DisplayName	The exact way that the parameter name will be displayed in the connection window.	
Visible	Whether or not the user will be prompted for this parameter before the connection is established.	
Type	The type that will be used to modify the appearance or behavior of a parameter UI field. Note: This is an internal parameter and must not be changed without consulting your CyberArk support contact.	
Required	Whether or not the user is required to provide this information for the remote connection to be activated.	
Value	The default value of this parameter.	
EnforceInDualControlRequest	Whether or not the user will be required to provide this information in order to create a dual control request.	

Parameter	Description	Override at account level
Target Settings		
Port	The port used to connect to the remote device. The default port for Windows transparent connections is 3389 .	✓

- **SSH platforms** –The following parameters are specific to the PSM-SSH connection components in SSH platforms.

Parameter	Description	Override at account level
Target Settings		
Port	The port used to connect to the remote device. The default port for SSH connections is 22 .	✓

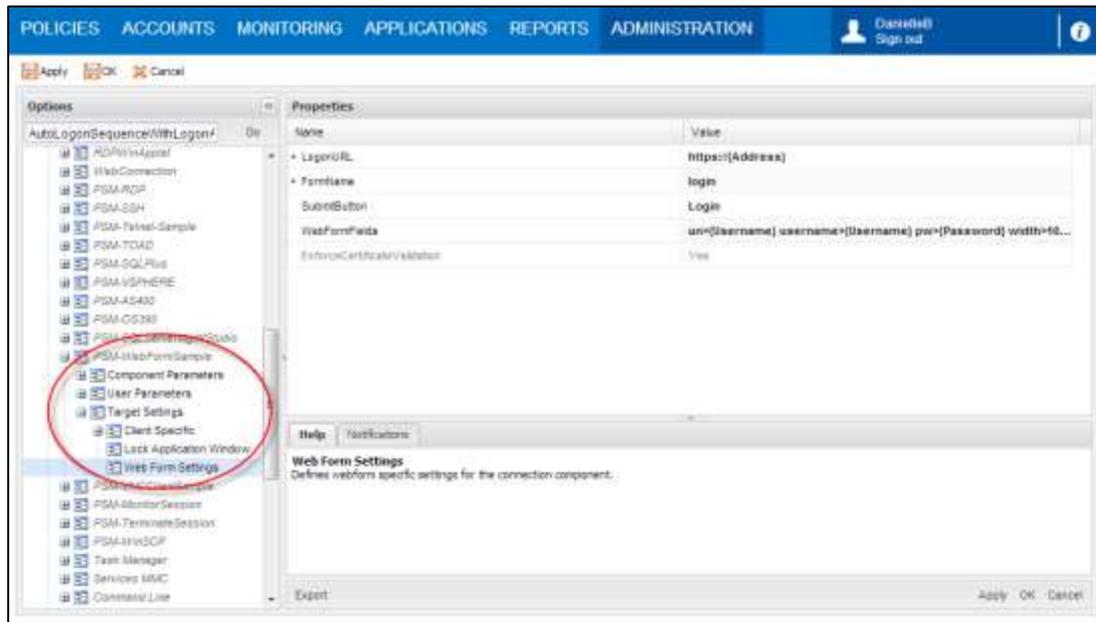
Configuring PSM Connection Components for Web Applications

You can configure connection components for web applications based on a default generic connection component that is created in the PVWA automatically during installation.

Note: Web applications with login forms in Java or Flash are not supported with this default connection component, although you can configure a PSM Universal Connector to support them. For more information, refer to *Configuring a PSM Universal Connector Connection Component*, page 668.

1. In the System Configuration page, in the **Component Settings**, click **Options**, then expand **Connection Components**; the defined connection components are displayed.
2. Right-click **PSM-WebFormSample**, then from the pop-up menu select **Copy**.
3. Right-click **Connection Components**, then from the pop-up menu select **Paste Connection Component**; a new connection component is added to the bottom of the list. This connection component is also called PSM-WebFormSample.
4. Change the **Id** of the new connection component and set any other parameters that will define the connection.
5. In **Target Settings**, select **Web Form Settings**; a list of web form properties is displayed in the Properties list.

The following examples show how to specify the Webform settings for a new connection component to Salesforce.



6. Specify the following properties:

- **LogonURL** – The address of the web application. Specify the address in one of the following ways:

- **Dynamic URL** – Specify the URL with a placeholder for the actual address, like `http://{Address}` or `http://{Address}/Welcome/Login`. When the connection component is used to connect to a web application, the address will be replaced by the value of the address property of the account being used to logon. Any relevant account property can be specified in the parentheses.

For IPv6 WebForms accounts:

- If the address property includes only the IP address of the website, specify the address in IPv6 format as shown in the following example:
`1000:1000:1000:1000:1000:1000:1000:51`
- If the address property includes the IP address of the website and the path of the login page, specify the address in IPv6 format surrounded by brackets, as shown in the following example:
`[1000:1000:1000:1000:1000:1000:1000:51]/homepage`

or,

- **URL** – Specify the actual URL of the login page. For example, to specify the login page of `Salesforce.com`, specify `https://login.salesforce.com/`. Make sure that if the main page of the website automatically transfers to another page for login, specify the address of the page where the login form resides.

- **SubmitButton** – The name of the button on the web page that submits the logon information. Using the Salesforce example, this is the name of the **Login** button. In the following example of the salesforce login page, the button name is circled:

```
<div class="inputs"><div class="inputbox"><p><label for="username">User Name</label></p><span><input class="inputbox glow" type="text" id="username" name="username" value="pm@cyber-ark.com" /></span></div><div class="inputbox"><p><label for="password">Password</label></p><span><input class="inputbox glow" type="password" id="password" name="pw" size="18" autocomplete="off" onkeypress="checkCaps(event)"/></span></div><div class="pwcaps" style="display:none"></div><div class="inputbox"><input class="checkbox" type="checkbox" id="rememberUn" name="rememberUn" checked="checked"/><label for="rememberUn">Remember User Name</label></div><div class="inputbox" class="loginButton" type="submit" id="Login" name="Login" value="Login" /><a class="forgotpass" href="/secur/forgotpassword.jsp?locale=uk">Forgot your password?</a></div></div><div class="hide
```

Note: Currently, the PSM does not support submit buttons that contain Java Script.

- **WebFormFields** – Pairs of form fields and account properties that define the fields on the login page for which credentials will be sent to the web application. Each pair includes the name of the textbox in the login page and the account property that will be sent as the value of that element.

Note: The WebFormFields must specify the form ID or name and all the other fields that are required to log onto the web application. Make sure that you specify all the relevant fields in the login form.

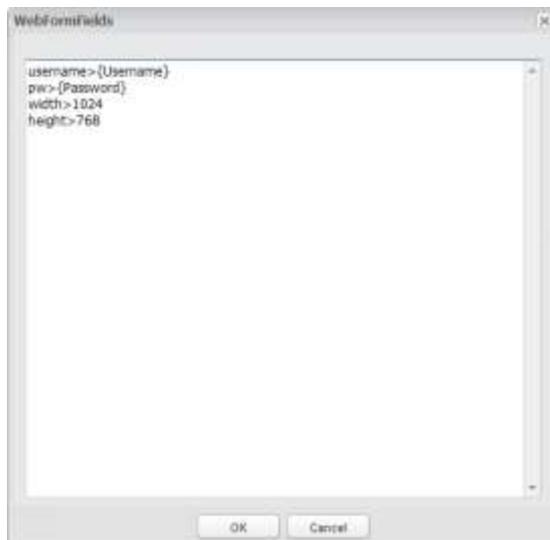
In the following example, you can see the username and password elements in the source code of the Salesforce login page.

```
<div class="inputs"><div class="inputbox"><p><label for="username">User Name</label></p><span><input class="inputbox glow" type="text" id="username" name="username" value="user@cyber-ark.com" /></span></div><div class="inputbox"><p><label for="password">Password</label></p><span><input class="inputbox glow" type="password" id="password" name="pw" size="18" autocomplete="off" onkeypress="checkCaps(event)"/></span></div><div id="pwcaps" class="pwcaps" style="display:none" src="/login/assets/warning16.png" alt="Caps Lock is ON!" /> Caps Lock is ON!</div><div class="inputbox"><input class="checkbox" type="checkbox" id="rememberUn" name="rememberUn" checked="checked"/><label for="rememberUn">Remember User Name</label></div><div class="inputbox" class="loginButton" type="submit" id="Login" value="Login" /><a class="forgotpass" href="/secur/forgotpassword.jsp?locale=uk">Forgot your password?</a></div></div><div class="hidesubmit"><input type="image" src="/login/assets/trans.gif" alt="Submit" value="Login" /></div></form><div id="signupbox">Don't have an account?<a class="signup_link_but
```

- Identify the name of the field elements in the source code of the login page.
- In the PVWA, in the WebFormFields property, change the name of the field elements in the login page and the account property that will be sent as the value of that element using the following syntax:

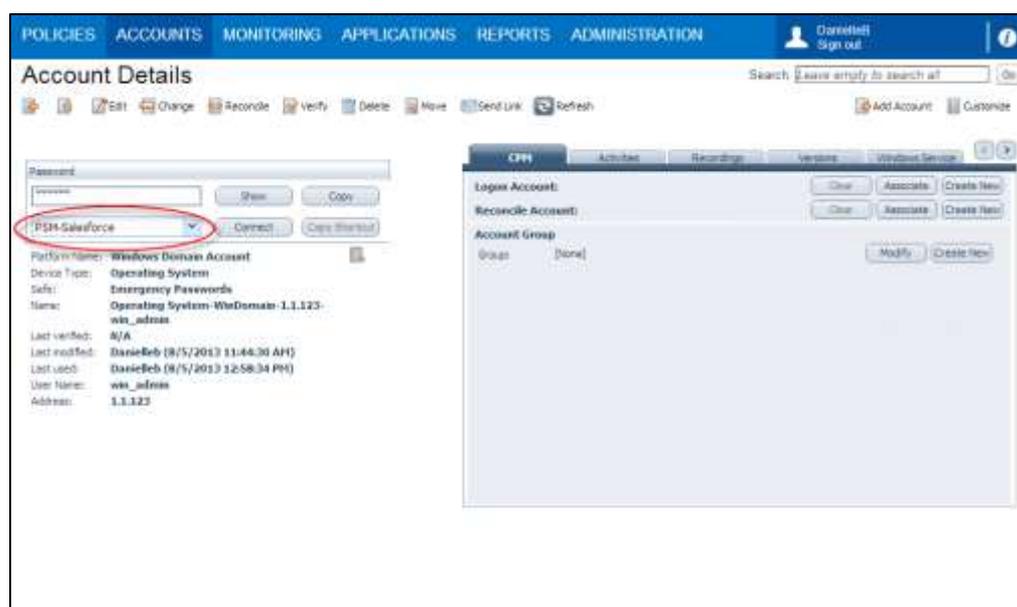
`<label name>>{account property}`

The following example shows the webform fields and account properties to specify for the Salesforce login page.



- iii. Click **OK**; the specified WebFormFields are listed in the WebFormFields property.
 - **EnforceCertificateValidation** – Whether or not PSM will validate target website certificates when initiating PSM connections. This enables PSM to connect to local websites that do not have valid certificates, such as LAN applications with self-signed certificates. To connect to local websites that use self-signed certificates, specify **No**.
 5. Click **OK** to save the new Connection Component configurations and return to the System Configuration page.
 6. Click **Platform Management** to display a list of supported target account platforms.
 7. Select the platform to configure, then click **Edit**; the settings page for the selected platform appears.
 8. Expand **UI & Workflows**, right-click **Connection Components**, then from the pop-up menu select **Add Connection Component**; a new connection component is added to the current list of connection components that are configured for this platform.
 9. In the new Connection Component, specify the following properties:
 - **Id** – The unique ID that identifies the connection component you created previously for the specific command. Using the Salesforce example, this will be PSM-Salesforce.
 - **Enable** – Whether or not this connection component will be enabled for this platform. Specify **Yes**.
 10. Click **Apply** to apply the new platform configurations,
- or,
- Click **OK** to save the new platform configurations and return to the System Configuration page.

A new connection component for web applications has now been configured and linked to the platform. The next time a user wishes to use an account to run a privileged session using the configured platform, the new connection will be displayed.



Troubleshooting

The PSM is redirected to another website and it takes a few seconds to load the page

- Add the **PageLoadTimeout** parameter to define the time that the PSM will wait for the system to load the web form.
1. In the System Configuration Options, expand the **PSM-WebFormSample** connection component.
 2. In **Target Settings**, right-click **Client Specific**, then select **Add Parameter**; a new parameter is added.
 3. In the Parameter properties list, specify the following:
 - **Name** – The name of the client specific parameter. Specify **PageLoadTimeout**.
 - **Value** – The number of milliseconds that the PSM will wait for the system to load the web form.
 4. Click **OK** to save the new Connection Component configurations.

Configuring PSM Connections for Database Connections

Configuring Connection Component Parameters

- **DBA tools sessions (PSM-TOAD, PSM-SQLPlus, PSM-SMSS)** – The following parameters are specific to Toad, SQL *Plus, SQL Server Management Studio connection components. These are in addition to the general parameters that are common to all connection components:

Parameter	Description	Override at platform level	Override at account level
ConnectionComponent (root level)			
EnableWindow Scrollbar	Whether or not scrollbars will be added to the transparent connection window. <ul style="list-style-type: none"> ▪ The default value for PSM-Toad connections is No. ▪ The default value for PSM-SQLPlus connections is Yes. 	x	x
Component Parameters			
RequireConnectAs OnAccounts	<ul style="list-style-type: none"> ▪ Whether or not additional system users can be used to connect to remote databases, and a list of users who can specify these additional users. <p>Note: This parameter is not relevant for PSM-SMSS connections.</p>	✓	x

Parameter	Description	Override at platform level	Override at account level
<ul style="list-style-type: none"> ▪ For RDP File connections – redirectclipboard:i ▪ For ActiveX connections – AdvancedSettings.RedirectClipboard 	<p>Whether or not users will be able to redirect the clipboard from their local machine to the remote server.</p> <p>Possible values:</p> <ul style="list-style-type: none"> ▪ 0 – Users will not be able to redirect the clipboard. ▪ 1 – Users will be able to redirect the clipboard. This is the default value. 	✓	✗
User Parameters			
AllowMappingLocalDrives	<p>Whether or not users will be allowed to redirect their local hard drives to the remote server.</p> <p>Note: This is not supported for remote devices that run on Windows 2000.</p>	✓	✗
ConnectAs	<p>Whether or not users will be able to specify additional system users to connect to remote databases.</p> <p>Note: This parameter is not relevant for PSM-SMSS connections.</p>	✓	✗
Target Settings			
Client Specific	Defines a dynamic list of parameters for a specific client.		
WaitBeforeCmdlineParmsHide	<p>Time, in milliseconds, to wait before hiding the parameters in the command line.</p> <p>Default value: 100 milliseconds.</p>	✓	✗
CmdLineParmsHideTimeout	<p>The time, in milliseconds, that the PSM waits for the command line parameters hiding process to finish its operation.</p> <p>Default value: 1000 milliseconds.</p>	✓	✗
Lock Application Window	Defines the behavior of the Lock Application Window process.		
Enable	<p>Whether or not the application window will be locked on the screen.</p> <p>Default value: Yes.</p> <p>Note: This parameter is ignored when the RemoteApp user experience is enabled. For more information, refer to the UseRemoteApp parameter in <i>Configuring the PSM Session User Experience</i>, page 610.</p>	✓	✗
MainWindow Title	Used to identify the main window.	✓	✗
MainWindow Class	Used to identify the main window.	✓	✗

Parameter	Description	Override at platform level	Override at account level
Timeout	The time, in milliseconds, to wait for the application window to be displayed. Default value: 8000 milliseconds.	✓	✗
SearchWindow WaitTimeout	The time, in milliseconds, to wait between each iteration when searching for the application window. Default value: 30 milliseconds.	✓	✗

Configuring PSM Connections for Virtualization Connections

Configuring Connection Component Parameters

- VMWare tools sessions (PSM-VSHERE)** – The following parameters are specific to the PSM-VSHERE connection components. These are in addition to the general parameters that are common to all connection components:

Parameter	Description	Override at platform level	Override at account level
ConnectionComponent (root level)			
EnableWindowScrollbar	Whether or not scrollbars will be added to the transparent connection window.	✗	✗
User Parameters			
AllowMapping LocalDrives	Whether or not users will be allowed to redirect their local hard drives to the remote server. Note: This is not supported for remote devices that run on Windows 2000.	✓	✗
Component Parameters			
<ul style="list-style-type: none"> For RDP File connections – redirectclipboard:i For ActiveX connections – AdvancedSettings.RedirectClipboard 	Whether or not users will be able to redirect the clipboard from their local machine to the remote server. Possible values: <ul style="list-style-type: none"> 0 – Users will not be able to redirect the clipboard. 1 – Users will be able to redirect the clipboard. This is the default value. 	✓	✗
Target Settings			
Client Specific	Defines a dynamic list of parameters for a specific client.		

Parameter	Description	Override at platform level	Override at account level
WaitBeforeCmdlineParmsHide	Time, in milliseconds, to wait before hiding the parameters in the command line. Default value: 2000 milliseconds.	✓	✗
CmdLineParmsHideTimeout	The time, in milliseconds, that the PSM waits for the command line parameters hiding process to finish its operation. Default value: 1000 milliseconds.	✓	✗
Lock Application Window	Defines the behavior of the Lock Application Window process.		
Enable	Whether or not the application window will be locked on the screen. Default value: Yes. Note: This parameter is ignored when the RemoteApp user experience is enabled. For more information, refer to the UseRemoteApp parameter in <i>Configuring the PSM Session User Experience</i> , page 610.	✓	✗
MainWindow Title	Used to identify the main window.	✓	✗
MainWindow Class	Used to identify the main window.	✓	✗
Timeout	The time, in milliseconds, to wait for the application window to be displayed. Default value: 8000 milliseconds.	✓	✗
SearchWindow WaitTimeout	The time, in milliseconds, to wait between each iteration when searching for the application window. Default value: 30 milliseconds.	✓	✗

Configuring a Default Connection Component

You can set a default connection component for each platform that is configured to work with a PSM connection component. This connection component is automatically displayed in the drop-down connection component list in the Account Details page.

1. Click **ADMINISTRATION** to display the **System Configuration** page, then click **Platform Management** to display a list of supported target account platforms.

2. Select the platform for which the connection component has been configured, then click **Edit**; the settings page for the selected platform appears.
3. Expand **UI & Workflows**, and then select **Connection Components**; the Connection Components parameters are displayed with their default values.
4. In the **PSMConnectionDefault** property, specify the default connection component for this platform. This connection component must be defined for this platform.
5. Click **Apply** to apply the new Connection Component configurations,
or,
Click **OK** to save the new Connection Component configurations and return to the System Configuration page.

Configuring a PSM Universal Connector Connection Component

You can create PSM Universal Connector connection components that integrate your own GUI client(s) into the PSM. Like all other PSM connection components, PSM Universal Connector connection components benefit from standard PSM functionality and security, ensuring seamless integration and smooth workflows.

The PSM Universal Connector allows you to automate the launch and authentication process of your application clients. The automation processes are written in AutoIt scripting language (<http://www.autoitscript.com>). These connection components can be either developed by the customer or as a CyberArk Professional Services engagement.

During PSM installation, the following files are copied to the Components subfolder of the PSM installation folder:

- **PSMRealVNCDispatcher.au3** – A sample AutoIt script of the PSM Universal Connector connection component for connecting to server over the VNC protocol. This sample is also created as a sample connection component in the PWWA.
- **PSMAutoItDispatcherSkeleton.au3** – The basic AutoIt script file. This file contains a skeleton script for the PSM Universal Connector connection component. You can copy this script and customize it.
- **PSMGenericClientWrapper.au3** – An AutoIt script that contains the API functions required for the PSM Universal Connector connection component. Do not modify this file.

- Files that comprise a driver that enables you to test your customized connection component:
 - PSMGenericClientDriver.dll
 - PSMGenericClientDriver.xml

Prerequisites for using PSM Universal Connector Connection Components

On the PSM machine:

- Install Autolt3, version 3.3.6.1. You can download it from <http://www.autoitscript.com/autoit3/files/archive/autoit/autoit-v3.3.6.1-setup.exe>
By default, Autolt3 will be installed in the following location on your computer:
%ProgramFiles%\Autolt3.

If you are using a separate development machine to develop your Autolt scripts, install the Autolt script on that machine as well.

To Configure a new PSM Universal Connector Connection Component

1. On a development machine, develop an Autolt script that will launch and authenticate to your application for your connection component, as follows:
 - i. Install the application to integrate on your development machine.
 - ii. In the Components subfolder of the PSM installation folder, copy **PSMAutoltDispatcherSkeleton.au3** to your development environment and rename it, but **do not** change the .au3 extension.
 - iii. Open the new .au3 file and change the code contained in it, as follows:
 - a. Replace each of the ";CHANGE_ME" comments with your own code/constants.
 - b. In the value of the \$CLIENT_EXECUTABLE constant, specify the path of the application to execute.
 - c. In the Main() function, replace the "Handle login here! comment" with the logic that handles the login using the PSM Universal Connector API. For more information about this API, see the *PSM Universal Connector API Reference*, page 673.
2. Debug and test the Autolt script using the Generic Client Driver supplied by CyberArk, as follows:
 - i. Copy the following files from the PSMComponents folder on the PSM machine to the same folder of the new au3 script on your development workstation:
 - PSMGenericClientDriver.dll
 - PSMGenericClientDriver.xml
 - ii. In the **PSMGenericClientDriver.xml** file, update the SessionProperties section and specify values that you will use to test connections using your application.

The following sample shows a typical PSMGenericClientDriver.xml file that specifies the functions to check and the session properties to return.

```
<?xml version="1.0" encoding="utf-8" ?>
<PSMGenericClientDriver>
  <Functions>
    <Function Name="SendPID" Return="Success" />
    <Function Name="MapTSDrives" Return="Success" />
    <Function Name="GetSessionPropertyBufferLength"
Return="Success" />
    <Function Name="GetSessionProperty" Return="Success" />
    <Function Name="LogWrite" Return="Success" />
  </Functions>
  <SessionProperties>
    <SessionProperty Name="SessionUUID" Value="0b4d3135-d824-
4044-8a3f-555a72b72577" />
    <SessionProperty Name="Username" Value="administrator" />
    <SessionProperty Name="Address" Value="10.10.10.10" />
    <SessionProperty Name="Password" Value="very-secured-
pass" />
  </SessionProperties>
</PSMGenericClientDriver>
```

- iii. Run the test driver, using the following command:

```
C:\<path>\autoit3.exe <your-script.au3> "c:\<Folder of the new
au3 script>\" /test
```

Note: In the above command, the second parameter specifies the path to the folder that holds the driver's dll and the new au3 script. This parameter must end with a backslash.

- iv. The file PSMGenericClientDriver.txt will be created and will include the log output of the script.
- v. To prevent hiding the main application window, thus causing the session not to be recorded, you must provide the title and class name of that window to the PSM Administrator. The Administrator will use this information when configuring the PSM to support the new PSM Universal Connector connection component.
3. Install the tested script on the PSM machine, as follows:
- i. Install the connection client you want to integrate on the PSM machine.
 - ii. Copy the .au3 file developed above to the Components subfolder of the PSM installation folder.

Note: You will specify this file when you add the connection component to the PVWA.
 - iii. If you are using applocker, uncomment the generic client support section in the PSM applocker xml file and modify the generic client sample rule to match the connection client you installed in step 3.i, as follows:
 - a. On the PSM server, configure the AppLocker to enable AutoIt and your new connection client to run. For specific instructions, refer to the Privileged Account Security Installation Guide.

- b. In the Hardening subfolder of the PSM installation folder, open the PSMConfigureAppLocker.xml configuration file and edit it as follows:
 - i. In the **AllowedApplications** section, remove the comment indication from the Generic client support section:
 - At the beginning of the Generic client support section, remove the following line:

```
<!-- If relevant, uncomment this part to allow generic clients support and add a rule for each generic connection client
```

- At the end of the Generic client support section, remove the following line:

```
End of Generic client support comment -->
```

- ii. Modify the path attribute of the GenericClient-Sample to match the path of the executable of the new connection client.
 - iii. Save the **PSMConfigureAppLocker.xml** configuration file and close it.
- c. Run PowerShell and start the applocker script using the following command:

```
CD "C:\Program Files (x86)\CyberArk\PSM\Hardening"  
%SYSTEMROOT%\WindowsPowerShell\v1.0\PowerShell.exe  
PSMConfigureAppLocker.ps1
```

4. Create a new PSM Universal Connector connection component in the PVWA, as follows:
 - i. Log onto the PVWA as an administrative user.
 - ii. In the System Configuration page, click **Options**; the Web Access Options are displayed.
 - iii. Expand **Connection Components**; the defined connection components are displayed in the properties list.
 - iv. Right-click the **PSM-VNCCClientSample** connection component, then from the pop-up menu select **Copy**.
 - v. Right-click **Connection Components**, then from the pop-up menu select **Paste Connection Component**; a copy of the PSM-VNCCClientSample connection component is added to the list of configured connection components with its original ID (PSM-VNCCClientSample).
 - vi. Configure the new connection component:
 - a. In the list of properties, specify a new ID that reflects the purpose of the connection. This ID will be displayed in the drop-down list of available connections in the Accounts Details page and must be self-explanatory.
 - b. Expand the connection component, then select **Target Settings**; the Target Settings properties are displayed.
 - c. In the **ClientDispatcher** property, replace **PSMRealVNCDDispatcher.au3** with the name of the .au3 file that you renamed and edited previously.

Note: Make sure that this file is in the Components subfolder of the PSM installation folder.

Name	Value
ClientApp	dummy.exe
ClientDispatcher	C:\Program Files (x86)\AutoK\AutoK1.exe\PSMComponentsFolder\PSMRes\WinCD\dispatcher.au3\PSMComponentsFolder\.
ClientInvokeType	Dispatcher
ConnectionComponentRefreshInterval	20000
Protocol	VNC

- d. Under **Target Settings**, select **Lock Application Window**; the Lock Application Window properties are displayed.
- e. In the **MainWindowTitle** and **MainWindowClass** properties, set the title and class name provided by the developer in step 2.v above.

Note: These properties can remain empty if they are not applicable. In this case, the PSM will lock the first window created by the application.
- vii. Click **OK** to save the new Connection Component configurations and return to the System Configuration page.
- viii. Click **Platform Management** to display a list of supported target account platforms.
5. Select the platform to configure for the connection component, then click **Edit**; the settings page for the selected platform appears.
6. Expand **UI & Workflows**, right-click **Connection Components**, then from the pop-up menu select **Add Connection Component**; a new connection component is added to the current list of connection components that are configured for this platform.
 - ix. In the new Connection Component, specify the following properties:
 - **Id** – The unique ID that identifies the connection component you created previously for the specific command.
 - **Enable** – Whether or not this connection component will be enabled for this platform. Specify **Yes**.
 - x. Click **Apply** to save the new values and stay in the platform settings page, or, Click **OK** to save the new values and return to the System Configuration page.

The changes will be applied after the period of time specified in the **ConfigurationRefresh Interval** parameter.

Security Recommendations and Best Practices

- Make sure that the password used by the PSM Universal Connector connection client cannot be exposed to the user interface.
- Make sure that sensitive data will not be saved to the disk during the session; in case some data is saved. Consider cleaning this data as part of the AutoIt script.

PSM Universal Connector API Reference

The PSM Universal Connector API is used from within the Autolt script to integrate with the PSM environment, and obtain information such as the session properties (e.g. address/user/password) that the PSM provides to the script.

The following **PSMDispatcherUtilsWrapper** functions enable you to create the Autolt wrapper:

Init

Description: This function initializes PSM's dispatcher API library. It must be called once at the beginning of the Autolt script implementation (before any other API calls).

Parameters: This function accepts no parameters.

Return values: \$PSM_ERROR_SUCCESS – Indicates that the API library was initialized successfully.

Any other return value means that an error occurred. For more details, use `PSMDispatcherUtilsWrapper_PSMGetLastErrorString`.

Term

Description: This function terminates PSM's dispatcher API library. It must be called once after the Autolt script finishes using the PSM API functions in order to free consumed resources.

Parameters: This function accepts no parameters.

Return values: This function has no return values.

IsInitialized

Description: This function checks if the API library is currently loaded.

Parameters: This function accepts no parameters.

Return values:

- True – The library is currently loaded.
- False – The library is not currently loaded.

PSMGetLastErrorString

Description: This function returns a description of the last error that occurred in prior PSM library API calls.

Parameters: This function accepts no parameters.

Return values: A string that describes the last error that occurred.

LogWrite

- Description:** This function writes the following types of log messages in the ClientDispatcher log file.
- Error messages – These messages will always be written in the log file, regardless of the client trace levels.
 - Non-error messages – These messages will only be written in the log file when the client trace level is set to “2”.
- Parameters:** This function has two parameters:
- Message – The message to write in the log file.
 - LogLevel:
 - Error message – If the message is an error message, use the `$LOG_LEVEL_ERROR` constant.
 - Trace message – If the message is a trace message, use the `$LOG_LEVEL_TRACE` constant.
- Return values:**
- `$PSM_ERROR_SUCCESS` – Indicates that the error was written in the log file successfully.
- Any other return value means that an error occurred. For more details, use `PSMDispatcherUtilsWrapper_PSMGetLastErrorString`.

SendPID

- Description:** This function reports the PID of the connection client started by the dispatcher to the PSM server, after which the PSM starts monitoring the connection client. If the function is not called, the session will be stopped as soon as the dispatcher exits.
- Parameters:** `$PID` - The process ID to send to the PSM server
- Return values:**
- `$PSM_ERROR_SUCCESS` – Indicates that the error was written in the log file successfully.
- Any other return value means that an error occurred. For more details, use `PSMDispatcherUtilsWrapper_PSMGetLastErrorString`.

MapTSDrives

- Description:** This function is used to enable RDP local drive mapping for connection clients that do not support “Windows Shell Extensions”.
- When the system detects that a session has mapped local drives, it adds shortcuts to these drives in the computer’s Explorer window. If the connection client application does not display the Windows’ standard File Open/Save windows, these drives won’t appear anywhere. In these cases, the MapTSDrives will actually allocate drive letters for the mapped drives so that the drives will appear in any kind of File Open/Save windows.
- Parameters:** This function accepts no parameters.
- Return values:**
- \$PSM_ERROR_SUCCESS – Indicates that RDP successfully mapped local drives for connection clients.
- Any other return value means that an error occurred. For more details, use PSMDispatcherUtilsWrapper_PSMGetLastErrorString

GetSessionProperty

- Description:** This function retrieves a session property (address/user/password) from the PSM by passing the property’s name as a parameter.
- Parameters:**
- SessionPropertyName - [IN parameter] The name of the session property whose value is required.
 - SessionProperty - [OUT parameter] The value of the session property value.
- Return values:**
- \$PSM_ERROR_SUCCESS – Indicates that the value of the session property was retrieved successfully.
- Any other return value means that an error occurred. For more details, use PSMDispatcherUtilsWrapper_PSMGetLastErrorString.

Configuring PSM Connection Components for Amazon Web Services (AWS)

PSM includes an out-of-the-box AWS Console connection component that integrates with AWS Secure Token Service (STS) and allows an administrator to configure accounts with specific AWS roles and/or policies. Once the user is connected to the AWS management console, they assume the specific AWS role and policy and can perform only authorized operations on the AWS platform. Every session is recorded and will be valid for a predefined period of time.

Notes:

- You can also configure connection components for AWS Console without AWS STS integration. For more information, refer to *Configuring PSM Connection Components for Web Applications*, page 659.
- This connection component cannot run with PSM installed on Windows that is FIPS enabled.

To Configure a PSM Connection Component for AWS Console with STS

The PSM Connection Component for AWS Console with STS is created automatically during PSM installation. In order to use this connection component, you must create two privileged accounts, as described in the following procedure.

1. Create a privileged account that contains the Secret Access Key and Access Key ID. This account will be used as the logon account for the AWS console.

Note: This account holds the Secret Access Key and Access Key ID, which are used to generate the temporary credentials. These keys must be attached to an AWS policy that grants permission to call the AssumeRole AWS API command.

- i. In the Accounts page, click **Add Account**; the Add Account page appears.
 - ii. From the Safe drop-down list, select the Safe where the account will be stored.
 - iii. From the Device drop-down list, select **Cloud Service**.
 - iv. From the Platform Name drop-down list, select **Amazon Web Services (AWS) Access Keys**; the properties for this type of account appear automatically.
 - v. In **AWS Access Key ID**, specify the Access Key ID.
 - vi. In **Password**, specify the Secret Access Key.
 - vii. Click **Save**; the new account is added and the Account Details page appears.
2. Create a privileged account that contains the AWS role definition and/or AWS policy. This account defines the user on the remote machine.
 - i. In the Accounts page, click **Add Account**; the Add Account page appears.
 - ii. From the Safe drop-down list, select the Safe where the account will be stored.
 - iii. From the Device drop-down list, select **Cloud Service**.
 - iv. From the Platform Name drop-down list, select **Amazon Web Services (AWS)**; the properties for this type of account appear automatically.
 - v. In **Username**, specify the account's username. This property is optional.

Note: To use the AWS CPM Plug-in to change and/or reconcile credentials, you must specify a username.
 - vi. In **Password**, specify the account's password. This property can be left empty.

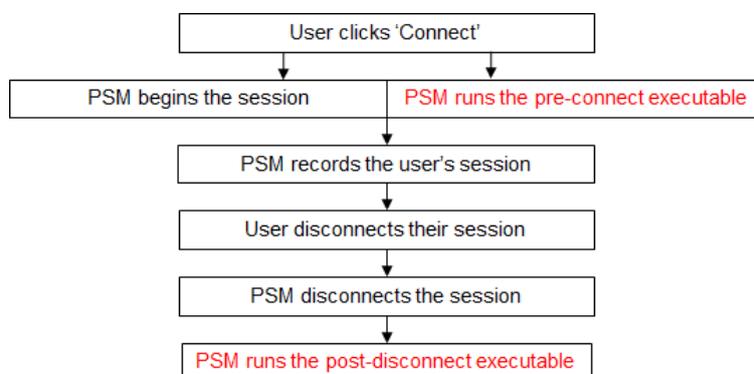
- vii. In **Address**, specify **aws.amazon.com**. This property is required.
- viii. In **AWS ARN Role**, specify the role amazon resource name (ARN) defined in AWS. This is a globally unique identifier for roles which includes the AWS account id and role name -
`arn:aws:iam::<aws_account_id>:role/<AWS_Role_Name>`
- ix. In **AWS Policy**, specify the AWS set of permissions policy. You can define a set of permissions in a JSON format (without carriage return) for the user. The user permissions in AWS console will be derived from that policy or will be unified with the AWS role permissions if the AWSARNRole attribute is populated.
 For more information about generating the AWS policy, refer to the AWS website.
- x. Click **Save**; the new account is added and the Account Details page appears.
- xi. Associate the logon account that you created in step 1 with this account:
 - a. In the CPM tab, in the Logon Account section, click **Associate**; the Associate Account window appears. This window lists the frequently used accounts. If the account you require does not appear in this list, search for the required account.
 - b. Select the account to associate, then click **Associate**; the selected account is linked to the account and its details are listed in the CPM tab.

Adding Custom Code in Session Pre-connection and Post-disconnection Phases

In some scenarios, there is a need to run specific custom code before the PSM session begins or after it ends. For example, when there is a need to get a session's start time and the connecting user, and store these details in an external system. In another example, your organizational policy requires the security team to be notified when the session is ended, to trigger certain activities and workflows.

The PSM enables you to create and run your own custom code in the pre-connection and post-disconnection phases, while exposing session and account properties (such as target address, username, session start time, etc.).

The following diagram shows where the executable files can be inserted into the PSM session flow:



After a user clicks 'Connect' to connect to a remote machine through the PSM, the session begins. At this point, the PSM searches for the pre-connect executable and if the PSM finds it, runs it. At this point, the PSM session is already running and does not wait for the pre-connect executable to finish. After the user has finished their session and disconnects, the PSM searches for the post-disconnect executable and, if the PSM finds it, runs it.

The PSM can identify and send any of the following system and account properties:

- **Account properties** – Any account property, such as the Address, DeviceType, PolicyID, AccountSafeName and UserName.
- **Associated accounts** – Details about associated logon and reconcile accounts, such as their address.
- **Client specific and user parameters** – The Client Specific and User Parameters settings for the connection client used to connect the session.
- **Session parameters** – Session properties such as PSMSessionUUID, PSMSessionStartTime, PSMSessionEndTime (only in the post-disconnect executable), AccessReason, PSMSourceAddress, SelectedConnCompID, PSMClientApp,
- **PSM properties** – Details of the PSM server, such as PSMComponentFolder and PSMComponentsLogFolder.
- **User properties** – Properties of the Vault user who is connected to the PWWA, such as HomeEmail, BusinessEmail and OtherEmail.
- **Ticketing properties** – Properties of the ticketing integration such as TicketID, TicketingSystem and TicketingAudit.

Your custom code is defined in executable files that are stored in the PSMComponents folder with the following names:

- **PSMPreConnect.exe** – The executable file that specifies the custom code to be run **when** the user is connected to the remote machine.
- **PSMPostDisconnect.exe** – The executable file that specifies the custom code to be run **after** the user is disconnected.

The PSM searches for these executable files automatically at the beginning and end of every session, and does not need to be configured for this. However, these files are not included in the PSMComponents folder by default and must be placed there manually.

The following procedures describe how to prepare these files so that the PSM can run user exits successfully.

Configuring the PSM to run the Customized Pre-connection and Post-Disconnection Phases

For all PSM connections, the PSM will check for executables called **PSMPreConnect.exe** and **PSMPostDisconnect.exe** in the PSM components folder. If the PSMPreConnect.exe is found, the PSM will run it when the session starts and if the PSMPostDisconnect.exe is found, the PSM will run it once the session is disconnected. These pre-connection and post-disconnection phases are run automatically if these files are in the PSM components folder, and no other configuration is required.

Creating Customized Pre-connection and Post-disconnection Phases

In order to access the provided session parameters, the PSM uses the **PSMUserExitAPI** which provides parameters for retrieving and logging utilities. You can find additional information about this API in the **PSMUserExitUtils.h** file in the installation package.

Files

The following files are included in the PSM installation package for pre-connection and post-disconnection phases.

Name	Description
PSMUserExitUtils.dll	Provides utilities, such as access to session parameters and writing in logs.
PSMUserExitUtils.lib	The import library for PSMUserExitUtils.dll.
PSMUserExitUtils.h	The PSMUserExitUtils.lib header file.

Prerequisites

Before creating pre-connection and post-disconnection phases, make sure of the following:

- The PSM server is installed and working properly.
- The pre-connect phase file is called PSMPreConnect.exe and is stored in the PSM components folder.
- The post-connect phase file is called PSMPostDisconnect.exe and is stored in the PSM components folder.
- PSMUserExitUtils.dll is in the same folder as the PSMPreConnect.exe and PSMPostDisconnect.exe,

Linkage

You can link PSMUserExitUtilsAPI.dll in either of the following ways:

- Using the PSMUserExitUtils.lib import library and PSMUserExitUtils.h (recommended).
- Using the LoadLibrary API function.

Usage Example

```
// Demonstration of a specific (PSMSessionStartTime) session parameter
retrieving using PSMUserExitAPI
int main()
{
    char* szPropertyName = "PSMSessionStartTime";
    char  szLogMessage[LOG_MESSAGE_MAX_SIZE];
    char* szPropertyValueBuffer = NULL;
    DWORD dwPropertyValueLength;
    int    nPropertyNameLength = strlen(szPropertyName);
    int    nRc = PSM_RC_SUCCESS;

    // Validate that the DLL was successfully loaded
    if (!InitializeUserExitUtils())
    {
        nRc = PSM_RC_FAILURE;
    }
    else
    {
```

```

        // Using PSMUserExitAPI to retrieve the property value's length in
        // order to know how much memory is needed to store the value
        // If the property doesn't exist, the function will fail
        if (!GetSessionPropertyBufferLength(szPropertyName,
nPropertyNameLength, &dwPropertyValueLength))
        {
            // Create the error log
            sprintf_s(szLogMessage, LOG_MESSAGE_MAX_SIZE, "%s was not
found", szPropertyName);
            // Write the log using PSMUserExitAPI
            WriteLog(szLogMessage, strlen(szLogMessage), 0);

            nRc = PSM_RC_FAILURE;
        }
        else
        {
            // create the trace log
            sprintf_s(szLogMessage, LOG_MESSAGE_MAX_SIZE,
"GetSessionPropertyBufferLength for %s has succeeded, length: [%lu]",
szPropertyName,dwPropertyValueLength);
            // Write the log using PSMUserExitAPI
            WriteLog(szLogMessage, strlen(szLogMessage), 1);

            // Allocate the required memory for storing the session proptry
            // value
            szPropertyValueBuffer = new char[dwPropertyValueLength];

            // Getting the session property's value (the
            // szPropertyValueBuffer will be filled) using PSMUserExitAPI
            if (GetSessionProperty(szPropertyName, nPropertyNameLength,
szPropertyValueBuffer, dwPropertyValueLength))
            {
                // Create the trace log
                sprintf_s(szLogMessage, LOG_MESSAGE_MAX_SIZE, "%s = [%s]",
szPropertyName, szPropertyValueBuffer);
                // Write the log using PSMUserExitAPI
                WriteLog(szLogMessage, strlen(szLogMessage), 1);
            }

            delete[] szPropertyValueBuffer;
        }

        // Release the allocated resources in the DLL (called automatically
        // when the DLL is unloaded from the process)
        if (!TerminateUserExitUtils())
        {
            // Create the error log
            sprintf_s(szLogMessage, LOG_MESSAGE_MAX_SIZE, "%s %s User exit
utils termination has failed",CYBER_ARK_PSRs, CLASS_NAME);
            // Write the log using PSMUserExitUtils
            WriteLog(szLogMessage, strlen(szLogMessage), 0);
        }

        // Create the trace log
        sprintf_s(szLogMessage, LOG_MESSAGE_MAX_SIZE, "%s %s User exit
ended", CYBER_ARK_PSRs, CLASS_NAME);
        // Write the log using PSMUserExitUtils
        WriteLog(szLogMessage, strlen(szLogMessage), 1);
    }

    return nRc;
}

```

Pre-connect Phase Return Codes

The return code of the pre-connect phase can affect the session, as described in the following table:

Code	Description
0 (zero)	The pre-connect phase ended successfully and will not affect the PSM session.
Number other than 0 (zero)	The user exit failed. The session will be stopped and a relevant message will be sent to the user.

There are no post-connection phase return codes.

Logging

The pre-connection and post-disconnection phases executed by the PSM create the following dedicated log files in the PSMLogs\Components folder:

- The **<session-UUID>.PreConnect.log** file contains activities that were carried out during the pre-connection phase.
- The **<session-UUID>.PostConnect.log** file contains activities that were carried out during the post-disconnection phase.

A new log file is started for each session.

These log files are rotated to the **old** folder in the same way as all other PSM log files. For more information about PSM logs rotation, refer to *PSM Activity Logs*, page 581.

Notes:

- The above log files are created automatically when messages with an "error" level are reported by the preconnect or post-disconnect phases. To enable these logs for tracing purposes, enable TraceLevels in the PSM General Setting in the PVWA system configuration. For more information about enabling the PSM component logging, refer to *Configuring the PSM Log Files*, page 684.
- Error messages with an "error" level also appear in the PSM trace and console logs.

Troubleshooting

The following troubleshooting options guide you through the main issues that may prevent the PSM user exits from working properly.

The user exit didn't run

- Solution 1: The executable file may not be saved in the PSMComponents folder. Make sure that this file is in the PSM installation folder where the PSM can find it. A relevant message is written in the log file.
- Solution 2: The name of the executable file may be spelled incorrectly. Make sure that this file is called either PSMPreConnect.exe or PSMPostDisconnect.exe. A relevant message is written in the log file.
- Solution 3: The details in the executable file may not be specified correctly. Make sure that you have specified the account/session/user parameters correctly. For more information, refer to the documentation above.

Configuring the Privileged Session Manager

In Web Access Options, the **General Settings** parameters in the **Privileged Session Management** section, define how the PSM will function. These parameters are divided into five groups:

- **Search properties** – These properties define the password and recording properties that will be searched when a search for session recordings is initiated.
- **Server settings** – These parameters define the general PSM server settings.
- **Session settings** – These settings define single sessions and the way that PSM handles recordings that cannot be uploaded to the Vault.
- **Recorder settings** – These settings configure the session recorder.
- **Connection client settings** – These parameters configure connection clients.

Searching for Recorded Sessions

The parameters in **SearchProperties** define the password and recording properties that will be searched when a search for session recordings is initiated:

- The **MaxRecords** parameter specifies the maximum number of session recordings that will be included in the Recordings search results. By default, 1000 recordings will be included in the search results.
- The **Recording** parameters define the recording properties that will be searched.
- The **Password** parameters define the password properties that will be searched.

Managing the PSM Server

The following parameters, in **Server Settings**, configure PSM server activities:

- The **MaxConcurrentTSSessions** parameter specifies the maximum number of allowed concurrent PSM sessions.
- The **MaxConcurrentUploaders** parameter specifies the maximum number of allowed concurrent processes to upload recording files to the Vault.
- The **ConfigurationRefreshInterval** parameter specifies the interval in seconds between each configuration refresh process.
- The **DisableExceptionHandling** parameter, in the Advanced Settings, determines whether or not a crash dump will be created when a system error occurs.
- The **ShutdownTimeout** parameter, in the Advanced Settings, specifies the maximum time in seconds to wait for internal jobs to finish when shutting down the server.

Configuring Privileged Sessions

The following parameters in **Session Settings** configure privileged sessions:

- The **MaxSessionDuration** parameter determines the maximum duration of the session, in minutes. This can be specified as a general PSM parameter or in a specific platform.
Note: When users log off from the remote Windows machine, the sessions on both the PSM and the remote machine are ended. However, when users disconnect the session by clicking **Close** or if the **MaxSessionDuration** parameter has expired, the PSM session is automatically ended, but the session on the remote machine continues running. The next time they log onto the same remote machine through the PSM, they will continue the same session as before. To prevent this, make sure that the Terminal Server is configured to end disconnect sessions after a specific time period.
- The **WarningDisconnectionInterval** specifies the number of minutes before the user's session will be disconnected that a warning message about the disconnection will be displayed.
- The **EndUserMessageTimeout** parameter specifies the maximum number of seconds that end user messages will be displayed.

Uploading Recorded Sessions to the Vault

The following parameters in **Session Settings** determine how the PSM handles retries when the Vault is not available and recordings cannot be uploaded.

- The **DelayBetweenUploadRetries** parameter specifies the delay in seconds between upload retries to the Vault.
- The **MaxUploadRetries** parameter specifies the maximum number of uploading retries to the Vault.

Managing Recording Sessions

The following parameters, in **Recorder Settings**, define how the PSM will manage recordings:

- The **LocalRecordingsFolder** parameter specifies the name of the local folder where recordings will be saved until they are uploaded to the Vault. By default, recordings are temporarily stored in the PSM installation folder.

Configuring the PSM Log Files

The types of messages included in the PSM log files are determined by the **TraceLevels** parameters in the Connection Client Settings, as follows:

The **PSMTrace.log** is configured by the following parameters in **Server Settings**:

- The **LogRotationSize** parameter defines the maximum size in MB of the log file before it is rotated to another location, and a new log file is started.
- The **TraceLevels** parameter sets the debug level of the PSM Server.

A new log file is created for each session for the recorder and the connection client. The trace levels for these files are specified in the following parameters:

- The **<SessionID>.Recorder.log** is configured in the Recorder Settings.
- The **<SessionID>.<connection client >.log** is configured in the Connection Client Settings.

For more information about logging for the PSM Recorder, refer to *PSM Activity Logs*, page 581.

Configuring the Server Connection Details

Configuring the PSM Server Details

The PSM server connection details determine how the PWWA will access the PSM server. You can configure as many PSM servers as you need.

The following parameters in the **Configured PSM Servers** parameters define the PSM server details:

- The **Address** parameter specifies the address of the PSM server machine used by passwords associated with the platform that uses this PSM server.
- The **Port** parameter specifies the port used to access the PSM Server machine used by passwords associated with the platform that uses this PSM server.
- The **Safe** and **Folder** parameters specify the location where the password of the logon account for the PSM Server is stored, and the **Object** parameter specifies the name of the password.

Configuring Secure Remote Access using a Remote Desktop Gateway

The PWWA can be configured to enable secure access to a remote machine through a Remote Desktop gateway (TSGateway).

Before Configuration:

- Make sure that a Remote Desktop Gateway is installed for the PSM Server. If the Remote Desktop Gateway is not installed on the PSM server machine, make sure that the machine where it is installed has RDP network access to the PSM machine. For more information, refer to Microsoft documentation.
- Make sure that the client machine meets the system requirements for the Remote Desktop Gateway:
 - Windows Vista / 2008 / XPSP3
 - For XP lower than SP3, make sure that RDP 6.1 is installed

- Make sure that the RD Gateway certificate is trusted so that users can access the machine through the gateway.

In the PVWA:

1. In the System Configuration page, click **Options**; the Web Access Options are displayed.
2. In the **Privileged Session Management** parameters, display the **Configured PSM Servers**, and select the PSM Server for which you will define the Remote Desktop Gateway.
3. In the **Connection Details** section, select **TS Gateway**.

If the RD Gateway is installed on the PSM server machine, the PSM parameters will be configured by default. However, if the RD Gateway is installed on a different machine, specify the following parameters:

- The **Address** parameter specifies the address of the Remote Desktop gateway machine used by passwords associated with this platform.
 - The **Domain** parameter specifies the name of the domain of the Remote Desktop gateway machine that will be used to connect to the remote machine.
 - The **Safe**, **Folder**, and **Object** parameters specify the location of the password for the logon account for the Remote Desktop gateway.
Note: This logon account is retrieved from the Vault by the internal PVWA application user (by default, PVWAAppUser). Therefore, make sure that the application user has permissions to retrieve this account.
 - The **Enable** parameter determines whether or not the Remote Desktop gateway is enabled.
4. As connections with RD Gateway are only supported when connecting with ActiveX, configure the PSM to always use ActiveX to connect to remote machines:
 - In the **Privileged Session Management UI** parameters, set **ConnectPSMWithRDPActiveX** to **Always**.

Accessing PSM Recording Sessions Directly

Users can generate direct URL links to PSM session recordings from activity logs and third party log systems, such as SIEM, by adding the ID of a session recording that appears in the audit log to a dynamic link that connects authorized users to PSM events. These recordings can be accessed directly by authorized users and circumvents the need to search for a specific session recording event.

For more information about creating links to open PSM recording sessions directly, refer to *Accessing Privileged Session Recordings*, page 576.

Disaster Recovery

The PSM benefits from Disaster Recovery features which provide seamless productivity during a failover. For more information about Disaster Recovery in the Vault, refer to *CyberArk Disaster Recovery Vault*, page 885.

Transparent Failover

As soon as the Production Vault cannot be reached by the PSM, the failover process begins in the DR Vault transparently, and no human intervention is required.

The IP address of both the Vault and the DR Vault can be specified in the Vault.ini configuration file. When the PSM cannot reach the Vault specified by the first IP address, it transfers automatically to the Vault specified by the second IP address, which is the DR Vault.

To Configure Transparent Failover

1. In the Vault.ini file, in the **Address** parameter, specify the IP addresses of the Vault and the DR Vault, separated by commas, as shown in the following example:

```
Address=1.1.1.102,1.1.1.232
```

The above example indicates that the IP address of the Production Vault is 1.1.1.102 and the IP address of the DR Vault is 1.1.1.232.

2. Add the **SwitchVaultAddressTimeOut** parameter.

This parameter specifies the number of seconds that the PSM will try to access additional Vault IP addresses after the initial timeout to the current Vault, specified in the **Timeout** parameter, expires.

If this parameter is not added, the default value of three seconds will be applied.

3. Save the Vault.ini file and close it.

Replicating PSM Users' Passwords

As each PSM user requires a credential file to authenticate to the Vault, it is essential that the credential files in the DR Vault are always identical to those on the Production Vault. At regular intervals, the PSM automatically initiates a password change in the Vault and in the corresponding credential file for the PSM. In order for the PSM user in the DR Vault to access the Vault and continue working seamlessly in a Disaster Recovery situation, the user's new credentials must be replicated to the DR Vault whenever they are changed.

This is configured by the following parameter in the CreateCredFile utility:

- **DisableSyncPasswordToDR** – Whether or not passwords in user credential files will be replicated to all DR sites before they are replaced. The default value of this parameter is 'No', which makes sure that user credential files on all DR sites (if they exist) are synchronized with the Production Vault and that users will be able to continue working with the Vault seamlessly after a failover. If this parameter is changed to 'Yes', passwords will be replaced in credential files regardless of whether or not they have been replicated to all DR sites.

Auditing in PSM

The Vault provides comprehensive auditing for every access to passwords and to privileged session recordings. Auditing information is displayed in a simple intuitive interface that includes the following information for each activity:

Information	Displays
User name	The name of the user who accessed the password or recording file.
Password or recording file	The name of the password or recording file that was accessed.
Destination address	The IP address/DNS of the target location where the password was used.
Protocol	The protocol used to access the target machine.
PSM ID	The unique ID of the PSM server.
Session ID	The unique ID of the PSM session that recorded this activity.
Session duration	The duration of the logon session.

Users can view an overall audit in the MONITORING page, and play privileged session recordings. Users can search for recordings using a free text search according to the properties that are associated with the privileged session (e.g. password, user, address, device, machine, or any other password keyword).

The screenshot shows the 'MONITORING' page in the PSM interface. The top navigation bar includes 'POLICIES', 'ACCOUNTS', 'MONITORING', 'APPLICATIONS', 'REPORTS', and 'ADMINISTRATION'. A user profile for 'DanielleB' is visible in the top right corner. The main content area is titled 'Search for Sessions' and contains several search filters: 'Search for Sessions:', 'Search for Commands and Events:', 'Search for sessions between:' (with date pickers), and 'Search for recordings:' (with radio buttons for 'Search for recordings' and 'Search for live sessions'). Below the filters, there are 'Search' and 'Clear' buttons. The 'Views' section on the left shows 'Live Sessions' and 'My Views' (with 'Search recordings: All recordings' selected). The main table, titled 'Search recordings: All recordings', displays the following data:

User	Client	Account User Name	Account Address	Account Policy ID	Start	Duration	Vid
Ron	RDP	Administrator	1.1.1.12	WinDesktop.local	8/8/2013 4:07:57 PM	00:03:10	28
Ron	RDP	Administrator	1.1.1.12	WinDesktop.local	8/8/2013 4:02:18 PM	00:03:10	63

At the bottom of the interface, there is a pagination control showing 'Page 1 of 1' and a status bar indicating 'Displaying recordings 1 - 2 of 2'.

In addition, more detailed audit information is available for individual password use.

Privileged Session Manager SSH Proxy

This chapter introduces you to the PSM SSH Proxy (PSMP), which preserves the benefits of PSM such as isolation, control and monitoring, whilst enabling users to connect transparently to target UNIX systems from their own workstation without interrupting their native workflow.

This chapter includes the following:

- Introduction
- Architecture
- Configuring the PSMP
- Defining Platforms for Assorted Scenarios
- Auditing
- Disaster Recovery

Introduction

The Privileged Session Manager SSH Proxy (PSMP) enables organizations to secure, control and monitor privileged access to network devices. Using the Vault technology, it manages access to privileged accounts at a centralized point and facilitates a control point to initiate privileged sessions. The PSMP pinpoints users who are entitled to use privileged accounts and initiate a privileged session, when, and for what purpose. The PSMP can record all activities that occur in the privileged session in a compact format. Text recordings are stored and protected in the Vault server and are accessible to authorized auditors. The PSMP also provides privileged Single Sign-On capabilities and allows users to connect to target devices without being exposed to the privileged connection password.

The PSMP separates end users from target machines, and initiates privileged sessions without divulging passwords, maintaining the highest level of security that is typical to all CyberArk components. In addition, the PSMP can display a broad overview of all activity performed on every privileged account, without exception. All activities are fully monitored and meet strict auditing standards.

The PSMP enables end users to connect transparently to target UNIX systems that use either SSH or Telnet protocol, including SSH tunneling. Users can also copy files to and from remote machines through the PSMP using a native SCP command. Accounts used on target machines are stored in the Vault, from where they are retrieved by the PSMP. These accounts include the user name and password or SSH Key that authenticate the user on the target system, providing a privileged SSO session. Alternatively, when privileged passwords are not managed in the Vault, accounts may be stored in the Vault without a password, which enables a non-privileged SSO session. Users authenticating to target systems with a non-privileged SSO session are required to supply the target system password manually to enable the PSMP to create a connection to the remote system.

The end user connects to the PSMP using an intuitive command line, which includes the target device, target user, vault user, and vault password. The PSMP prompts for any missing parameters. Users authenticate once to the Vault and do not need to specify additional connection passwords. The end user can launch a session to a target system from their own workstation without interrupting their workflow, and work efficiently in the same way as they would without the PSMP.

The PSMP ensures that only authorized users can connect to the target systems by first authenticating them to the Vault. The CyberArk or LDAP password authentication can be used, as well as stronger methods like Radius or SSH keys authentication. With SSH keys authentication, the private SSH keys can be stored on smart card devices, which facilitates an even stronger authentication policy.

The PSMP can also be configured to integrate with Microsoft's Active Directory (AD) to provision users transparently on UNIX systems, streamlining user management and reducing administrative overhead. In addition to automatic user provisioning, this CyberArk solution benefits from all standard CyberArk security and management features, including access control and auditing. Users can log onto a UNIX machine using their AD credentials as their user is automatically synchronized with a corresponding user in the Vault. Likewise, existing groups in AD directories are automatically synchronized with a corresponding group in the Vault. Users have immediate access to UNIX machines, based on their AD permissions and groups, facilitating an uninterrupted workflow and maintaining productivity. For more information, refer to *Integrating with AD Bridge Capabilities*, page 724.

The combination of user authentication that uses the SSH keys residing on smart card devices with Active Directory integration allows transparent user provisioning on Unix systems, based on their strong authentication to the Vault. For information about configuring authentication methods for PSMP, refer to *Configuring Authentication Methods*, page 693.

Monitoring Privileged Sessions

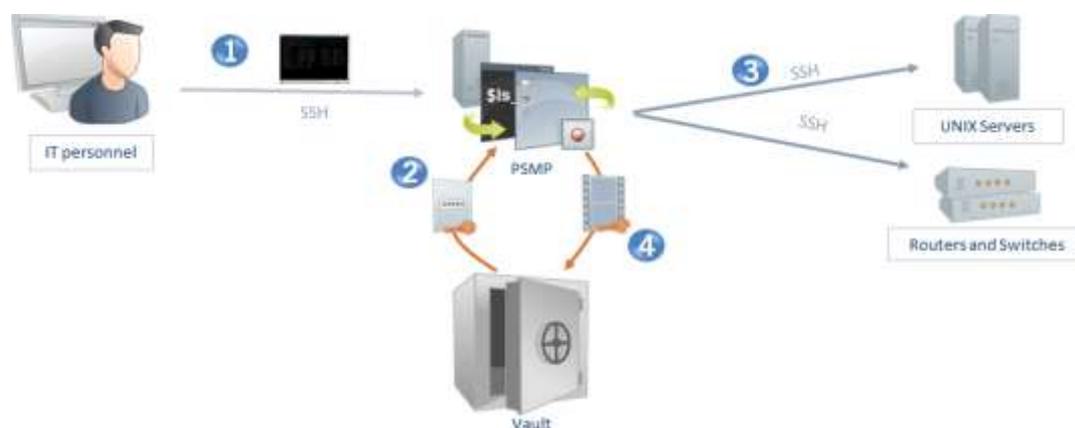
PSMP enables organizations to secure, control and monitor privileged access to network devices by using the Vault technology to manage privileged accounts and record all IT administrator privileged sessions on remote machines.

PSMP provides the following features:

- **Recorded Privileged Sessions** – All the activities in each privileged session can be recorded in text format, and stored in the Vault, compressed, for future auditing. These recordings are transparent to users and cannot be bypassed.
- **Privileged Remote Access** – Only authorized users can initiate privileged sessions to the PSMP machine using SSH protocol. This meets standards for secure remote access by ensuring encrypted sessions and by not requiring the corporate firewall to be opened to additional native protocols.
- **Privileged Single Sign-On** – Users connect transparently to remote target applications and systems via the PSMP.
- **Centralized Management** – In the PVWA, users can see all the recordings archives, where auditors can retrieve and view comprehensive recordings of privileged sessions. Search features enable auditors to locate specific recordings.
- **Transparent Integration** – The PSMP can be integrated transparently and seamlessly into existing enterprise infrastructures, including a variety of authentication, monitoring, ticketing, and workflow systems.

Architecture

The following diagram shows the different components of the PSMP solution and how they interact.



Workflow and Architecture

Users connect to the remote target system from their native client through the PMSP using a standard SSH port (step 1). The PSMP machine authenticates the user to the Vault and retrieves the privileged credentials, according to the user's permissions in the Safe, (step 2) that are required to connect to the target system (step 3). The session to the target system can be an SSH session or a Telnet session based on the platform definitions. During the session, each keystroke and command is recorded in the Vault for immediate auditing. At the end of the session, a text recording of the entire session is stored in the Vault (step 4).

Configuring the PSMP

Users can either access remote devices transparently or by specifying the user name and password before connecting. This enables enterprises to record sessions for which accounts are not managed in the Vault, and makes sure that only authorized users can log onto the target device. In addition, it facilitates complete monitoring without needing to provision privileged accounts.

Accounts that are used to connect to remote target machines must be created in the Vault so that the PSMP can retrieve and use them in order to authenticate the user.

Configuring Accounts for Privileged Single Sign-On

Privileged accounts can provide the following connection methods:

- **Privileged SSO** – Users connect to target machines transparently, without being prompted for a username or password. For more information, refer to *To Add an Account for Privileged SSO* below.
- **Non-privileged SSO** – Users are prompted for logon credentials before they can connect to the target machine.
 - **Accounts can be created with a predefined username, but no password** - Users must supply the password before they can connect to the target machine.
 - **Accounts can be created without a username or a password** - Users must supply these logon credentials before they can connect to the target machine.

For more information about non-privileged SSO accounts, refer to *To Add an Account for Non-privileged SSO*, page 693.

By default, these connection methods are set at platform level so that it is applied automatically to all the accounts associated with a specific platform.

To Add an Account for Privileged SSO

1. Log onto the PVWA as a user with permission to add accounts.
2. In the **Accounts** page, click **Add Account**; the Add Account page appears.
3. From the Safe drop-down list, select the Safe where the account will be stored.
4. From the Device drop-down list, select the type of device where the account will be used; the Platform Name drop-down list appears. This lists all the platforms that can be specified for the selected device.
5. From the Platform Name drop-down list, select a platform that will enable the user to connect to the target machine **with Privileged SSO**; required and optional properties for the type of account that you have selected will appear automatically.
6. Specify the address, username, and other required properties and, if possible, the optional properties.

Note: In the Address property, specify either the IP address or the DNS.
7. In the **Password** field, specify the password that is required to log onto the remote device with this account.
8. In the **Confirm Password** field, specify the password again.

9. To disable automatic password management by the CPM for this password, select **Disable automatic management for the account**. You can also enter a reason for doing this.

Note: The CPM user must be an owner of the Safe where the password will be stored in order for the password to be managed by the CPM.

10. Click **Save**; the new account is added. Users who use this account will be able to log onto the target device transparently, with no intervention.

To Add an Account for Non-privileged SSO

1. Log onto the Password Vault Web Access as a user with permission to add accounts.
2. In the **Accounts** page, click **Add Account**; the Add Account page appears.
3. From the Safe drop-down list, select the Safe where the account will be stored.
4. From the Device drop-down list, select the type of device where the account will be used; the Platform Name drop-down list appears. This lists all the platforms that can be specified for the selected device.
5. From the Platform Name drop-down list, select a platform that will enable the user to connect to the target machine **with non-privileged SSO**; required and optional properties for the type of account that you have selected will appear automatically.
6. Specify the required properties.
Note: In the Address property, specify either the IP address or the DNS.
7. Optionally, specify the username.
8. Do not specify the password.
9. Click **Save**; the new account is added. Users who use this account will be prompted for user credentials before they can connect to the target device.

Configuring Authentication Methods

Users can authenticate to the PSMP using any of the following authentication methods:

- CyberArk password
- LDAP
- RADIUS
- SSH Key

The Vault administrator can enforce a specific authentication method for all users, or allow users to authenticate with any of the above authentication methods, according to the method that is configured for them in their Vault user account, without forcing a specific method. This is useful when different users in the organization use different authentication methods.

To Configure the User Authentication Method

1. Log onto the Password Vault Web Access as a user with permission to configure platforms.
2. Click **ADMINISTRATION**, then in the System Configuration page click **Options**; the Web Access Options are displayed.
3. Expand **Privileged Session Management**, then **General Settings**, and then **Server Settings**.
4. Select SSH Proxy Settings; the SSH Proxy Settings properties are displayed.
5. In **AuthenticationMethod**, specify the authentication method that the Vault will use to authentication PSMP users. Specify one of the following valid values:
 - Password
 - LDAP
 - RADIUS
 - SSH Key

Note: Before you can set this value to SSH Key, upgrade all the PSM and PSMP servers in your environment to v9.6.

 - Default – Enables users to authenticate with the authentication method that is configured for them, without forcing a specific method. This is the default value.
6. Click **Apply** to save the new configuration and stay in the same page,
or,
Click **OK** to save the new configuration and return to the System Configuration page.
7. Restart the PSMP service to apply the configuration changes:
At a command line, run the following commands:

```
/etc/rc.d/init.d/psmpsrv stop  
/etc/rc.d/init.d/psmpsrv start
```

Configuring Platforms to Enable Connections through the PSMP

In the PVWA:

1. Log onto the Password Vault Web Access as a user with permission to configure platforms.
2. Click **ADMINISTRATION** to display the **System Configuration** page, then click Platform Management to display a list of supported target account platforms.
3. Select the platform in which you will enable the PSMP, then click **Edit**; the settings page for the selected platform appears.
4. Expand **UI & Workflows**, and then expand **Connection Components**, and make sure that the **PSMP-SSH** Connection Component is defined and enabled.

5. Click **Apply** to save the new parameter values and stay in the same page,
or,
Click **OK** to save them and return to the System Configuration page.

In the PrivateArk Client:

6. Log onto the PrivateArk Client with an administrative user.
7. Open the **PVWAConfig** Safe and retrieve and open the **Policies.xml** configuration file.
8. Identify the node that defines the platform you enabled for PSMP in the PVWA.
9. In the **PrivilegedSessionManagement** parameters, set the **Enable** property to **Yes**.

The following example shows how to enable connections through the PSMP for the **CiscoSSH** platform:

```
<Policy ID="CiscoSSH" PlatformBaseID="CiscoSSH"
PlatformBaseType="Cisco" PlatformBaseProtocol="SSH">
...
<PrivilegedSessionManagement Enable="Yes">
  <ConnectionComponents>
    <ConnectionComponent Id="PSMP-SSH" />
    ...
  </ConnectionComponents>
...
</Policy>
```

10. Save the **Policies.xml** file and return it to the **PVWAConfig** Safe.

On the PSMP machine:

11. Restart the PSMP service to apply the configuration changes:
At a command line, run the following commands:

```
/etc/rc.d/init.d/psmpsrv stop
/etc/rc.d/init.d/psmpsrv start
```

Configuring Platforms for Copying Files with PSMP

In the PVWA:

1. Log onto the Password Vault Web Access as a user with permission to configure platforms.
2. Click **ADMINISTRATION** to display the **System Configuration** page, then click **Platform Management** to display a list of supported target account platforms.
3. Select the platform in which you will enable the PSMP to copying files securely, then click **Edit**; the settings page for the selected platform appears.
4. Expand **UI & Workflows**, and then expand **Connection Components**, and make sure that the **PSMP-SCP** Connection Component is defined and enabled.
5. Click **Apply** to save the new parameter values and stay in the same page,
or,
Click **OK** to save them and return to the System Configuration page.

Configuring SSH Key Authentication to Target Systems

The Unix via SSH Keys platform enables you to add accounts that will authenticate to target systems with SSH key authentication instead of password authentication. If additional SSH Policies are needed, duplicate the Unix via SSH Keys built-in platform.

Configuring Login Sequences

As different devices may have different logon processes, you can configure how the PSMP will logon to a device using the `AutoLogonSequence` parameter. This parameter defines a multiline sequence that is used by PSMP during the automatic logon process and contains regular expression prompts and responses that define the process. The regular expressions can include dynamic values that the PSMP reads from the account properties, user parameters, or client-specific parameters, in this order. You can override this configuration at platform level.

Configuring Telnet Connections

The logon process using Telnet may differ depending on the device that you are connecting to. To enable Telnet connections, define the logon sequence to suit your device using the `AutoLogonSequence` parameter.

To Specify a Logon Sequence for a Telnet Connection in your Platform

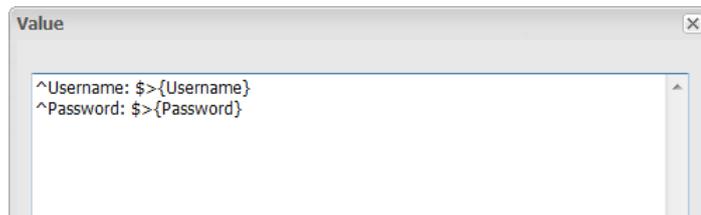
Note: The PSM SSH Proxy works only with the PSMP-SSH connection component to perform SSH connections to targets. The configurations in the PSMP-SSH connection component affect all connections made with the PSMP. To change the configuration for some accounts, override the PSMP-SSH settings at platform level. For example, you can configure the PSMP-SSH connection component with a setting for SSH connections, such as an `AutomaticLogonSequenceWithLogonAccount` for SSH. To define this setting for Telnet, create a platform for Telnet connections that overrides `AutomaticLogonSequenceWithLogonAccount` with a value suitable for Telnet connections.

1. Log onto the Password Vault Web Access as a user with permission to configure platforms.
2. Click **ADMINISTRATION** to display the **System Configuration** page, then click **Platform Management** to display a list of supported target account platforms.
3. Select the platform to configure for PSMP, then click **Edit**; the settings page for the selected platform appears.

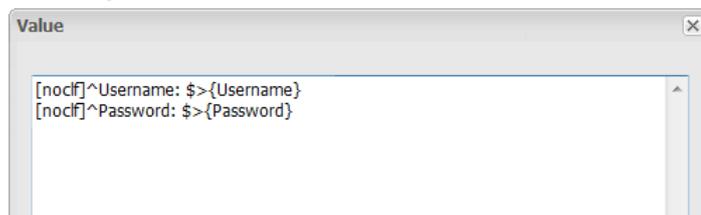
Note: If your platform was not configured to enable connections via PSMP, configure the platform to enable connections via PSMP first. For more information, refer to *Configuring Platforms to Enable Connections through the PSMP*, page 694.
4. Expand **UI & Workflows**, and then expand **Connection Components**; the Connection Components parameters are displayed with their default values.
5. Select **PSMP-SSH**, expand **Target Settings**, and then expand **Client Specific**; a list of Client Specific parameters appears.
6. Right-click **Client Specific** parameters, then select **Add Multiline Parameter**; a new parameter is added.
7. In the Properties list, set the value of the **Name** property to **AutoLogonSequence**.
8. In the Properties list, click the value of the **Value** property; the Value edit box appears.

- Specify the logon process, as shown in the following examples.

The following example shows a simple logon process that includes a username and password then logs the user on.



To prevent the client from adding a CRLF character (new line) to the end of the response, specify (nocrlf) at the beginning of the prompt, as shown in the following example:



In each line, the text to the left of the '>' (parenthesis) represents the regular expression for the prompt on the remote machine. The text to the right of the '>' (parenthesis) represents the PSM response, including a dynamic reference to an account property.

This response can include one or more dynamic references. The PSM reads these references in the following order: account properties, user parameters, then client specific parameters.

To specify '>' as a character in the prompt, use the character code **\x3e**.

- Click OK.**
- Click **Apply** to apply the new platform configurations,
or,
Click **OK** to save the new platform configurations and return to the System Configuration page.
- Restart the PSMP service to apply the configuration changes:

At a command line, run the following commands:

```
/etc/rc.d/init.d/psmpsrv stop  
/etc/rc.d/init.d/psmpsrv start
```

Using Logon Accounts for SSH and Telnet Connections

A logon account can be used to initiate sessions to machines that do not permit direct logon. When a logon account is associated with a privileged account, it will be used to log onto the remote machine and then elevate itself to the role of the privileged user.

As different types of machines might have different logon prompts or elevation commands, you can configure how the PSMP will perform the logon process and the elevation to the privileged account by using the **AutoLogonSequenceWithLogonAccount** parameter.

This parameter defines a multiline sequence that is used by the PSMP during the automatic sign-on process. It contains regular expression prompts and responses that define the logon process and subsequent activities. The regular expressions can include dynamic values that the PSMP reads from the account properties, user parameters, or client-specific parameters, in this order. You can override this configuration at platform level.

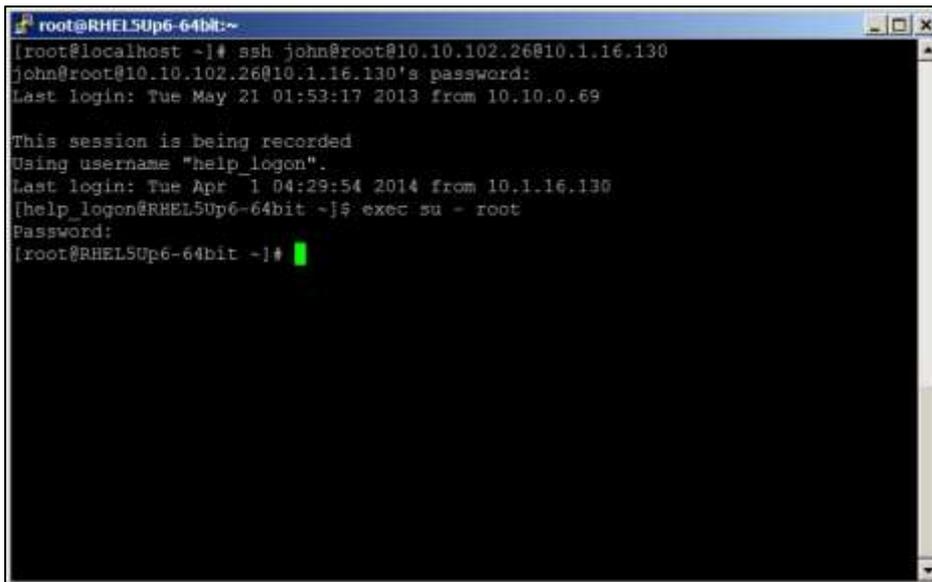
Note: For SSH connections, the logon account can use either password or SSH key authentication. If the logon account uses SSH key authentication, the associated privileged account must use password authentication.

The following example shows the process that takes place using a logon account.

Step 1: Link a logon account to the account that cannot be used for direct logon, but will be used to run sessions on the remote machine. The following screen shows the Account Details page of the root account that will be used to run sessions on a remote machine. In this scenario, this account cannot be used to log onto the remote machine, so the **UNIXSSH-help_logon-10.10.102.26** logon account has been associated with the account.



Step 2: The PSMP connects to the remote machine automatically using the associated **help_logon** logon account and elevates the user to the privileged account. After the user runs the PSMP connection command, a session is opened in the remote machine and the logon account is used to log on. In this example, after successfully logging on, the **help_logon** user issues the **su** command and elevates itself to the **root** user.



```
root@RHEL5Up6-64bit:~  
[root@localhost ~]# ssh john@root@10.10.102.26@10.1.16.130  
john@root@10.10.102.26@10.1.16.130's password:  
Last login: Tue May 21 01:53:17 2013 from 10.10.0.69  
  
This session is being recorded.  
Using username "help_logon".  
Last login: Tue Apr 1 04:29:54 2014 from 10.1.16.130  
[help_logon@RHEL5Up6-64bit ~]# exec su - root  
Password:  
[root@RHEL5Up6-64bit ~]#
```

To Define an Automatic Logon Sequence with a Logon Account

Note: The PSM SSH Proxy works only with the PSMP-SSH connection component to perform SSH connections to targets. The configurations in the PSMP-SSH connection component affect all connections made with the PSMP. To change the configuration for some accounts, override the PSMP-SSH settings at platform level. For example, you can configure the PSMP-SSH connection component with a setting for SSH connections, such as an `AutomaticLogonSequenceWithLogonAccount` for SSH. To define this setting for Telnet, create a platform for Telnet connections that overrides `AutomaticLogonSequenceWithLogonAccount` with a value suitable for Telnet connections.

1. Log onto the Password Vault Web Access as a user with permission to configure platforms.
2. Make sure that the Connection Client capabilities are configured for a logon account:

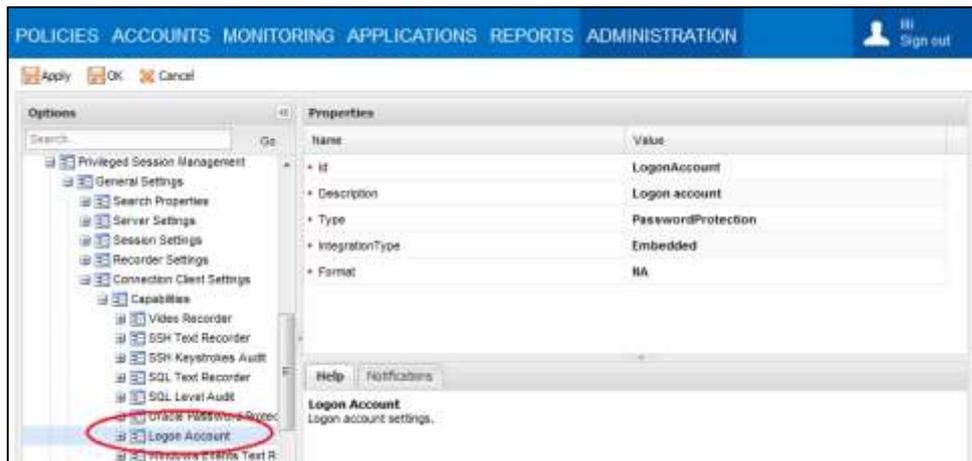
Note: The logon account capability is added automatically by the PSM installation. If your first PSM installation was PSM v7.0, enable the logon account capability manually as described below.

- i. Click **ADMINISTRATION**, then in the **System Configuration** page click **Options**; the Web Access Options are displayed.
- ii. Select and expand **Privileged Session Management**, then expand **General Settings**.
- iii. In **Connection Client Settings**, expand **Capabilities**.
- iv. Right-click Capabilities, then from the pop-up menu, select **Add Logon Account**; a new Logon Account parameter is created.

- v. In the Logon Account properties, make sure that the following property values are specified:

Property	Specifies
Id	LogonAccount
Description	LogonAccount
Type	PasswordProtection
IntegrationType	Embedded
Format	NA

These values are shown in the following window:



3. Specify the automatic logon sequence with the logon account:

- **For SSH connections:**

To change the default automatic logon sequence with logon account for all SSH connections that will be done with the PSMP-SSH connection component:

- i. Click **ADMINISTRATION**, then in the **System Configuration** page click **Options**; the Web Access Options are displayed.
- ii. Expand **Target Settings** and then expand **Client Specific**; a list of Client Specific parameters appears.
- iii. Select **AutoLogonSequenceWithLogonAccount**, then in the Properties list, click the value of the **Value** property; the Value edit box appears.
- iv. Specify the prompts and responses to include in the automatic logon process, using regular expressions and dynamic account properties to mimic the **exact** sequence that will be run on the remote machine.

As prompts differ according to machine, it is important to make sure that you write the prompt exactly as the machine requires.

Specify the command that will elevate the logon user to the user who will run sessions on the remote machine. Use regular expression prompts and responses with dynamic values, as shown in the following example:



In each line, the text to the left of the '>' (parenthesis) represents the regular expression for the prompt on the remote machine. The text to the right of the '>' (parenthesis) represents the PSMP response, including a dynamic reference to an account property.

This response can include one or more dynamic references. The PSMP reads these references in the following order: account properties, user parameters, then client specific parameters.

To specify '>' as a character in the prompt, use the character code `\x3e`.

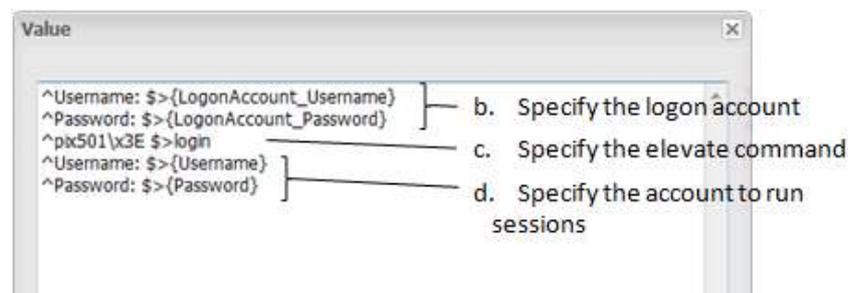
- **For Telnet connections:**

To define an automatic logon sequence with logon account for your specific platform. In the platform for Telnet connections, override the `AutoLogonSequenceWithLogonAccount` client specific parameter of the PSMP-SSH connection component as follows:

- i. Click **ADMINISTRATION** to display the **System Configuration** page, then click **Platform Management** to display a list of supported target account platforms.
- ii. Select the platform to use for Telnet connections with logon account, then click **Edit**; the settings page for the selected platform appears.
- iii. Expand **UI & Workflows**, and then expand **Connection Components**.
- iv. Select the **PSMP-SSH** in the Connection Components section.

Note: If your platform was not configured to enable connections via PSMP, configure the platform to enable connections via PSMP first. For more information, refer to *Configuring Platforms to Enable Connections through the PSMP*, page 694.

- v. Expand **Target Settings**, and then expand **Client Specific**; a list of Client Specific parameters appears.
- vi. Right-click on **Client Specific** parameters, then select **Add Multiline Parameter**; a new parameter is added.
- vii. In the Properties list, set the value of the **Name** property to **AutoLogonSequenceWithLogonAccount**.
- viii. In the Properties list, click the **Value** property; the Value edit box appears.
- ix. Specify the logon command that enable the logon account to log onto the remote machine.
- x. Specify the command that will elevate the logon user to the user who will run sessions on the remote machine.
- xi. Specify the username and password of the user who will run sessions on the remote machine.



In each line, the text to the left of the '>' (parenthesis) represents the regular expression for the prompt on the remote machine. The text to the right of the '>' (parenthesis) represents the PSMP response, including a dynamic reference to an account property.

This response can include one or more dynamic references. The PSMP reads these references in the following order: account properties, user parameters, then client specific parameters.

To specify '>' as a character in the prompt, use the character code **\x3e**.

4. Click **OK**; the logon sequence is displayed in the Value property as one line.
5. Click **Apply** to apply the new configurations,
 - or,
 - Click **OK** to save the new configurations and return to the System Configuration page.
6. Restart the PSMP service to apply the configuration changes:
 - At a command line, run the following commands:

```
/etc/rc.d/init.d/psmpsrv stop
/etc/rc.d/init.d/psmpsrv start
```

7. In the Account Details page of the account that will be used to run sessions on a remote machine, associate the account that will be used to log onto the remote machine.

For more information about adding a linked account to new and existing accounts, refer to *Linked Accounts*, page 220.

Troubleshooting

The client 'skips' characters while imitating the login sequence

1. In the relevant connection component, add the **SendRateValue** parameter in the Client Specific target settings.
2. Set the parameter value to higher than 100 milliseconds.
3. Save your configuration changes and restart the PSMP.

The following message appears: **PSMSH059E Failed to execute login sequence: Incorrect sequence defined in configuration, or network timeout occurred**

1. Make sure that the value of the **AutoLogonSequence** or the **AutoLogonSequenceWithLogonAccount** parameter is configured correctly.
2. Compare the specified login sequence with the login sequence from the text recording file after a session fails.
 - i. From the Client Specific target settings, remove the **AutoLogonSequence** or the **AutoLogonSequenceWithLogonAccount** parameter.
 - ii. Run the logon sequence again to make the client text record the session from the beginning.
 - iii. After the session fails, copy the prompts for the login sequence from the text recording file.
 - iv. In the Client Specific target settings, add the **AutoLogonSequence** or the **AutoLogonSequenceWithLogonAccount** parameter again.
 - v. Save your configuration changes and restart the PSMP.
3. If the specified login sequence is identical to the recorded text and the error message is still displayed, set the value of the **PromptTimeout** parameter to a much higher value. For example, **10000**. Then save your configuration changes and restart the PSMP.

Configuring Accounts to Provide Specific Connection Methods

Accounts can be configured to provide the following connection methods:

- **Privileged SSO** – Users connect to target machines transparently, without being prompted for a username or password..
- **Non-privileged SSO** – Users are prompted for logon credentials before they can connect to the target machine.
 - **Accounts can be created with a predefined username, but no password** - Users must supply the password before they can connect to the target machine.
 - **Accounts can be created without a username or a password** - Users must supply these logon credentials before they can connect to the target machine.

By default, you set Privileged SSO at platform level, so that it can be set automatically for all the accounts associated with a specific platform. You can override this setting at account level for individual accounts.

To Configure Privileged SSO at Platform Level

1. Log onto the Password Vault Web Access as a user with permission to configure platforms.
2. Click **ADMINISTRATION** to display the **System Configuration** page, then click Platform Management to display a list of supported target account platforms.
3. Select the platform to configure, then click **Edit**; the settings page for the selected platform appears.
4. Expand **UI & Workflows**, and then select **Privileged Session Management**; the PSM parameters are displayed with their default values.
5. Set the **EnablePrivilegedSSO** parameter to **Yes**.
6. Click **Apply** to save the new parameter values and stay in the platform settings page,
or,
Click **OK** to save them and return to the System Configuration page.

To Configure Platforms for Non-Privileged SSO

1. Log onto the Password Vault Web Access as a user with permission to configure platforms.
2. Click **ADMINISTRATION** to display the **System Configuration** page, then click Platform Management to display a list of supported target account platforms.
3. Select the platform to configure, then click **Edit**; the settings page for the selected platform appears.
4. Expand **UI & Workflows**, and then select **Privileged Session Management**; the PSM parameters are displayed with their default values.
5. Set the **EnablePrivilegedSSO** parameter to **No**.
6. Expand **Properties**; a list of **Required** and **Optional** properties are displayed.
7. From the **Required** properties list, delete **Username**.
8. In the **Optional** properties list, specify the following optional property:
 - Username
9. Click **Apply** to save the new parameter values and stay in the platform settings page,
or,
Click **OK** to save them and return to the System Configuration page.

Authenticating with your Personal Password

The PSMP can be configured to authenticate Vault users without prompting them for additional authentication credentials when they attempt to log onto a target machine using non-privileged SSO accounts.

This is relevant for Vault users whose credentials are stored in the following accounts:

- LDAP accounts
- RADIUS accounts
- CyberArk accounts

By default, this configuration is set at platform level, so that it can be applied automatically for all accounts associated with a specific platform.

To Configure Authentication with your Personal Password at Platform Level

1. Log onto the Password Vault Web Access as a user with permission to configure platforms.
2. Click **ADMINISTRATION** to display the **System Configuration** page, then click Platform Management to display a list of supported target account platforms.
3. Select the platform to configure, then click **Edit**; the settings page for the selected platform appears.
4. Expand **UI & Workflows**, and then select **Privileged Session Management**; the PSM parameters are displayed with their default values.
5. Set the **UsePersonalPassword** parameter to **Yes**.

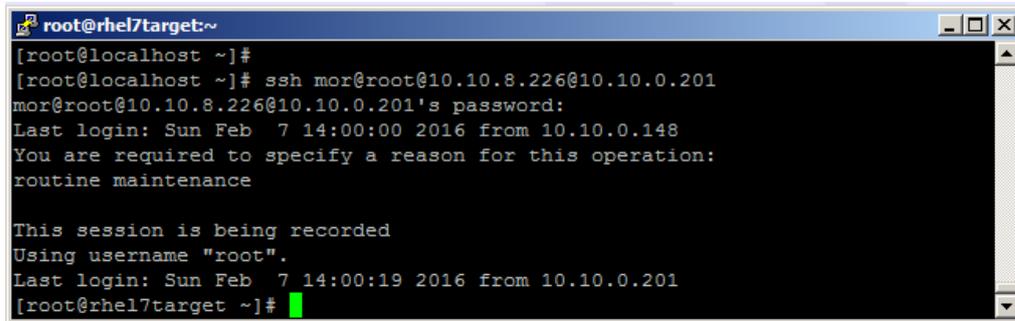
Note: This parameter can only be set to **Yes** if **EnablePrivilegedSSO** is set to **No**. If both **UsePersonalPassword** and **EnablePrivilegedSSO** are set to **Yes**, the PSMP will not authenticate the user.

6. Click **Apply** to save the new parameter values and stay in the platform settings page,
or,
Click **OK** to save them and return to the System Configuration page.

Specifying a Reason for Accessing Accounts

The 'Require users to specify reason for access' rule in the Master Policy determines whether or not users can only retrieve passwords or SSH keys after they specify a reason that explains their retrieval. If this rule is active, users are prompted to provide a reason before the remote session begins.

Note: When copying files through PSMP, users will not be prompted for a reason.



```

root@rhel7target:~
[root@localhost ~]#
[root@localhost ~]# ssh mor@root@10.10.8.226@10.10.0.201
mor@root@10.10.8.226@10.10.0.201's password:
Last login: Sun Feb  7 14:00:00 2016 from 10.10.0.148
You are required to specify a reason for this operation:
routine maintenance

This session is being recorded
Using username "root".
Last login: Sun Feb  7 14:00:19 2016 from 10.10.0.201
[root@rhel7target ~]#
  
```

Customizing the Prompt Message

You can customize the message that is displayed to prompt users for a reason before they can access accounts. The default message is "You are required to specify a reason for this operation".

To Customize the Prompt Message

1. Open the `basic_psmserver.conf` file.
2. In the **ReasonPromptCaption** parameter, specify the message that will be displayed, as shown in the following example:

```
ReasonPromptCaption="Please specify a reason for accessing this account"
```

3. Save the file and close it.

Displaying the Prompt Message for a Reason in a Non-English Language

The message that will be displayed in this prompt can be displayed in several non-English languages.

To Configure the Language of the Prompt Message for a Reason

1. From a machine where the non-English language is installed, connect to the PSMP and open the `basic_psmserver.conf` file with an advanced text editor, such as notepad++, WinSCP built-in editor, gedit, etc.
2. In the **ReasonPromptCaption** parameter, specify the message to display in the non-English language.
3. Save the file in ASCII characters.
4. Make sure that the machine where the message will be displayed is configured to display non-English text.

Configuring Recordings and Audits

The PSMP records privileged sessions and stores them in the Vault where they can be viewed at any time by authorized users. It provides the following recording and audit options:

- **Recordings** – The PSMP can create **text and video recordings**, including any keystroke, of privileged sessions on SSH and Telnet connections. You can access these recordings, view their details and their contents, including the location from where the user connected. You can also see detailed information and properties of the recording file. Recordings can be configured at platform level, overriding the general configuration and enabling you to customize recordings for platforms. To configure SSH text or video recording, refer to *Customizing Recordings*, page 593.
- **Audits** - The PSMP can create **audit records** for activities that are performed during SSH, Telnet and SCP connections for privileged sessions. To configure the audit, refer to *Configuring Detailed Audit*, page 597.

For more information about configuring and accessing recordings and audit records, refer to *Accessing Privileged Session Recordings*, page 322.

Customizing Recordings

Video and text recordings for PSMP-SSH connections are configured at PSM general level (in Web Access Options). These instructions describe how to customize these recordings at platform level, which overrides the general level.

You can customize settings for the following text recorders:

- **SSH text recorder** – The PSMP can record all the keystrokes that are typed during privileged sessions on SSH connections. The recording can be viewed either as a text file or as a video. This type of recording is supported for the following connection component:

- PSMP-SSH

Note: This configuration also affects the SSH text recording in PSM.

To Customize Recordings

1. Click **ADMINISTRATION** to display the **System Configuration** page, then click **Platform Management** to display a list of supported target account platforms.
2. Select the platform to configure, then click **Edit**; the settings page for the selected platform appears.
3. Expand **UI & Workflows**, then right-click **Privileged Session Management**, a pop-up menu displays the parameter sets that you can add and customize to manage your PSM recordings.
4. From the pop-up menu, select **Add Recorder Settings**; a new set of parameters called Recorder Settings is added.

5. Disable or customize text and video recordings for this platform:

Notes:

- Video recording in PSMP is affected by the same parameter as text recording.
 - These settings affect SSH text recordings for SSH connections through PSM as well as PSMP connections.
- i. Right-click **Recorder Settings**, then select **Add SSH Text Recorder**; a new set of parameters called SSH Text Recorder is added.
 - ii. By default, SSH text recordings for SSH connections are enabled. To disable these recordings for this platform, set the value of **Enabled** to **No**.
 - iii. Define the channels that will be recorded during the session. By default, the following channels are recorded for SSH connections:

Property	Default Value	Description
In	Yes	Whether or not the terminal's STDIN stream will be recorded.
Out	Yes	Whether or not the terminal's STDOUT and STDERR streams will be recorded.
Keystrokes	Yes	Whether or not all the keystrokes logged by the user from the start of the line until the user presses Enter will be recorded. Note: Control characters are not recorded.

To disable recordings on any of these channels, set the value of the channel property to **No**.

6. Click **Apply** to save the new parameter values and stay in the platform settings page,
or,

Click **OK** to save them and return to the System Configuration page. The changes will be applied after the period of time specified in the **ConfigurationRefresh Interval** parameter.

Configuring Detailed Audit

By default, the PSMP records all the activities that take place during privileged sessions and provides audits for the following events:

- **SSH keystrokes** – The PSMP can record all the keystrokes that are carried out during privileged SSH sessions. This type of auditing is supported for the following connection component:

- PSMP-SSH

Note: This configuration also affects SSH keystrokes audit in PSM.

To Configure Detailed Audit

1. Click **ADMINISTRATION** to display the **System Configuration** page, then click **Platform Management** to display a list of supported target account platforms.
2. Select the platform to configure, then click **Edit**; the settings page for the selected platform appears.

3. Expand **UI & Workflows**, and then right-click **Privileged Session Management**.
4. From the pop-up menu, select **Add Audit Settings**; a new parameter is added to the Privileged Session Management settings.
5. Select the Audit Settings, then from the pop-up menu, disable or customize SSH Keystrokes Audits for PSMP-SSH connection component using this platform:
 - i. Right-click **Audit Settings**, then from the pop-up menu, select **Add SSH Keystrokes Audit**.
 - ii. By default, SSH keystrokes auditing is enabled for the supported connection component. To disable auditing for this component, in the Properties list, set the value of **Enable** to **No**.
 - iii. Configure advanced properties to determine how the PSMP will manage audit records. For more information about these properties, refer to the Privileged Account Security Reference Guide.
6. Click **Apply** to save the new parameter values and stay in the platform settings page,
or,
Click **OK** to save them and return to the System Configuration page. The changes will be applied after the period of time specified in the **ConfigurationRefreshInterval** parameter.

Hiding Passwords during Recordings

The PSMP identifies passwords that are typed by users during SSH and Telnet sessions by looking for password prompts. By default, the prompts that the PSMP looks for include common prompts for Unix platforms or for Vault passwords. Customize this list to include all password prompts that are received in your environment. When users type a character that cannot be included in a password, such as a space, or when they press Enter, the PSMP resumes the audit and recording. You can update this list of characters too.

This can be configured at platform level, overriding the general configuration.

Note: This configuration affects both PSM and PSMP.

To Hide Passwords during Recordings

1. Click **ADMINISTRATION** to display the **System Configuration** page, then click Platform Management to display a list of supported target account platforms.
2. Select the platform to configure, then click **Edit**; the settings page for the selected platform appears.
3. Expand **UI & Workflows**, then right-click **Privileged Session Management**: a pop-up menu displays the parameter sets that you can add and customize to manage your PSM recordings.
4. From the pop-up menu, select **Internal Capability Settings**; a new set of parameters called Internal Capability Settings is added.
5. Right-click **Internal Capability Settings**, then from the pop-up menu, select **Add SSH Password Hiding**; a new capability parameter is added.

6. Select **SSH Password Hiding**, then specify the following properties:
 - **Enabled** – Determines whether or not passwords will be recorded during PSMP sessions. The default value is **Yes**, indicating that this feature is enabled and passwords will not be recorded.
 - **PasswordPrompts** – This is a regular expression that is used to identify password prompts. When the system finds a match to this regular expression, it omits the password from the PSM session recording.
 - **InvalidPasswordChars** – Defines characters that cannot be included in passwords. When the user specifies one of these characters, the PSM resumes auditing and recording each keystroke. The default values are spaces and tabs.
7. Click **Apply** to save the new parameter values and stay in the Web Access Options page,
or,
Click **OK** to save them and return to the System Configuration page. The changes will be applied after the period of time specified in the **ConfigurationRefreshInterval** parameter.

Configuring SCP Audit Capabilities

By default, the PSMP records SCP commands that are issued in order to copy files securely. This type of auditing is supported for the PSMP-SCP connection component. SCP auditing is automatically configured and enabled at system level and can be overridden at platform level, enabling you to customize detailed audit for platforms.

In order to customize SCP auditing, the following CyberArk component compatibility is required:

- All PSM servers in your environment must be V9.5 or above.
- All PSM SSH Proxy servers in your environment must be V7.2.17 or above.
- The Vault and the PVWA must both be V9.5 or above.

To Configure SCP Audit Capabilities

1. Click **ADMINISTRATION** to display the **System Configuration** page, then click Platform Management to display a list of supported target account platforms.
2. Select the platform to configure, then click **Edit**; the settings page for the selected platform appears.
3. Expand **UI & Workflows**, then right-click **Privileged Session Management**: a pop-up menu displays the parameter sets that you can add and customize to manage your PSM recordings.
4. From the pop-up menu, select **Add Audit Settings**; a new parameter is added to the Privileged Session Management setting.

5. Right-click **Audit Settings**, then from the pop-up menu, select **SCP Audit**.
 - By default, SCP Audit is enabled for the supported connection components. To disable auditing for these components, in the Properties list, set the value of **Enable** to **No**.
 - Configure advanced properties to determine how the PSMP will manage SCP audit records. For more information about these properties, refer to the *Privileged Account Security Reference Guide*.
6. Click **Apply** to save the new parameter values and stay in the platform settings page,
or,
Click **OK** to save them and return to the System Configuration page. The changes will be applied after the period of time specified in the **ConfigurationRefreshInterval** parameter.

Configuring SSH Tunneling for PSMP

The PSMP enables authorized users to initiate and use an SSH tunnel to access a target SSH server, while providing start/end tunnel session audit capabilities. Through this tunnel, users can launch GUI applications such as HTTPs or X from their workstation, maintaining their existing workflow.

Using the PSMP, Security Managers can control access by determining which users can access different target systems. PSMP's flexible configuration also enables them to enable and disable tunneling for specify systems, according to access and security needs.

All access through the PSMP is monitored and stored as a full audit trail in the Vault, where authorized auditors can access it at any time.

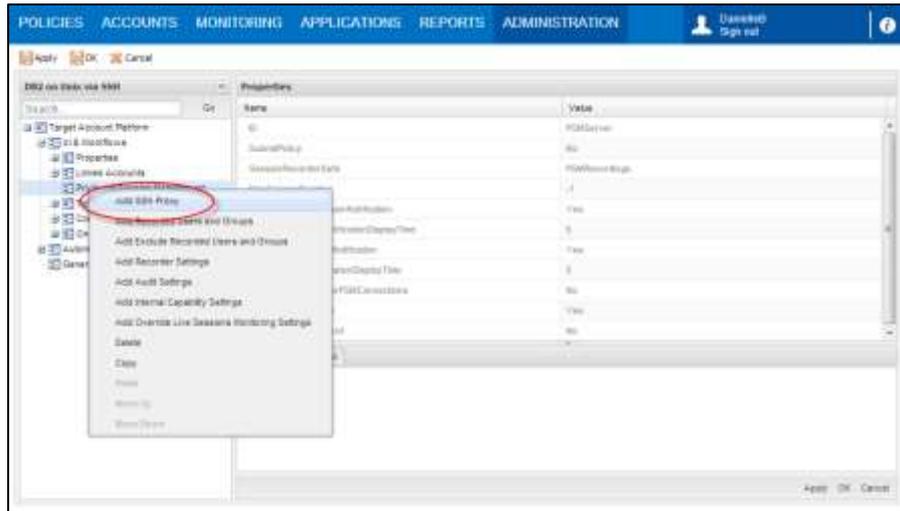
Configuring Platforms to Enable Access through an SSH Tunnel

To enable users to use accounts to access remote machines through an SSH Tunnel, configure the associated platform.

1. Log onto the PVWA as an administrator user.
2. Click **ADMINISTRATION** to display the **System Configuration** page, then click Platform Management to display a list of supported target account platforms.
3. Select the platform to configure for PSMP, then click **Edit**; the settings page for the selected platform appears.
4. Expand **UI & Workflows**, then expand **Privileged Session Management**, and then select **SSH Proxy**; the SSH tunneling parameters are displayed in the Properties list.

Note: If **SSH Proxy** doesn't exist (after an upgrade of PVWA or when using platforms that are not are configured with **SSH Proxy**), add it manually:

- Right-click **Privileged Session Management** then, from the pop-up menu, select **Add SSH Proxy**.



5. Specify the following parameters:
 - **EnableSSHTunneling** – Whether or not SSH tunneling will be enabled for accounts associated with this platform. The default value is **No**. Change it to **Yes**.
 - **TunnelingPorts** – A list of target ports for which the associated account can be used. This parameter is only relevant if the **EnableSSHTunneling** parameter is set to **Yes**. To avoid an error message, make sure that you specify at least one target port.
6. Click **Apply** to save the configuration changes.

To Disable SSH Tunneling

This procedure describes how to disable SSH tunneling after you have specified ports in the **TunnelingPorts** parameter.

1. In the SSH Proxy parameter, set **EnableSSHTunneling** to **No**.
2. Right-click **TunnelingPorts** and, from the pop-up menu, select **Revert to Default**.

Logging

The following parameters define the location of the log file and the amount of information that is stored in this file.

- The **PSMPOpenSSHTraceLevels** parameter specifies the level of debug messages that will be included in the log file.

You can set several values, separated by commas.

Debug level	Indicates
0	No messages will be written to the trace log. This is the default value.
1	PSMP exceptions will be written to the trace log.
2	PSMP trace messages will be written to the trace log.

- The **PSMPOpenSSHLogFolder** parameter specifies the PSMP log folder. The default folder is `/var/opt/CARKpsmp/logs/components`.

To Set Log Values

1. At a command line, specify the following command to open the sshd file:

```
vi /etc/init.d/sshd
```

2. At the beginning of the file, add the following lines:

```
export PSMPOpenSSHTraceLevels=1,2
export PSMPOpenSSHLogFolder=/var/opt/CARKpsmp/logs/components
```

3. Restart the sshd service to apply the changes.

```
/etc/init.d/sshd restart
```

Configuring Tunnel Port Response Timeout

The following parameter determines the time in milliseconds that the PSMP waits for a response time from the tunnel port before displaying a timeout error:

- PSMPOpenSSHTunnelPortTimeout

The default value is 10000 milliseconds.

To Set Log Values

1. At a command line, specify the following command to open the sshd file:

```
vi /etc/init.d/sshd
```

2. At the beginning of the file, add the following lines:

```
export PSMPOpenSSHTunnelPortTimeout=10000
```

3. Restart the sshd service to apply the changes.

```
/etc/init.d/sshd restart
```

Configuring a Subnet Mask

The PSMP enables you to create subnet masks to enable users from multiple locations to initiate remote PSMP sessions without having to provision each IP address or DNS separately.

Platform configurations determine whether or not subnets can be defined for associated accounts. All these accounts can be used from the IP addresses or DNS included in the defined subnets.

To Configure a Platform to Support Subnets

1. Create a new account platform or configure an existing one. For more information about creating platforms, refer to *Managing Target/Service Account Platforms*, page 108.
2. Click **ADMINISTRATION** to display the **System Configuration** page, then click Platform Management to display a list of supported target account platforms.
3. Select the platform to configure for subnet support, then click **Edit**; the settings page for the selected platform appears.

4. Expand **UI & Workflows**, and then select **Privileged Session Management**; the PSM parameters are displayed with their default values.
5. Set **SubnetPolicy** to **Yes**.
6. Click **Save** to save the new configurations.

To Define a Subnet

You can define a subnet for an existing account or create a new one.

1. In the **Add Account** page or the **Edit Account** page, select the platform name that supports subnets.
2. In **Address**, specify the Network ID that combines with the subnet mask to define the network subnet.
3. In **Subnet mask**:
 - IPv4 – Specify the subnet mask that combines with the Network ID to define the network subnet.

The screenshot shows the 'Add Account' page in a web application. The navigation bar at the top includes 'POLICIES', 'ACCOUNTS', 'MONITORING', 'APPLICATIONS', 'REPORTS', and 'ADMINISTRATION'. The user is logged in as 'Clarewell' and can 'Sign out'. The form fields are as follows:

- Store in Safe:** Passwords (dropdown)
- Device Type:** Operating System (dropdown)
- Platform Name:** Linux Shell Subnet (dropdown)
- Address:** 1.1.1.0 (text input, circled in red)
- Subnet Mask:** 255.255.255.0 (text input, circled in red)
- User Name:** (text input)
- Password Content:** (text input)
- Name:** Radio buttons for 'Auto-generated (Name pattern: {device}/{deviceID}-{address}/username-{y-y34}', '{service}/testName (system/cluster-client)', and 'Custom'.
- Disable automatic management for the account:** (checkbox) Reason: (text input)

- IPv6 – Specify the prefix length that combines with the Network ID to define the network subnet.

The screenshot shows the 'Add Account' form with the following fields and options:

- Store in Defc: Passwords
- Device Type: Operating System
- Platform Name: Link Subnet via SSH
- Address: 2001:2000:1000:1000:1000:2000:2000:1000
- Subnet Mask: 128
- Optional Fields:
 - Username: _____
- Password Control:
 - Password: _____
 - Confirm Password: _____
 - Name:
 - Auto-generated (Name pattern: DeviceType-PolicyID-Address-Gateway-IP-OSID)
 - Custom: _____
- Disable automatic management for this account:
 - Name: _____

- Specify the rest of the account properties, then click **OK**; the account is saved and the Account Details page displays the details of the new/updated account.

Viewing Accounts according to Subnet

Authorized users can configure the Accounts List to display the Subnet mask so that you can sort accounts according to the subnet.

- Click **ADMINISTRATION** to display the **System Configuration** page, then click **Options**; the Web Access Options are displayed.
- Expand the **Accounts UI Preferences** parameters, then expand the **Main** parameters, and then expand the **Accounts** parameters.
- Expand **Displayed Columns**; a list of the columns that are displayed in the Accounts page appears.
- Select **SubnetMask** then, in the Properties list, set **Visible** to **Yes**.

The screenshot shows the 'Options' page with the following structure:

- Options
 - Mobile Application
 - Setup Configurations
 - Accounts UI Preferences
 - Main
 - Accounts
 - Displayed Columns
 - Username
 - Address
 - Safe
 - PolicyID
 - Folder
 - Name
 - AuthenticatedBy
 - LastUsedBy
 - LockedBy
 - SubnetMask (Selected)
 - Old actions
 - Usages
 - Users Collections
 - Web Help
 - Application Assemblies
 - Audit Filters

The Properties list for SubnetMask is as follows:

Name	Value
Name	SubnetMask
DisplayName	Subnet Mask
Web	No
Data Type	String
Visible	Yes

- Click **OK** to save the new account UI settings.

The next time you display the Accounts list, an additional column displays the Subnet mask.

Configuring UNIX Domain/NIS Accounts

You can use the PSMP to access target machines using UNIX Domain/NIS accounts. This platform is not predefined and must be configured manually.

Note: In SSH protocol, there is no foolproof way to ensure the identity of the target machine, which could potentially lead to a security risk. Please take this into consideration when using this feature.

To Configure a UNIX Domain/NIS Platform

1. Click **ADMINISTRATION** to display the **System Configuration** page, then click Platform Management to display a list of supported target account platforms.
2. Select an existing SSH platform that is similar to the new target account platform. For example, Unix via SSH.
3. Click **Duplicate**; the Duplicate Platform window appears.
4. Type the name and a description of the new platform, then click **Save & Close** to create the new platform.
5. Select the new target account platform, and then click **Edit**; the configuration page for the selected platform appears.
6. Expand **UI & Workflows**, and then expand **Connection Components**; the Connection Components parameters are displayed with their default values.
7. Set the Override User Parameters for **PSMP-SSH** and **PSMP-SCP** connection components:
 - i. Right-click the connection component and select **Add Override User Parameters**; a new set of parameters is added.
 - ii. Right-click **Add Override User Parameters** and select **Add Parameter**; a new parameter is added.
 - iii. Select the new parameter then, in the Properties list, set the name of this parameter to **PSMRemoteMachine**.
8. Change any additional parameter values and/or add new values to define the new platform.
9. Click **Apply** to save the new configurations and apply them immediately,
or,
Click **OK** to save the new configuration and return to the System Configuration page.
10. Restart the PSMP service to apply the configuration changes:
At a command line, run the following commands:

```
/etc/rc.d/init.d/psmpsrv stop  
/etc/rc.d/init.d/psmpsrv start
```

To Add a UNIX Domain/NIS Account

1. In the Add Account page, add the domain account that will be used to access the target account. Specify the following account properties:
 - **Platform Name** – Select the platform that you created for UNIX domain/NIS accounts in the previous section.
 - **Address** – Specify the IP address or DNS of the domain server in the domain where the target machine resides.
2. Specify additional required and optional account properties.

For more information about adding accounts, refer to *Adding Accounts*, page 122.

For specific information about the information required to add Windows Domain accounts, refer to *Windows Domain Accounts*, page 141.

For specific information about the information required to add Unix Domain/NIS accounts, refer to *Unix Domain/NIS Accounts*, page 140.

Configuring PSMP Syntax Delimiters

The PSMP command includes two different delimiters, one for separating the required parameters (which is '@' by default) and another one for separating the optional parameters (which is '#' by default).

You can change these default delimiters, which allows the following usages:

- Connections with domain user names that include @ as part of their name. To allow this, configure the PSMP to use a delimiter other than @ to separate the required parameters.
Note: You cannot use PSMP prompts to connect through PSMP with a domain user name, and you must specify all the parameters in the command line.
- Use of optional parameters when copying files securely through PSMP with SCP syntax (since in SCP syntax, '#' (hash) cannot be used as a delimiter). To allow this, configure the PSMP to use a delimiter other than # to separate the optional parameters.

Note: When using a delimiter other than '@' to separate the required parameters, the delimiter before the 'proxyaddress' parameter must be '@' and must not be replaced.

Changing the PSMP Delimiter that Separates Required Parameters

You can add additional delimiters to separate required parameters in the PSMP command by adding the PSMP_AdditionalDelimiter parameter to the sshd_config configuration file. When the PSMP detects the specified delimiter in the PSMP command, it is used to parse the command, otherwise the default delimiter (@) is used.

1. In the `/etc/ssh` directory, open the `sshd_config` configuration file for editing.
2. Add the following parameter to the file:

```
PSMP_AdditionalDelimiter <delimiter>
```

The following example shows how to enable users to specify % as a delimiter character in addition to @.

```
PSMP_AdditionalDelimiter %
```

3. Save the changes and close the **sshd_config** configuration file.
4. Restart the sshd service for these changes to take affect:

```
/etc/init.d/sshd restart
```

Note: You cannot use the same delimiter for both required and optional parameters.

Changing the PSMP Delimiter that Separates Optional Parameters

You can replace the delimiter that separates optional parameters in the PSMP command by adding the `PSMP_TargetAddressPortAdditionalDelimiter` parameter to the `sshd_config` configuration file. When the PSMP detects the specified delimiter in the PSMP command, it is used instead of the default delimiter (#).

1. In the `/etc/ssh` directory, open the **sshd_config** configuration file for editing.
2. Add the following parameter to the file:

```
PSMP_TargetAddressPortAdditionalDelimiter <delimiter>
```

The following example shows how to enable users to specify % as a delimiter character instead of #.

```
PSMP_AdditionalDelimiter %
```

3. Save the changes and close the **sshd_config** configuration file.
4. Restart the sshd service for these changes to take affect:

```
/etc/init.d/sshd restart
```

Note: You cannot use the same delimiter for both required and optional parameters.

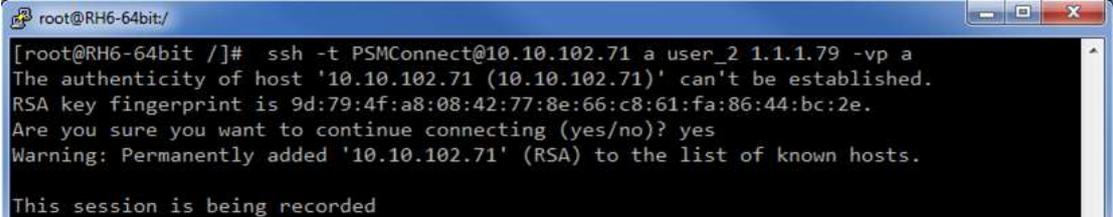
Configuring the Verification of a Server's Host Key

One of the security controls in Unix environments is the ability to block access to unknown machines until their host keys are examined and verified by an administrator or an end-user. When the PSMP encounters a target machine's host key which is not cached or does not match the one cached, it either completes the connection to the target machine or abandons it, based on predefined settings. You can configure this response by adding the **ServerHostKeyNotInCacheBehavior** parameter in the `basic_psmserver.conf` file. This parameter has three settings, which are described below.

- **ServerHostKeyNotInCacheBehavior=1** – The connection will be abandoned. The system administrator can examine the host key that was received from the target machine and, if it is trusted, they can add the new host key to the "known_host" file in the PSMP cache. The administrator can do this in advance for all the target systems when configuring the PSMP environment. This is the most secure option as only a system administrator can store a target machine's host key in PSMP cache.
- **ServerHostKeyNotInCacheBehavior=2** – The connection will be established normally and the new key will be added or updated in the cache automatically.
- **ServerHostKeyNotInCacheBehavior=3** – The connection will be established normally and the new key will not be added nor updated in the cache. If this parameter was not added or an invalid value is set, the PSMP will use this as a default value.

Displaying Notifications when Sessions are Recorded

A notification can be displayed when a remote session is opened and the PSMP starts recording it, and ensures that users know that their session is being recorded.



```
root@RH6-64bit/
[ root@RH6-64bit /]# ssh -t PSMConnect@10.10.102.71 a user_2 1.1.1.79 -vp a
The authenticity of host '10.10.102.71 (10.10.102.71)' can't be established.
RSA key fingerprint is 9d:79:4f:a8:08:42:77:8e:66:c8:61:fa:86:44:bc:2e.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '10.10.102.71' (RSA) to the list of known hosts.
This session is being recorded
```

This notification can be configured with the following parameters in the `basic_psmserver.conf` file:

- **PSMPRecordingNotificationMessage** – This parameter specifies the message to display. The default message is "The session is being recorded"
- **PSMPRecordingNotificationTimeout** – This parameter determines the number of seconds that the recorded session notification is displayed. The default value is 5 seconds. If 0 (zero) is specified, the notification will not be closed automatically and will be displayed until the user presses Enter.

Displaying Notifications in a Non-English Language

Recorded session notifications can be displayed in several non-English languages.

To Configure the Notification Language

1. From a machine where the non-English language is installed, connect to the PSMP and open the `basic_psmptserver.conf` file with an advanced text editor, such as notepad++, WinSCP built-in editor, gedit, etc.
2. In the **PSMPRecordingNotificationMessage** parameter, specify the message to display in the non-English language.
3. Save the file in ASCII characters.
4. Make sure that the machine where the message will be displayed is configured to display non-English text.

Administering the PSMP

Managing the PSMP Service

The PSMP is installed as an automatic system service called `psmptsrv`. You can manage this service using the following command:

```
/etc/init.d/psmptsrv {start|stop|restart|status} [{psmp|psmptadb}]
```

The PSMP service enables you to manage PSMP and AD Bridge servers, either separately or together, using one of the following commands:

- To manage only the PSMP server, run the following command:

```
/etc/init.d/psmptsrv {start|stop|restart|status} psmp
```

- To manage only the PSMP AD Bridge server, run the following command:

```
/etc/init.d/psmptsrv {start|stop|restart|status} psmptadb
```

- To manage both the PSMP and the PSMP AD Bridge server together, do not specify a server in the command, as shown below:

```
/etc/init.d/psmptsrv {start|stop|restart|status}
```

Adminstrating the SSH Proxy Machine

Administrative users can connect to the PSM SSH Proxy machine to perform management tasks on the machine itself without being forwarded to target machine, using the following command:

```
<ssh client> <administrative user>@<proxyaddress>
```

The PSMP identifies the following administrative users:

- Built-in Unix user:
 - root
- Administrative users that you must create manually in the PSM SSH Proxy machine:
 - proxymng
 - proxymng<number>

You can configure more administrative user names by adding the **PSMP_MaintenanceUsers** parameter to the **sshd_config** configuration file.

Note: Only local users can connect as maintenance users.

1. In the **/etc/ssh** directory, open the **sshd_config** configuration file for editing.
2. Add the following parameter to the file:

```
PSMP_MaintenanceUsers <username>,<username>
```

Note: You can either set specific usernames or use '*' at the beginning and/or the end of the username string.

This example will allow the following administrative users: user1, all users that end with "user2", all users that starts with "user3" and all users that include "user4".

```
PSMP_MaintenanceUsers <user1>,<*user2>,<user3*>,<*user4*>
```

3. Save the changes and close the **sshd_config** configuration file.
4. Restart the sshd service for these changes to take affect:

```
/etc/init.d/sshd restart
```

Defining Platforms for Assorted Scenarios

To Configure Platforms to support SSO (user + password in the Vault)

1. Log onto the Password Vault Web Access as a user with permissions to configure platforms.
2. Click **ADMINISTRATION** to display the **System Configuration** page, then click Platform Management to display a list of supported target account platforms.
3. Select **Unix via SSH**, then click **Duplicate**; the Duplicate Platform window appears.
4. Type any name you choose for the new platform and specify a description, then click **Save & Close**; the new platform appears in the list of target account platforms.
5. Select the new target account platform, and then click **Edit**; the platform settings page appears.
6. Expand **UI & Workflows**, and then select **Privileged Session Management** and do the following:
 - Make sure that the **PSMServer** parameter is enabled.
 - Set the **EnablePrivilegedSSO** parameter to **Yes**.
7. In the **Connection Components** section, make sure that the **PSMP-SSH** Connection Component ID is defined and enabled.
8. Click **OK** to save the updates and return to the System Configuration page.
9. When assigning an account to this platform, provide the username, password and address.

To Configure Platforms to support Secure Connect (user in the Vault, but with no password)

1. Log onto the Password Vault Web Access as a user with permissions to configure platforms.
2. Click **ADMINISTRATION** to display the **System Configuration** page, then click Platform Management to display a list of supported target account platforms.
3. Select **PSMSecureConnect**, then click **Duplicate**; the Duplicate Platform window appears.
4. Type any name you choose for the new platform and specify a description, then click **Save & Close**; the new platform appears in the list of target account platforms.
5. Select the new target account platform, and then click **Edit**; the platform settings page appears.
6. Expand **UI & Workflows**, and then select **Privileged Session Management** and do the following:
 - Make sure that the **PSMServer** parameter is enabled.
 - Set the **EnablePrivilegedSSO** parameter to **No**.

7. In the **Connection Components** section, make sure that the **PSMP-SSH** Connection Component ID is defined and enabled.
8. Under **Properties** → **Required**, add the **Username** and **Address** properties.
9. Click **OK** to save the updates and return to the System Configuration page.
10. When assigning an account to this platform, provide the username and address.

To Configure Platforms to support Secure Connect (user not in the Vault)

1. Log onto the Password Vault Web Access as a user with permissions to configure platforms.
2. Click **ADMINISTRATION** to display the **System Configuration** page, then click Platform Management to display a list of supported target account platforms.
3. Duplicate the **Unix SSH Subnet** platform and name the new platform with any name you choose.
 - Make sure that the **PSMServer** parameter is enabled.
 - Set the **EnablePrivilegedSSO** parameter to **No**.
4. In the **Connection Components** section, make sure that the **PSMP-SSH** Connection Component ID is defined and enabled.
5. Under **Properties** → **Required** add the **address** property.
6. Click **OK** to save the updates and return to the System Configuration page.
7. When assigning an account to this platform, provide the address.

To Configure Platforms to support Secure Connect with Subnet (user not in Vault)

1. Log onto the Password Vault Web Access as a user with permissions to configure platforms.
2. Click **ADMINISTRATION** to display the **System Configuration** page, then click Platform Management to display a list of supported target account platforms.
3. Duplicate the **Unix SSH Subnet** platform and name the new platform with any name you choose.
 - Make sure that the **PSMServer** parameter is enabled.
 - Set the **EnablePrivilegedSSO** parameter to **No**.
4. In the **Connection Components** section, make sure that the **PSMP-SSH** Connection Component ID is defined and enabled.
5. Click **OK** to save the updates and return to the System Configuration page.
6. When assigning an account to this platform, provide the address and mask properties.

Integrating with AD Bridge Capabilities

CyberArk Privileged Account Security solution integrates with Microsoft's Active Directory (AD) to provision users transparently on remote UNIX systems, streamlining user management and reducing administrative overhead. In addition to automatic user provisioning, this CyberArk solution benefits from all standard CyberArk security and management features, including access control and auditing.

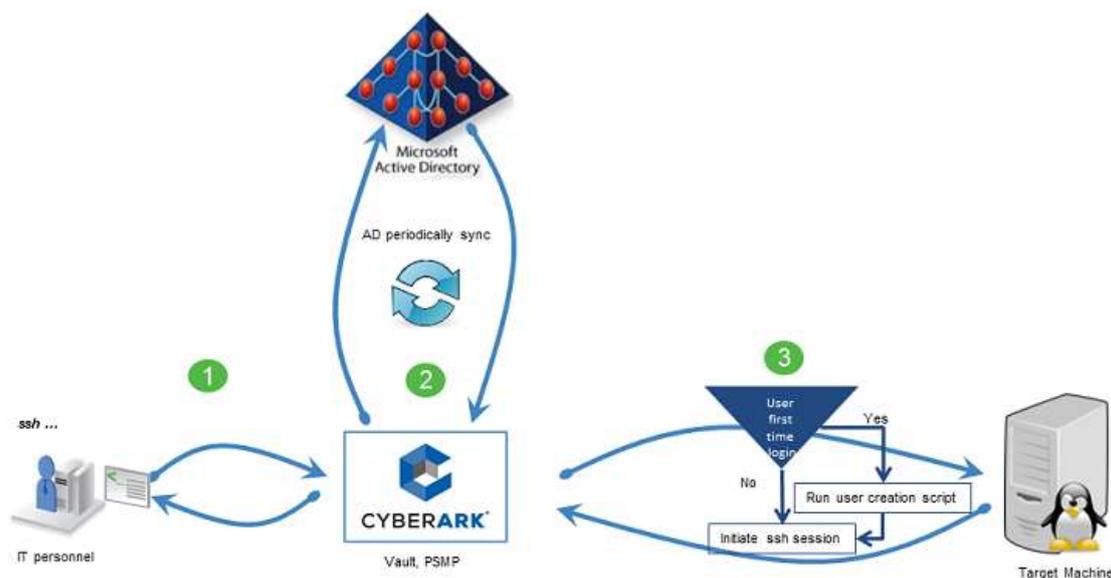
The solution allows users who authenticate with passwords to log onto a UNIX machine using their AD credentials as their user is automatically synchronized with a corresponding user in the Vault. Likewise, existing groups in AD directories are automatically synchronized with a corresponding group in the Vault. Users have immediate access to UNIX machines, based on their AD permissions and groups, facilitating an uninterrupted workflow and maintaining productivity.

Specifically, this solution provides the following functionality:

- **User authentication through the Vault** – Users and groups listed in an Active Directory can connect to a target UNIX machine as a local user through the Vault, which authenticates them. The Vault supports multiple authentication methods, such as LDAP, password, RADIUS, PKI, and more. For more information about authenticating to the Vault, refer to the Privileged Account Security Installation Guide.
- **First time login** – The first time that users try to log onto the target UNIX machine, this solution automatically provisions them, and enables them to connect without any manual intervention.
- **Provisioning users** – Users are automatically given permissions on the target machine based on their permissions in the Active Directory. By default, they are created on the target machine with predefined default settings. The same provisioned user on all Unix/Linux systems is allocated the same UID. Each user can be assigned a single shell configuration file to customize the user when it is created, such as user profile attributes, environment variables, etc.
- **Provisioning groups** – Users are automatically given permissions on the target machine based on the group they belong to in the Active Directory. Users will be assigned to local groups on the target machine with a name that corresponds to their groups in the Active Directory.
- **Recording and monitoring** – All the activities in each session can be recorded in text format, and stored in the Vault, compressed, for future auditing. These recordings are transparent to users and cannot be bypassed. Auditors can see all the recordings archives, and can retrieve and view comprehensive recordings of these sessions. Search features enable auditors to locate specific recordings. For more information, refer to the *Privileged Session Manager SSH Proxy* chapter in the Privileged Account Security Implementation Guide.
- **Auditing and Reports** – The Vault provides comprehensive auditing for every user provisioning and session recording. Auditing information is displayed in a simple intuitive interface that includes the user's name, the address of the target machine, the duration of the session, and more.
- **Automatic user deprovisioning** – At regular intervals, the PSMP compares provisioned users with their authorizations for the target machine account. If a user's authorizations have been removed and the user is no longer authorized to access the target machine, the PSMP will automatically deprovision the user and they will no longer be able to access the target machine.

Architecture

The following diagram shows how the CyberArk Digital Vault retrieves information about users and groups from the Microsoft Active Directory and creates corresponding users and groups in the target UNIX machine.



The end user, called Mike in the above example, issues an SSH command that will enable him to log on to a target UNIX machine (step 1). The CyberArk Vault intercepts Mike's request and refers to the Microsoft Active Directory for his AD credentials (step 2). When the Vault receives this information, it passes it on to the PSMP which accesses the target machine and checks whether or not Mike has an account there. If he does, the PSMP initiates an `ssh session` and logs him onto the target machine. If Mike does not yet have an account on the target machine, the PSMP creates one for him, and then initiates an `ssh session` and logs him onto the target machine (step 3).

The logon process is audited by the Vault. In addition, subsequent tasks that the user performs on the target machine are recorded by the PSMP and can be viewed in real time or later by auditors.

Configuring the PSMP to Integrate with AD Bridge Capabilities

Configure the Vault to integrate with your enterprise Active Directory. For more information, refer to *Configuring User Management via LDAP* in the Privileged Account Security Installation Guide.

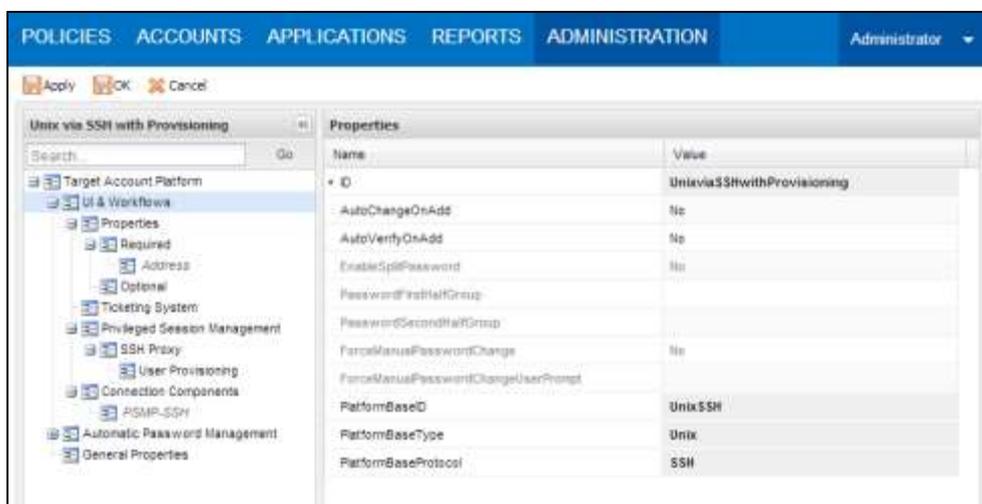
Before configuring the PSMP, make sure that the users you will use to configure the PSMP have the relevant permissions in the Safes where the accounts required to access the target machine and the provisioning account are stored, as described below.

- Administrator user:
 - On the **target machine** account:
 - Add account
 - Update password properties
 - On the **provisioning** account:
 - List accountsand either:
 - Retrieve accounts – users can retrieve accounts and view passwords
 - or
 - Use accounts – users can use accounts but not view passwords
- PSMP-ADBridge application user:
 - On the **provisioning** account:
 - List accounts
 - Retrieve accounts
- End user
 - On the **target machine** account:
 - List accounts
 - Use accounts

Users can be allocated permissions for entire Safes or for specific accounts. For more information about both options and for details about configuring them, refer to *Adding and Managing Safe Members*, page 68, or *Object Level Access Control*, page 76.

Create a Platform for User Provisioning

1. Click **ADMINISTRATION** to display the **System Configuration** page, then click Platform Management to display a list of supported target account platforms.
2. Select **Unix via SSH**, then click **Duplicate**; the Duplicate Platform window appears.
3. Change the name of the duplicated platform to **Unix via SSH with Provisioning**, then click **Save & Close** to create the new platform.
4. Select the new target account platform, and then click **Edit**; the configuration page for the Unix via SSH with Provisioning platform appears.
5. In the **UI & Workflow** parameters, do the following:
 - i. Delete the following properties:
 - **Username** – Expand **Properties**, then select **Required**, and delete the **Username** property.
 - **Linked Accounts** – Expand **Linked Accounts**, and delete the configured linked accounts.
 - ii. Select **Privileged Session Management**, then do the following:
 - a. Set the **EnablePrivilegedSSO** parameter to **No**.
 - b. Make sure that the **UsePersonalPassword** parameter is set to **No**.
 - c. Expand the **Privileged Session Management** properties, then right-click **SSH Proxy**; a drop-down menu appears.
 - d. Select **Add User Provisioning**; a new set of parameters is added for User Provisioning.
 - e. Expand the User Provisioning parameters, then set the value of **Enable User Provisioning** to **Yes**.



6. Click **Apply** to save the new platform configurations and return to the Platform Management page.
7. Select the **Unix via SSH with Provisioning** platform and change its status to Active.

Add the Target Machine Account

1. In the Accounts page, click **Add Account**; the Add Account page appears.
2. From the Safe drop-down list, select the Safe where the account will be stored.
3. From the Device drop-down list, select the UNIX platform where this account will be used.
4. From the Platform Name drop-down list, select **Unix via SSH with Provisioning**; the properties for the type of account that you have selected appear automatically, according to the definitions in the target platform configurations.
5. Specify the required account properties and, if necessary, the optional account properties.

Note: To connect to remote machines on IPv6, specify the IPv6 address using the global format, as shown in the following example:
1000:1000:1000:1000:1000:1000:1000:0055

6. Click **Save**; the new account is added and the Account Details page appears.

Associate the Target Machine Account with the User Provisioning Account

An additional privileged user account is required to create ad-hoc users on target machines. This account is associated with the target machine account, and is used to create users transparently.

1. In the Account Details page of the target machine account, display the User Provisioning tab.
2. In the User Provisioning tab, click **Associate**; the Associate Account window appears. This window lists the frequently used accounts. If the account that you require does not appear in this list, do a search for the required account.
3. Select the required account, then click **Associate**; the selected account is linked to the target machine account and its details are listed in the User Provisioning tab.

Note: The user provisioning account can use password or SSH Key to authenticate to target systems.

Creating Logon Accounts for the User Provisioning Account

A logon account can be used to enable the Provisioning account to log onto machines that do not permit direct logon and create ad-hoc users. When a logon account is associated with a Provisioning account, it is used to log onto the remote machine and then elevates itself to the role of the Provisioning user.

Note: A logon account can be either a regular account or an SSH Key account. However, if a logon account is used, the Provisioning account must be a password account.

For more information, refer to *Using Logon Accounts for PSM-SSH and PSM-Telnet Connection Components*, page 647.

Synchronize Groups with Vault Groups

During user provisioning, Vault users are added to all groups on the target machine that have corresponding groups in the Vault to which they belong. Corresponding groups in the Vault must have the same name as the group on the target machine, with **ADB_** as a prefix. Groups that do not exist on the target machine will not be created.

For example, a user called David belongs to **ADB_Group1** in the Vault. On the target machine there is a group called “Group1” to which David will be added when his user is provisioned. If “Group1” does not exist on the target machine, David’s user will be created but he will not be allocated to a corresponding group.

- In the Vault, create user groups that correspond to the user groups in the Active Directory, with an **ADB_** prefix. For more information about creating user groups in the Vault, refer to *Managing Groups*, page 56.

Enable SFTP on the Target Machine

SFTP must be enabled on the remote target UNIX machine.

1. On the remote target UNIX machine, open the main sshd configuration file, **/etc/ssh/sshd_config**.
2. Enable SFTP.

Set a Default Shell

Whenever a user logs onto the target machine for the first time, their user is provisioned automatically with a predefined default shell.

1. On the remote target UNIX machine, open the **/etc/default/useradd** file.
2. Define the default shell that determines how new users will be provisioned.

Managing Users' UID

You can manage users' UIDs for all users who use AD Bridge connections to log onto remote UNIX machines.

To Enable and Disable Users' UID Management

You can enable or disable UID management for AD Bridge connections at system level.

1. Click **ADMINISTRATION** to display the **System Configuration** page, then click **Options**; the Web Access Options are displayed.
2. Select **PSMP-ADBridge** then, in the Properties list, set the following property:
 - **ADBridgeManageUID** – This parameter determines whether or not the UID of provisioned users can be managed at system level.
 - Specify **Yes** to enable the Privileged Account Security solution to manage UIDs for provisioned users. This is the default value.
 - Specify **No** to indicate that the target machine OS will assign UIDs to provisioned users.
3. Click **Apply** to apply the new configurations immediately,
or,
Click **OK** to save the new configurations and return to the System Configuration page.

To Set a Minimum UID for AD Bridge Provisioned Users

You can define a minimum UID number for provisioned users, so that all provisioned users will be allocated a higher UID number than existing UIDs.

1. Click **ADMINISTRATION** to display the **System Configuration** page, then click **Options**; the Web Access Options are displayed.
2. Select **PSMP-ADBridge** then, in the Properties list, set the following property:
 - **ADBridgeMinimumUID** – This parameter determines the minimum number that can be used as a UID of a PSMP-ADbridge provisioned user. Specify a number higher than 1000. The default value is **15000**.
3. Click **Apply** to apply the new configurations immediately,
or,
Click **OK** to save the new configurations and return to the System Configuration page.

Managing AD Bridge Scripts

The AD Bridge provisioning and deprovisioning scripts enable you to customize provisioning and deprovisioning processes according to your enterprise standards. You can customize scripts for existing platforms and you can also create new scripts for platforms that the Privileged Account Security solution does not store out of the box.

During installation, the following Safes are created to store the provisioning and deprovisioning scripts:

- **PSMP-ADBridgeConf** – All provisioning scripts that are supported out-of-the-box are stored in the **Scripts** folder of this Safe, and are downloaded to the AD Bridge machine each time the PSMP service is restarted.

To retrieve these provisioning scripts during AD Bridge provisioning and deprovisioning, users require the following permissions in this Safe:

- List Files
- Retrieve Files
- **PSMPADBridgeCustom** – A template provisioning script that can be used to customize provisioning scripts is stored in this Safe. All customized scripts for specific device types must be stored in this Safe.

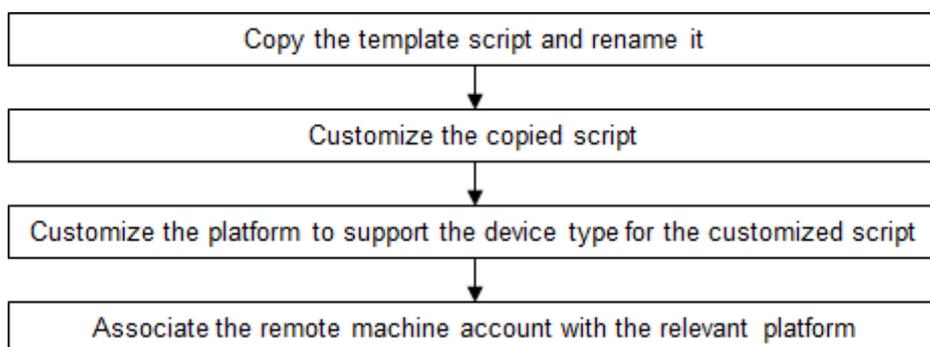
To edit the provisioning and deprovisioning scripts, users require the following permissions in this Safe:

- List Files
- Retrieve Files
- Update Files

By default, Vault administrators have these permissions and can edit these scripts.

By default, users in the PSMP_ADB_AppUsers group have these permissions for both Safes and can use all the provisioning scripts.

The following diagram shows the process of customizing provisioning and deprovisioning steps:



To Customize Provisioning and Deprovisioning Scripts

In the PrivateArk Client:

1. Copy a built-in script to use as a template:
 - i. Log onto the PrivateArk Client with an administrative user.
 - ii. Open the **PSMPADBridgeCustom** Safe and, in the Root folder, right-click the Prov-Sample script.
 - iii. From the pop-up menu, select **Retrieve and Save As**, then save the script in the same Safe with a unique name that describes its purpose.
2. Customize the copied script:
 - i. Open the script and customize it, using the script functions listed in *The Custom Provisioning Script*, page 733.
 - ii. Save the customized script and return it to the Safe.

In the PVWA:

3. Customize the AD Bridge platform to support the device type for the customized script:
 - i. Log onto the PVWA as an administrator user.
 - ii. Click **ADMINISTRATION** to display the **System Configuration** page, then click **Platform Management** to display a list of supported target account platforms.
 - iii. Select the platform that you use to connect to the remote machine, eg, Unix via SSH – ADBridge, then click **Edit**; the platform settings page for this platform appears.
 - iv. Expand **UI & Workflows**, then expand **Privileged Session Management**, and then **SSH Proxy**.
 - v. Expand **User Provisioning**, then right click **Device types** and, from the drop-down menu, select **Add Device type**; a new device type is added to the list of existing supported device types.
 - vi. In the Properties list of the new device, specify the following:
 - **Display name** – The unique name of the device type. This property is mandatory.
 - **Device identifier** – The name of the device that is returned from the 'uname' UNIX command. This property is mandatory.
 - **Provisioning script** – The name of the provisioning script file for the target device type. This property is mandatory.
 - **Additional files** – The names of additional files needed for provisioning. Separate multiple file names with a comma. The property is optional.
 - vii. Click **Apply** to apply the new configurations immediately,
or,
Click **OK** to save the new configurations and return to the System Configuration page.
4. Associate the target machine account with the customized ADBridge platform. You can either add a new account or associate an existing account.
 - i. Display the Account Details page for the account to link to a reconciliation account.
 - ii. On the toolbar, click **Edit**; the Edit Account page appears.
 - iii. In Device Type, specify the type of device on which this account will be used.

- iv. In Platform Name, specify the name of the platform that you customized in the previous step.
 - v. Click **Save**; the PVWA associates the account with the customized platform.
5. Restart the AD Bridge service, using the following command:

```
/etc/init.d/psmpsrv restart psmpadb
```

You can now use this account to log onto the target machine, and the associated platform will automatically use the customized provisioning and/or deprovisioning scripts.

The Custom Provisioning Script

Function	Description	Input
retrieveUID	Retrieves the uid of the given username and returns 0.	\$1 - username
retrieveUsername	Retrieves the name of the user who has the specified uid and returns 0.	\$1 - uid
addUser	Creates a new user on the target machine with the specified username and uid, and returns 0.	\$1 - username \$2 - uid (not mandatory)
removeUser	Deletes the user from the target machine and returns 0.	\$1 - username
resetUserPassword	Resets the user's password to the password specified in the data file and returns 0.	\$1 - datafile path (format "username:password")
changeUID	Resets the uid of the user to the specified uid and returns 0.	\$1 - username \$2 - uid
retrieveHomeFolder	Retrieves the home folder of the specified user and returns 0.	\$1 - username
retrieveUserGroups	Retrieves a list of groups of which the specified user is a member, separated by a comma, and returns 0.	\$1 - username
addUserToGroup	Adds the specified user to the specified group and returns 0.	\$1 - username \$2 - group name
removeUserFromGroup	Removes the specified user from the specified group and returns 0.	\$1 - username \$2 - group name
copyFile	Copies the specified source file to the specified destination file path and returns 0.	\$1 - source file \$2 - destination file path
removeFile	Removes the specified file or folder and returns 0.	\$1 - file or folder path
changeFileOwner	Changes the owner on the given file or folder (recursively) to the specified user and returns 0.	\$1 - username \$2 - file or folder path

addLineToFile	Appends the specified line at the end of the specified file and returns 0.	\$1 - file path \$2 - line to add
removeLineFromFile	Removes the specified line from the specified file and returns 0.	\$1 - file path \$2 - line to remove

Customizing User Profiles

You can customize user profiles for users who are automatically provisioned by the PSMP-AD Bridge. This enables you to assign a single shell configuration file to customize the user when it is created, such as user profile attributes, environment variables, etc.

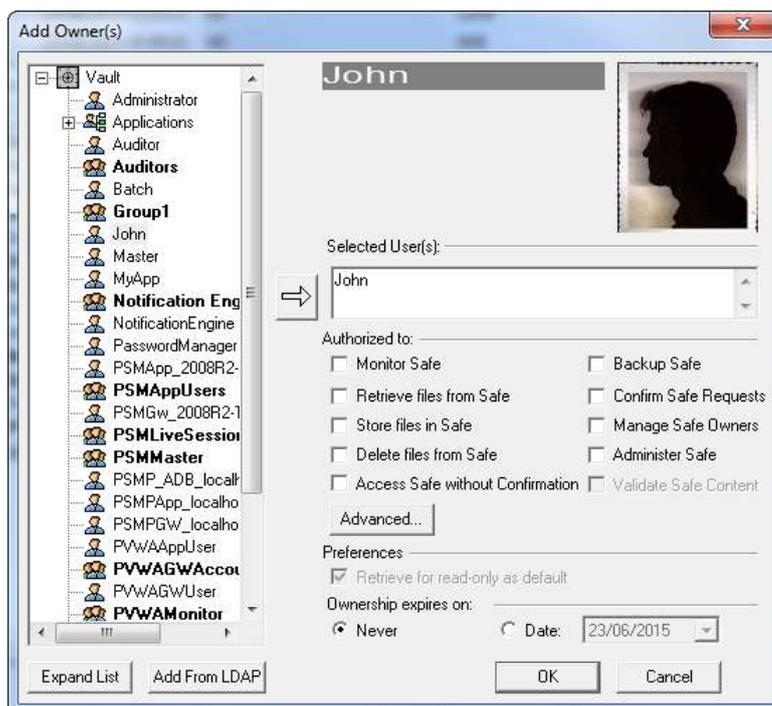
You create your own profile configuration files and store them in the **PSMPADBUserProfile** Safe, which is configured for Object Level Access Control. When the PSMP-ADBridge provisions users, it checks this Safe for a profile configuration file that is relevant to the user being provisioned and, if it finds one, it provisions the user with the customizations defined in the file.

To customize user profile scripts, users require the following permissions in the **PSMPADBUserProfile** Safe:

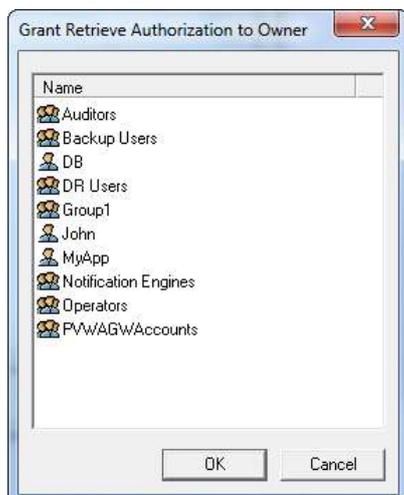
- List Files
- Retrieve Files
- Create Files
- Update Files
- Rename Files
- Delete Files
- View Owners
- Manage Safe Owners

To Customize User Profile Scripts

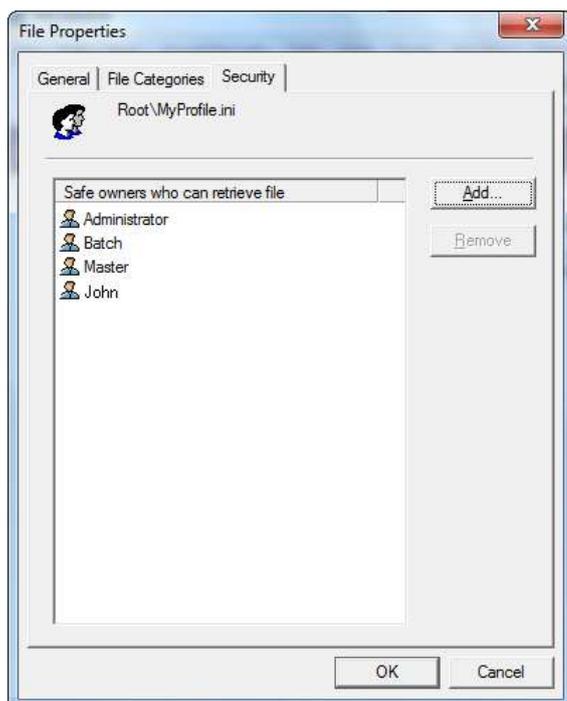
1. Log onto the PrivateArk Client with an administrative user.
2. Open the **PSMPADBUserProfile** Safe, and upload the profile configuration file to the Safe.
3. Add the user who will be provisioned on a remote machine using this customized profile script as a Safe Owner, without granting them any authorizations.



4. Right-click the profile script that you previously uploaded to the Safe and, from the drop-down menu, select **Properties**; the file's Properties window appears.
5. In the Security tab, click **Add**; a list of Safe Owners who can be granted retrieve authorization for the profile script appears.



6. Select the user who will be provisioned on a remote machine using this customized profile script, then click **OK**; the selected user is added to the list of Safe owners who can retrieve the profile script in the Security tab.



7. Click **OK** to close the File Properties window.

Monitoring PSMP Integration with AD Bridge Capabilities

As soon as users try to access target Unix machines, their activities in the Vault are monitored. The entire process of provisioning in the Vault and in the target machine, as well as their sessions on target machines are monitored using the PSMP's regular monitoring features. For more information, refer to *Monitoring Privileged Sessions*, page 690.

Auditing

The Vault provides comprehensive auditing for every access to passwords and to privileged session recordings. Auditing information is displayed in a simple intuitive interface that includes the following information for each activity:

Information	Displays
User name	The name of the user who accessed the password or recording file.
Password or recording file	The name of the password or recording file that was accessed.
Destination address	The IP address or DNS of the target location where the password was used.
Protocol	The protocol used to access the target machine.
PSM ID	The unique ID of the PSM server.
Session ID	The unique ID of the PSM session that recorded this activity.
Session duration	The duration of the logon session.

Users can view an overall audit in the MONITORING page, and play privileged session recordings. Users can search for recordings using a free text search according to the properties that are associated with the privileged session (e.g. password, user, address, device, machine, or any other password keyword).

The screenshot shows the 'MONITORING' page in the Vault interface. The top navigation bar includes 'POLICIES', 'ACCOUNTS', 'MONITORING', 'APPLICATIONS', 'REPORTS', and 'ADMINISTRATION'. The user is logged in as 'DanielleB' and can 'Sign out'. The main area is titled 'Search for Sessions' and contains several search filters: 'Search for Sessions:', 'Search for Commands and Events:', 'Search for sessions between:' (with date pickers), and 'Search for recordings' (selected) vs 'Search for live sessions'. Below the filters is a 'Search recordings: All recordings' table with the following data:

User	Client	Account User Name	Account Address	Account Policy ID	Start	Duration	Vid
Ron	RDP	Administrator	1.1.1.12	WinDesktop.local	8/8/2013 4:07:57 PM	00:01:10	28...
Ron	RDP	Administrator	1.1.1.12	WinDesktop.local	8/8/2013 4:02:18 PM	00:03:10	63...

The interface also shows a 'Views' sidebar with 'Live Sessions' and 'My Views' (containing 'Search recordings: All recordings'). At the bottom, it indicates 'Page 1 of 1' and 'Displaying recordings 1 - 2 of 2'.

In addition, more detailed audit information is available for individual account use in the Activities tab of each Account Details page.

Disaster Recovery

The PSMP benefits from Disaster Recovery features which provide seamless productivity during a failover. For more information about Disaster Recovery in the Vault, refer to *CyberArk Disaster Recovery Vault*, page 885.

Transparent Failover

As soon as the PSMP cannot reach the Production Vault, the failover process begins in the DR Vault transparently, and no human intervention is required.

The IP address of both the Vault and the DR Vault can be specified in the Vault.ini configuration file. When the PSMP cannot reach the Vault specified by the first IP address, it transfers automatically to the Vault specified by the second IP address, which is the DR Vault.

To Configure Transparent Failover

1. In the Vault.ini file, in the **Address** parameter, specify the IP addresses of the Vault and the DR Vault, separated by commas, as shown in the following example:

```
Address=1.1.1.102,1.1.1.232
```

The above example indicates that the IP address of the Production Vault is 1.1.1.102 and the IP address of the DR Vault is 1.1.1.232.

2. Add the **SwitchVaultAddressTimeOut** parameter.

This parameter specifies the number of seconds that the PSMP will try to access additional Vault IP addresses after the initial timeout to the current Vault, specified in the **Timeout** parameter, expires.

If this parameter is not added, the default value of three seconds will be applied.

3. Save the Vault.ini file and close it.

On-Demand Privileges Manager

CyberArk's On-Demand Privileges Manager (OPM) enables organizations to secure, control and monitor privileged access to UNIX commands by using the Vault technology to allow end users to perform super-user tasks with their own personal account, whilst maintaining the least-privilege concept.

This chapter introduces you to the OPM, describes how it works and how you can configure and begin working with it.

This chapter comprises the following sections:

- Introduction
- Architecture
- Implementing the On-Demand Privileges Manager
- Enabling the On-Demand Privileges Manager
- Defining Privileged Commands
- Configuring the On-Demand Privileges Manager
- Issuing Privileged Commands When There Is No Direct Connection to the Vault
- Integrating with UNIX Centralized User Management Products
- Disaster Recovery
- Accessing the Password Vault
- Managing the OPM
- Auditing
- Monitoring
- Administrating the On-Demand Privileges Manager

Introduction

In many organizations, multiple users have permanent and continuous, yet anonymous, super user privileges. As a result, too many people have the potential to access business critical systems and data that are not part of their day-to-day role or responsibilities.

While it is essential for employees to have access to privileged accounts in order to work seamlessly and productively, organizations cannot know or control who accesses their business-critical systems and information, when and why they access them, and what actions they take. The fact that employees must be able to access and use such powerful and sensitive accounts raises multiple security and access concerns, as well as tracking and compliance issues.

CyberArk's On-Demand Privileges Manager (OPM) provides a comprehensive solution that empowers IT and enables complete visibility and control of super users and privileged accounts across the enterprise. Using the OPM, the complete Privileged Account Security solution enables centralized management and auditing from a unified product to all aspects of privileged account management.

The privileged account can be accessed in the following ways:

- **Audited and secured login with root accounts** – Users can log on as the root account using the root password. In order to do this securely, CyberArk utilizes the EPV for managing the root account's password and the PSM for seamlessly connecting with the root account in an audited manner.
- **On-demand access** – Users log into the UNIX machine with a non-privileged personal account, and when required, they request the privileged account's (e.g. root) root permissions on-demand to elevate a session to a root session. Although UNIX systems provide the built-in *sudo* command which allows on-demand elevation to root, the *sudo* command is not an enterprise-class option, and has challenges in centralized management and auditing. Furthermore, the *sudo* solution is a silo solution to the wider problem of privileged account management.

As part of CyberArk's Privileged Account Security solution, this solution benefits from the following features:

- **Platform-based granular access** – Access to each privileged command on the UNIX host systems is permitted according to an extremely granular set of permissions that are defined in the highly secured Vault server. Access control rules can be assigned to Privileged Account Security solution platforms and can be overridden for specific managed accounts. This facilitates a 'least privileges' scenario and limits root account use to very specific tasks. Users can execute privileged commands from a native interface according to platform-based access permissions.
- **Centralized audit** – All privileged commands activities are audited and stored in the tamper proof Vault server, and can be viewed in reports that can be customized to meet your enterprise standards and audit compliance requirements. You can choose from a variety of reports and contents, and determine the report output format.

- **Recordings** – Activities that are performed in privileged sessions are recorded and uploaded to the Vault, where they can be accessed and viewed by auditors and other authorized users at any time. These recordings are stored in the tamper proof Vault server and can be used as a valuable audit source. The inherent separation of duties in the Vault server assures that recordings are only available to authorized parties.
- **Centralized management** – All management tasks for users and accounts are centralized and streamlined in the PVWA, including account definitions and policies. Audits can also be accessed and managed in the PVWA, as well as privileged session recordings. This combination of management features provides a centralized, full-life cycle solution for privileged accounts.
- **Avoids exposing root passwords** – CyberArk’s granular access permissions enable you to grant users permission to use the root account without actually viewing its password. Privileged passwords remain secret while, at the same time, usable. Users can access shells and carry out commands using passwords that they cannot see, but are permitted to use.
- **Restricted Shell** – Authorized users can access fully delegated root shells and work intuitively according to their regular workflow, while the privileged command features of the OPM are enforced. As the entire session is executed transparently through the OPM, commands can be restricted for specific users and a complete audit is stored in the Vault.
- **Automatic User Provisioning** – The OPM can be configured to integrate with Microsoft’s Active Directory (AD) to provision users transparently on UNIX systems, streamlining user management and reducing administrative overhead. Users have immediate access to UNIX machines, based on their AD permissions and groups, facilitating an uninterrupted workflow and maintaining productivity. As their user is automatically synchronized with a corresponding user in the Vault, all user activity is monitored. For more information, refer to *Integrating with AD Bridge Capabilities*, page 724.
- **restrict superuser’s write-access** – When a user is elevated to superuser (root) to execute a privileged command, OPM does not give the superuser automatic access to every file on the system (i.e. according to the OS file system permissions). The superuser write operation is subject to authorization checking based on a defined policy according to the OPM file access lists restriction. If Security Managers do not specifically authorize a user to access a file during command elevation, OPM will prevent any write action on the file accessed based on the defined policy.

OPM leaves the UNIX file system permissions intact but adds a layer of enhanced access control to it. OPM intercepts each of the following file access operations and verifies that the user has authorization for the specific path before returning control to the OS. The access type is in parentheses.

- File /directory create (create)
- File open for write (write)
- File /directory delete (delete)
- File /directory rename (delete, rename)
- OS File link
- chmod command
- chown command

Overview

CyberArk's OPM enables users to granularly access and use privilege accounts according to command permissions (i.e., Access Control Lists – ACL) that define commands and access permissions. Each ACL determines the commands that can be issued by each user or group for a given account or an entire platform. Command permissions can be created and managed in the PVWA in a simple and straightforward process.

Authorized users can issue privileged commands from their Unix machine according to the ACL defined in the Vault, and elevate their standard login session to a privileged session so that they can run privileged commands.

A Unix IT daily workflow typically consists of the following steps:

1. **Standard login** – Users log into non-privileged personal accounts, and perform all the tasks that their user is permitted to do.
2. **Elevate to an on-demand privileged session** – At some point during the day, this user will probably need to use a privileged account to log onto a privileged session and perform a particular task.

The CyberArk 'pimsu' utility that enables users to run specific privileged commands on behalf of a privileged account, using their own user account. After the command has been carried out, the privileged session returns to a regular session. Authorized users can also open a full root shell session or a restricted shell session. For more information about restricted shells, refer to *Working in a Restricted Shell*, page 767.

The following Vault users each play an important part in the OPM workflow:

- **Administrators** – These users configure the OPM, and define the ACLs that enable users to perform privileged commands. They configure privileged commands in Unix policies and determine which users and groups will be permitted to use them, and the users and groups who will not be authorized. Exceptions to these configurations can be specified in individual accounts. For more information, refer to *Defining Privileged Commands*, page 748.
- **IT users** – These users log onto their Unix system at their own workstations and perform privileged tasks using an account stored in the Password Vault, according to the privileged command permissions that have been configured by an Administrator. For more information, refer to *On-Demand Privileges for UNIX Environments*, page 310.
- **Auditors** – Auditors can use the PVWA as a centralized administration point, where they can view activity records and recordings for all privileged sessions that have been performed using accounts stored in the Vault. For more information, refer to *Auditing*, page 783 and *Monitoring*, page 784.

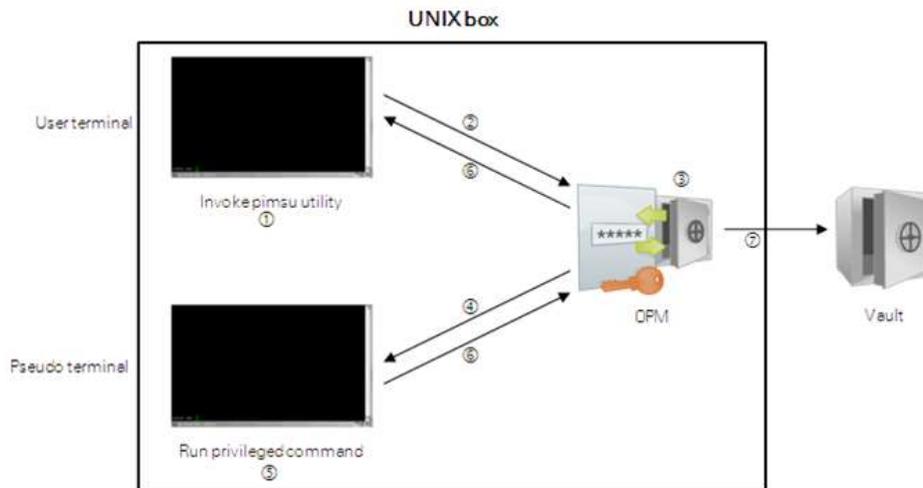
Architecture

Users who require privileged account privileges in order to execute a privileged task invoke the OPM pimsu utility which refers to the local OPM to start the privileged session. The OPM maintains a local cache that contains the access control details that permit each user to invoke the specific privileged commands that they requested and no other.

The OPM uses a unique user to access the Vault, retrieve the access control details, and store the session recordings and audit information. This user must be defined in the Vault and must have relevant access permissions in the Safe where the accounts are stored.

The OPM constantly refreshes its cache from the Vault, so that it always contains accurate information. It maintains audit logs and session recordings, so that there is complete accountability for each privileged command request by every user, and monitoring logs that register OPM activity and status.

This solution provides high availability and business continuity, regardless of Vault or network availability. The local cache eliminates the need to access the Vault for every privileged command invocation and raises the level of performance, especially at remote sites.



When the user needs to perform a privileged task, they invoke the OPM pimsu utility from their terminal (step 1). The pimsu utility connects to the OPM (step 2) that checks whether the user has permission to access the account required to perform this task or run this session (step 3). If the command permissions in the Vault give the user the appropriate permissions to run this task, the OPM automatically opens a privileged session on a pseudo terminal (step 4) without exposing the root password to the user. The OPM runs the privileged command (step 5) and redirects the input / output of the command to the user's terminal (step 6) where the user can follow the process of the command.

The OPM records the entire privileged session. When the session has been completed, the recording is uploaded into the Vault (step 7) where it can be accessed by authorized users.

Implementing the On-Demand Privileges Manager

In the PVWA:

1. Configure the Safe where privileged account passwords will be stored. For more information about configuring Password Safes, refer to *Adding Safes*, page 570.
2. In the Password Safe, add the privileged accounts that will be required to run the privileged commands. You can either do this in either of the following ways:
 - **Manually** – Add accounts manually one at a time, and specify all the account details.
 - **Automatically** – Add multiple accounts automatically using the Password Upload feature.

Before adding privileged accounts, all the standard considerations for controlling and managing these privileged accounts must be defined within the context of the OPM implementation and the timeframe in which passwords must be supplied.

- **Access Control** – The users that require access to these accounts, including OPM, automated, and human users.
- **Workflows** – The workflows that will be used for accessing these accounts.
- **Compliance** – The compliance requirements for managing these accounts, and whether an account management platform has already been defined for these accounts.
- **Account Management** – The methods that will be implemented to enforce the defined compliance requirements.
- **Monitoring and auditing** – The way that the enterprise monitors the system and ensures that policies are enforced properly.

For more information about adding and managing privileged accounts, refer to the *Working with the Privileged Account Security Solution* chapter, page 25.

3. Add the OPM user and the end users as Members of the Password Safes where the privileged passwords are stored. This can be done manually in the Safes page.
 - i. Add the OPM user as a Safe Member with the following authorizations:
 - List accounts
 - View Safe Members
 - ii. Add the users who will run preconfigured privileged command(s) as Safe Members with the following authorizations:
 - Use accounts

Note: If this user belongs to a group, make sure that the group has this authorization.

- iii. If the Safe is configured for object level access, make sure that the end user has access to the password(s) to use with the following authorization:
 - Use accounts

For more information about configuring Safe Members, refer to *Creating and Managing Safes and Owners*, page 64.

4. Configure the privileged commands, either at platform level or at account level:
 - **Platform level** – In the platform that will manage accounts that will be used to run privileged commands, configure the commands that users will be able to issue.
 - **Account level** – In the Account Details page of the account that will be used to run privileged commands, in the Commands tab, add a new command or specify changes to the command that was configured at platform level.

For more information, refer to *Defining Commands*, page 749.

On the OPM machine:

5. The OS user (end user) requires read and write permissions in the /tmp folder. If this does not conform to enterprise policy standards, an alternative Temp folder can be specified in the following environment variable:
 - AIM_TEMP_FOLDER

This folder must be available to the OPM pimsu utility as well as to the OPM.

If you specify a different folder, the OPM pimsu utility requires write and execute permissions on the folder. The OPM has these permissions by default as it runs under root.

Enabling the On-Demand Privileges Manager

After installing the OPM, it must be enabled for each platform so that you can configure and implement on-demand privileges for privileged Unix commands. The OPM can be configured so that all the accounts associated with a certain platform can use the same privileged commands or at account level for specific commands, depending on the ACL.

Users who are members of the Vault Admins Group can enable and configure the On-Demand Privileges Manager. This group can be changed by modifying the `PimsuPoliciesManager` parameter in `DBParm.ini`.

In order to manage ACLs at account level, users do not need to be a member of the Vault Admins group, but they require the following permissions in the Safe where the privileged account is stored:

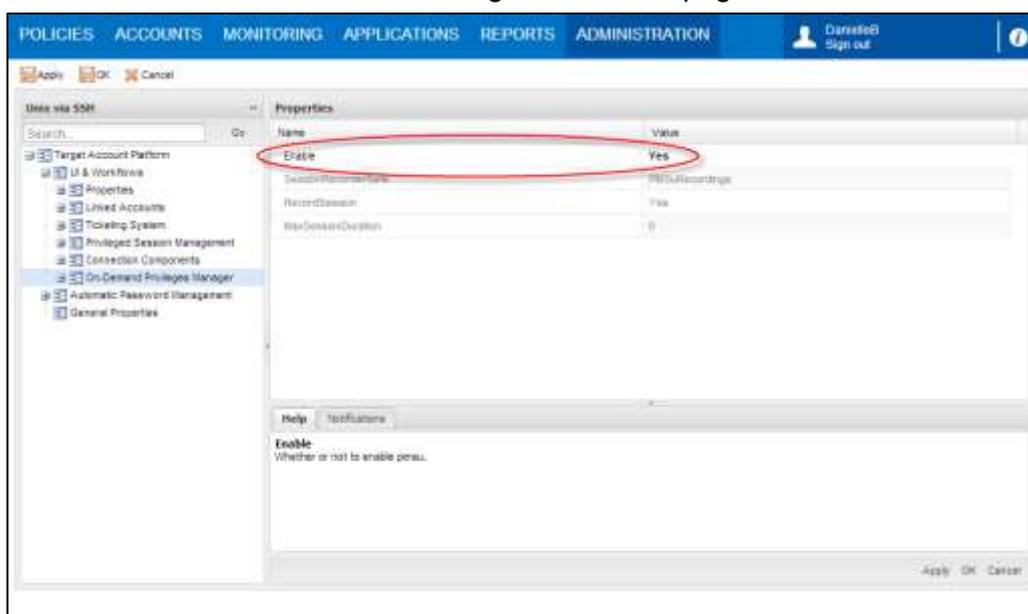
- Manage Safe members
- Use accounts

To Enable the On-Demand Privileges Manager

1. Click **ADMINISTRATION** to display the System Configuration page, then click **Platform Management** to display a list of supported target account platforms.
2. Select the platform to configure for On-Demand Privileges, then click **Edit**; the settings page for the selected platform appears.
3. Expand **UI & Workflows**, and then select **On-Demand Privileges Manager**; the OPM parameters are displayed with their default values.
4. In the Properties list, change the value of **Enable** to **Yes**; the On-Demand Privileges Manager has been enabled and can be configured for the selected platform.

In addition, commands can be configured for specific accounts at account level.

For more information, refer to *Defining Commands*, page 749.



Defining Privileged Commands

Privileged commands that are defined in the PVWA determine which commands can be run by users natively in their UNIX environment using their own account.

Privileged commands can be configured at the following levels:

- **Platform level** – Users and groups can execute privileged commands with any account associated with a specific platform.
- **Account level** – Users and groups can execute privileged commands with specific accounts.

Each privileged command definition includes the command's pattern that designates the Unix command that is permitted, for example, `/usr/bin/kill`. In addition, the permission rules include the following properties:

- **Deny/Allow** – Each ACL can be defined as **Allow** or **Deny** permission. **Allow** permission adds the command to the whitelist of allowed commands. The **Deny** permission adds the command into a blacklist and prevents users from running it. When multiple permissions are applicable for specific commands, if one of them has the **Deny** permission, the command will be blacklisted, even if **Allow** permissions are applied. For example:
 - A user has the **Allow** permission to run the `"/bin/kill"` command at platform level.
 - The same user has the **Deny** permission to run the `"/bin/kill"` command at account level.
 - As the **Deny** permission added the `"kill"` command to the blacklist, this user will not be allowed to run the `"kill"` command.
- **Restrictions** – Restrictions allow users to granularly limit or control the command permission. For example, you can control the authentication required to issue a command or how the UNIX environment variables will be passed to the command.

Restrictions are applied according to the following rules:

- **Yes/No/AlwaysNo** – Restrictions that specify **Yes** override **No** for the same restriction at a different level. For example, a restriction in a command whose value is **No** at platform level, but **Yes** at account level, will be permitted for the specified account. However, **AlwaysNo** overrides **Yes**, so if a command restriction's value is **Yes** at account level, but is **AlwaysNo** at platform level, the OPM will override the **Yes** value and apply a **No** value.
- **Text values** – The OPM combines the text values specified for restrictions and applies the combination.

For more information about restrictions and their possible values refer to *Restrictions*, page 757.

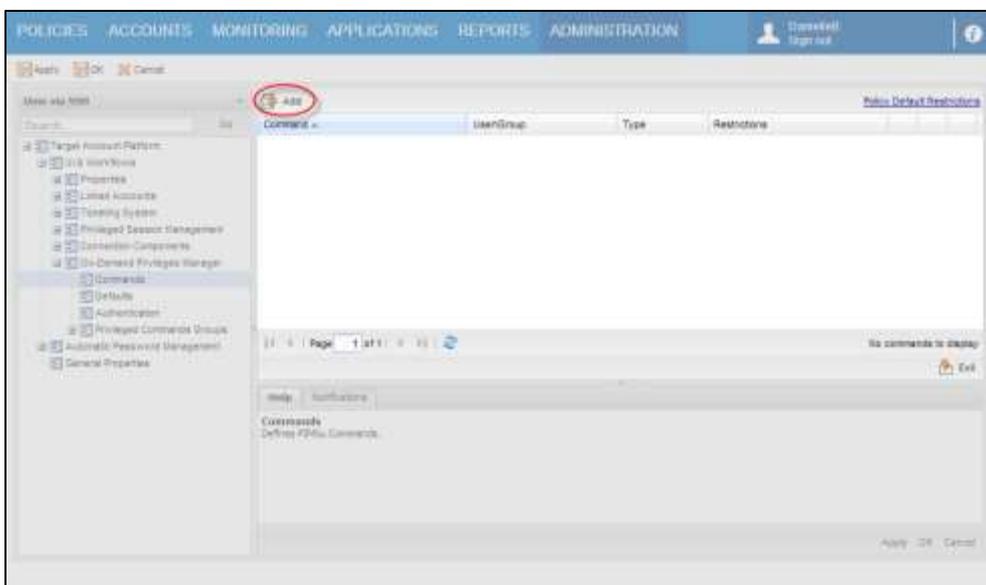
Defining Commands

Defining Commands at Platform Level

Commands that are defined at platform level can be issued using any of the accounts that are managed by the configured platform. Users who are members of the **Vault Admin** group can define commands at platform level.

To Define Commands at Platform Level

1. Click **ADMINISTRATION** to display the **System Configuration** page, then click Platform Management to display a list of supported target account platforms.
2. Select the platform to configure, then click **Edit**; the settings page for the selected platform appears.
3. Expand **UI & Workflows**, and then expand **On-Demand Privileges Manager**; the OPM parameters are displayed with their default values.
4. Click **Commands**; the Command grid is displayed.



5. Click **Add**; the Add Command window appears. This window enables you to specify a privileged command group or a specific privileged command.
6. Specify the command and its definitions according to the procedure described in *Adding Privileged Command Definitions*, page 751.

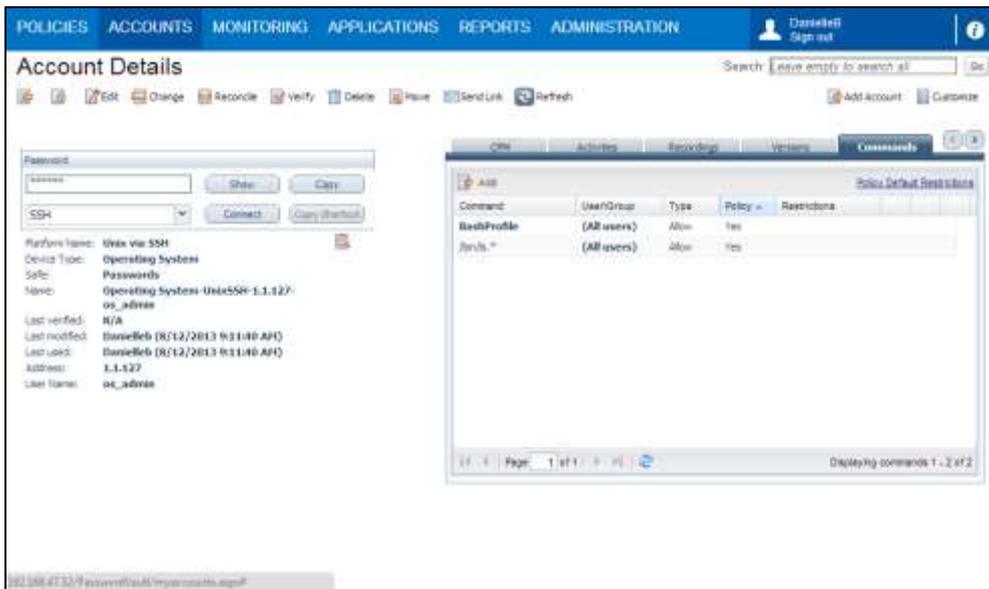
Defining Commands at Account Level

Users who have the following permissions in the Safe where the account is stored or OLAC permissions for the account itself can define and modify commands at account level:

- Manage Safe members
- Use accounts

To Define Commands in an Account

1. In the Account Details page of the privileged account to configure, display the Commands tab.



In the above example, the Policy column indicates that the command was defined at platform level.

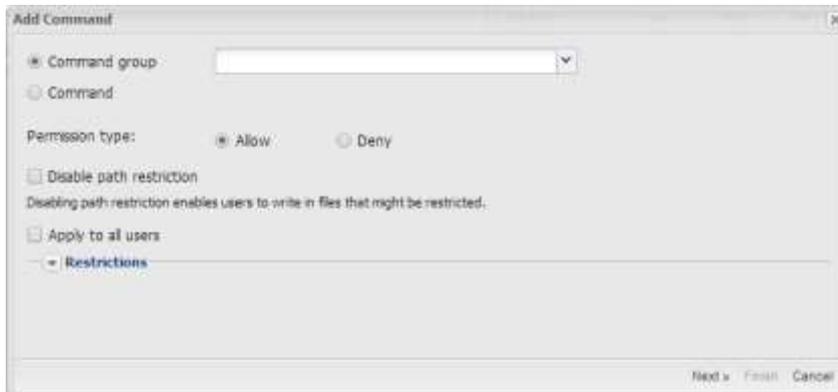
2. Click **Add**; the Add Command window appears. This window enables you to specify a privileged command group or a specific privileged command for this privileged account.
3. Specify the command and its definitions according to the procedure described in *Adding Privileged Command Definitions*, page 751.

Adding Privileged Command Definitions

1. In the Add Command window, specify the command group or command.

To specify a command group:

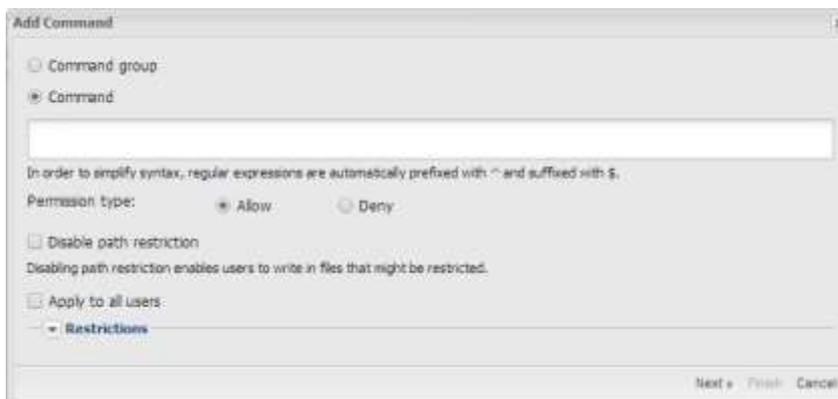
- Select **Command group**, then from the drop-down command group list, select the command group that contains the commands you will allow or deny for this user or group.



For more information about command groups, refer to *Defining Privileged Command Groups*, page 762.

To specify a privileged command:

- Select **Command**, then specify the privileged command pattern to define in the Vault.



The command pattern is a regular expression. In order to simplify the notation, the command is automatically prefixed with ^ and suffixed with \$ (e.g. `/bin/kill .*` will be automatically translated to `^(/usr/bin/kill .*)$`).

The command pattern must include the full path of the privileged command. For example, in order to permit the **kill** command, the command pattern must include the full path (e.g. **/bin/kill**)

If the command includes parameters, these parameters must be explicitly specified in the regular expression. For example, in order to permit the **/bin/kill** command with any parameter, the command pattern must specify **/bin/kill .***.

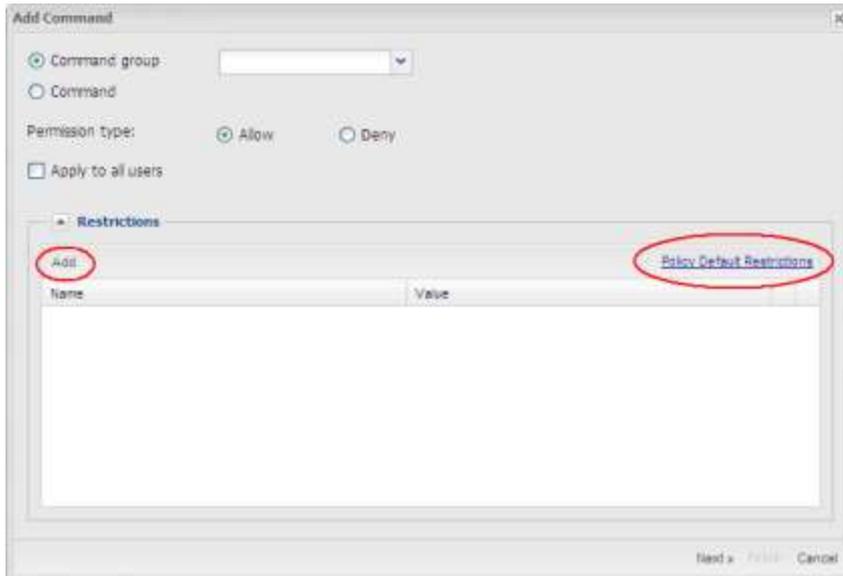
2. To allow users to issue the selected command group or the specified command, select **Allow**,

or,

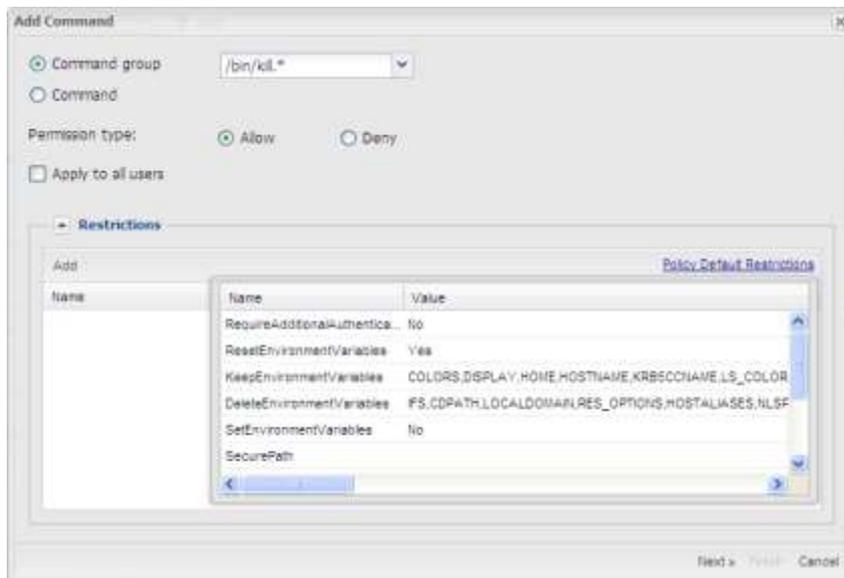
To prevent users from issuing this command, select **Deny**.

3. To enable write operations by the requested command group or command in files or folders that may be restricted, select **Disable path restriction**. For more information about specifying this path, refer to AllowWritablePath in the *Restrictions*, on page 757.
4. To apply this setting to all users, select **Apply to all users**.
5. To view the default platform restrictions that will be applied to this command and specify customized restrictions for this command, display the drop-down **Restrictions** list.

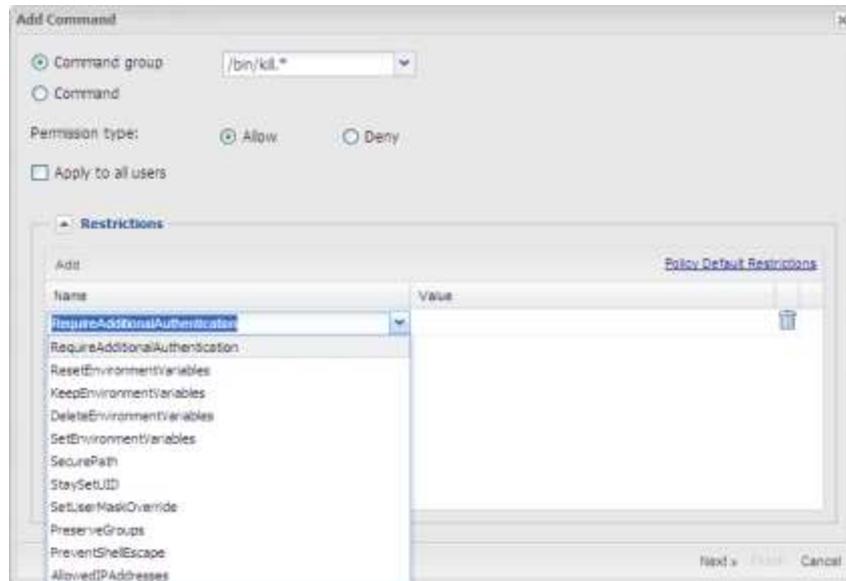
For more information about restrictions and their values, refer to *Restrictions*, page 757.



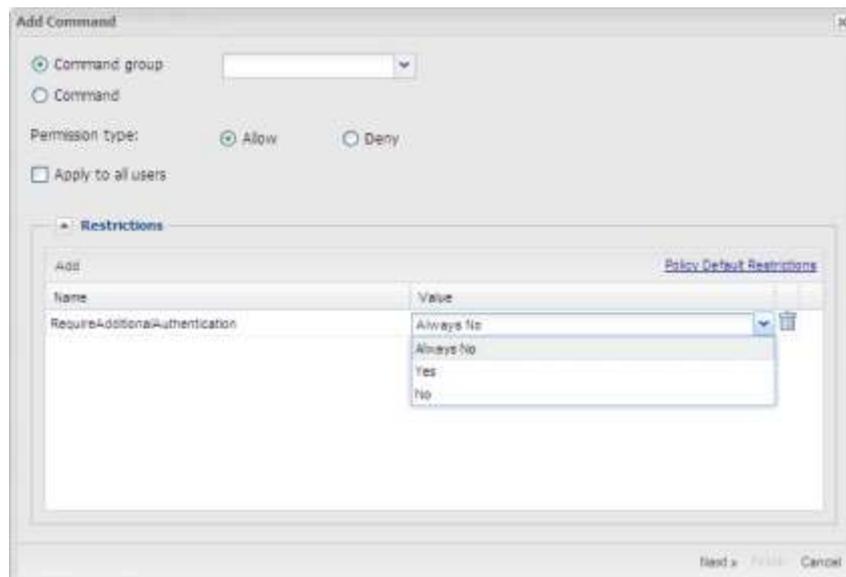
- To view the default platform restrictions that will be applied to this command, click **Policy Default Restrictions**.



- To customize a restriction for this command:
 - i. Click **Add**, a new line is added to the Restrictions grid.
 - ii. From the Name drop-down list, select the command restriction to customize.

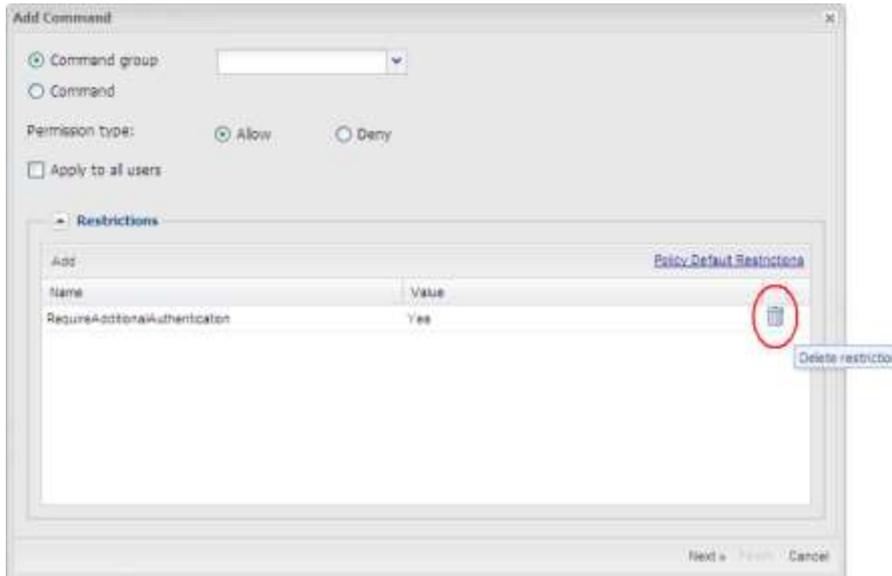


- iii. From the Value drop-down list, select the value to apply to the restriction or specify a text value.

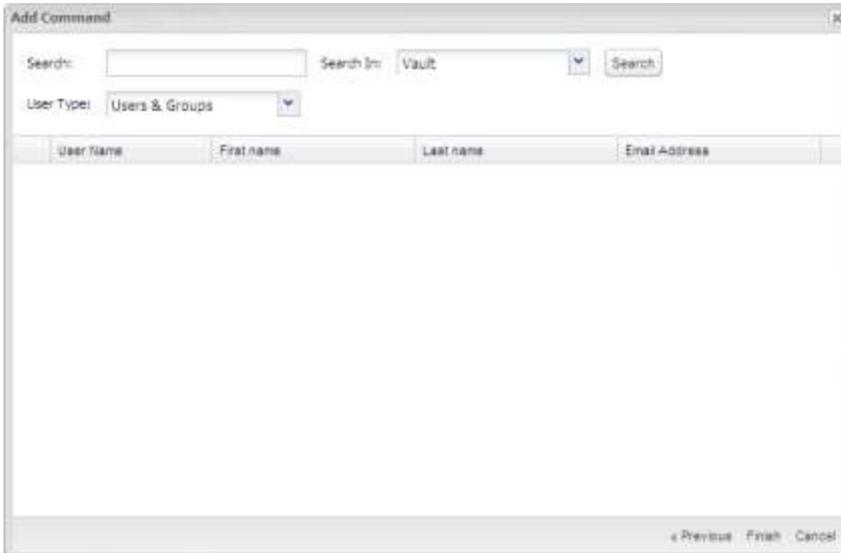


For more information about restrictions and their values, refer to *Restrictions*, page 757.

- To delete a customized restriction, click **Delete Restriction**.

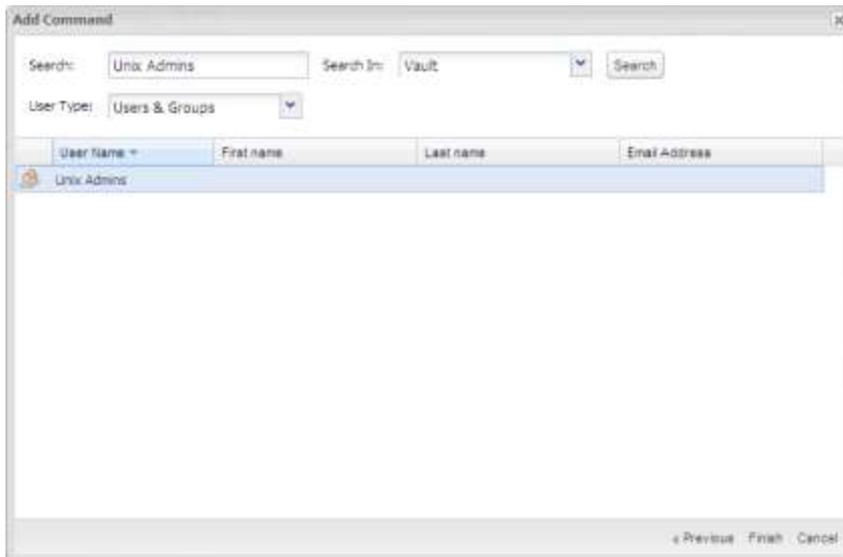


- If **Apply to all users** is selected, the **Next** button is disabled and you will only be able to click **Finish**.
 - If **Apply to all users** is not selected, click **Next** to display the Users & Groups window and select the users and groups who will be allowed or denied use of this command group or command.
6. If you clicked **Next**; the Search for Users & Groups window appears.



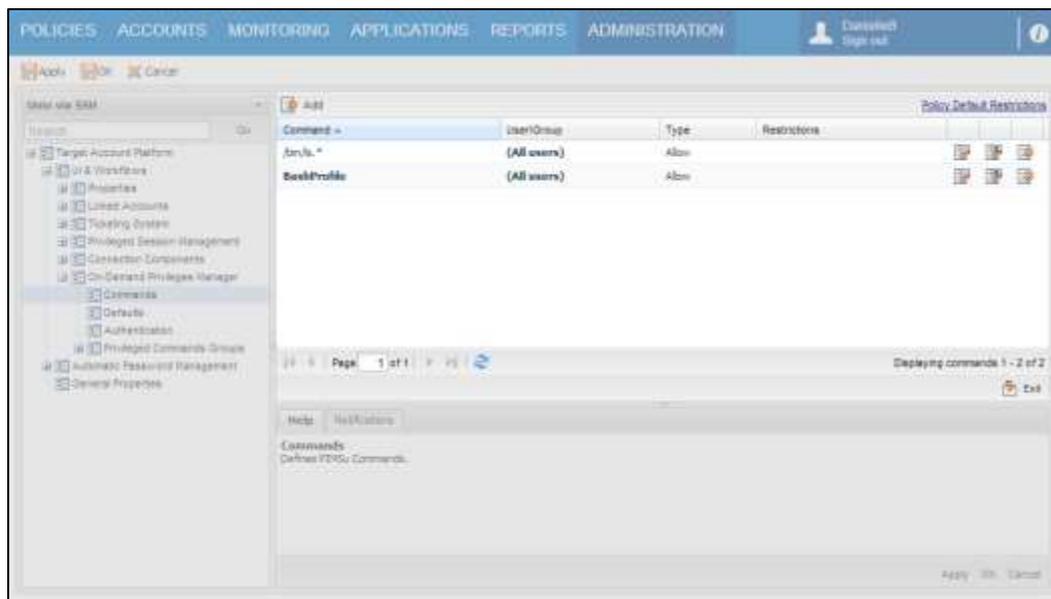
7. Search for the user or group that will be able to issue this command:
- i. In the **Search** edit box, specify all or part of the name of the user or group to search for.
 - ii. From the **Search In** drop-down box, select either the Vault to search in or the name of the external directory to search for the user or group.
 - iii. From the **User Type** drop-down box, select the user type to search for.

- iv. Click **Search**; a list of users and/or groups that meet these criteria is displayed.

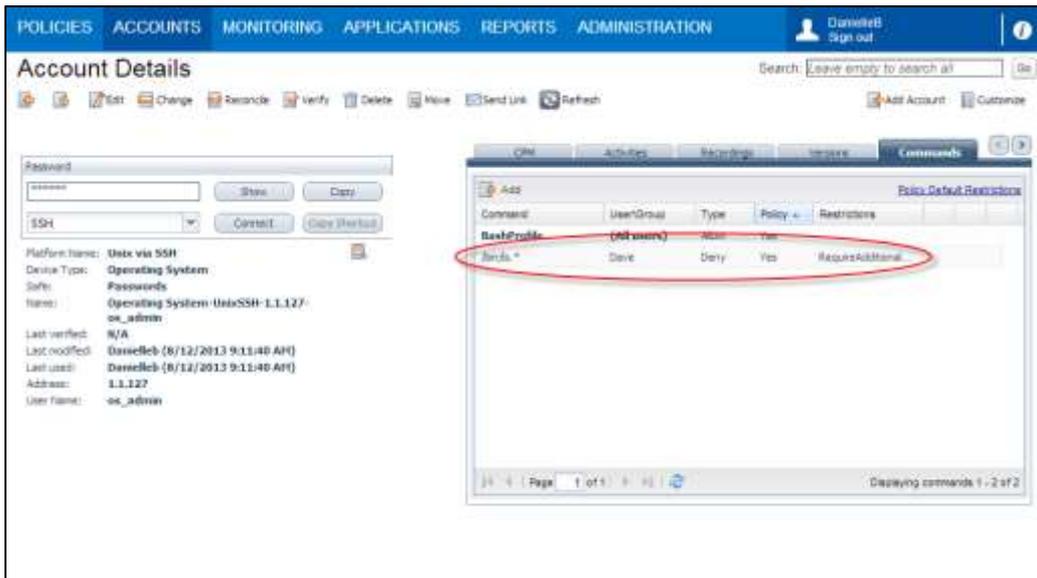


8. Select the user or group to configure for this command, then click **Finish**; the command is displayed in the commands list with the user or group that is authorized to use it.

The following example shows the commands list at platform level:



The following example shows the commands list at account level:



The Policy column indicates the command that was defined at platform level, and the restriction that was set for the user called Dave appears in the Restrictions column.

- Click on the line of the command in the grid to view the command properties and restrictions.



- Click **Close** to close the Command Properties window.

At platform level:

- The Commands grid is displayed. Click **Exit** to exit the Commands editing mode and return to the platform parameters configuration mode.

At account level:

- The Account Details page is displayed.

Restrictions

Restrictions allow users to granularly limit or control the command permission. For example, you can control the authentication required to issue a command or how the UNIX environment variables will be passed to the command.

The following table lists the parameters in the **Defaults** section that define the privileged commands restrictions. The default values can be changed at platform level and overridden in commands at either platform or account level.

For more information about applying and modifying restrictions at both platform and account level, refer to *Defining Privileged Commands*, page 748.

Notes: Some of the parameters in the following table enable you to specify **AlwaysNo**. When this value is specified, 'No' will always be applied, regardless of the value specified at either platform or account level.

When working inside a restricted shell, all the restriction parameters are inherited from the shell, except **RequireAdditional Authentication** and **AllowedIPAddresses**.

Parameter	
RequireAdditionalAuthentication	
Description	Whether or not users will be required to provide authentication to authenticate to the Vault. If no authentication is required, SSO authentication will be applied. For more information, refer to <i>Authentication</i> , page 765.
Acceptable Values	Yes/No/AlwaysNo
Default Value	Yes
ResetEnvironmentVariables	
Description	Whether or not the OPM will reset the environment variables of the privileged command that will be executed to only contain the USER, USERNAME, LOGNAME, SHELL, HOME environment variables and additional environment variables that are specified in the KeepEnvironment Variables restriction.
Acceptable Values	Yes/No/AlwaysNo
Default Value	Yes
KeepEnvironmentVariables	
Description	List of environment variables separated by commas to be preserved in the user's environment when the ResetEnvironment Variables restriction is in effect.
Acceptable Values	String
Default Value	COLORS,DISPLAY,HOME,HOSTNAME, KRB5CCNAME, LS_COLORS, MAIL,PATH, PS1, PS2, TZ, XAUTHORITY, XAUTHORIZATION, LANGUAGE,LANG,LC_ALL,LC_COLLATE, LC_CTYPE,LC_MESSAGES, LC_MONETARY, LC_NUMERIC, LC_TIME, COLORTERM,TERM, LINGUAS

Parameter	
DeleteEnvironmentVariables	
Description	List of environment variables separated by commas to be removed from the user's environment.
Acceptable Values	String
Default Value	IFS,CDPATH, LOCALDOMAIN,RES_OPTIONS,HOSTALIASES,NLSPATH,PATH_LOCALE,LD_LIBRARY_PATH,TERMINFO,TERMINFO_DIRS,TERMPATH,TERMCAP,ENV,BASH_ENV,PS4,GLOBIGNORE,SHELLOPTS,JAVA_TOOL_OPTIONS,PERLIO_DEBUG,PERLLIB,PERL5LIB,PERL5OPT,PERL5DBFPATH,NULLCMD,READNULLCMD,ZDOTDIR,TMPREFIX,PYTHONHOME,PYTHONPATH,PYTHONINSPECT,RUBYLIB,RUBYOPT
SetEnvironmentVariables	
Description	Whether or not the OPM will enable end users to override environment variables restrictions by specifying the <code>-E</code> switch in the <code>pimsu</code> command line.
Acceptable Values	Yes/No/AlwaysNo
Default Value	No
SecurePath	
Description	The <code>PATH</code> environment variable that will be used for every command executed by OPM.
Acceptable Values	String
Default Value	-
StaySetUID	
Description	Whether or not the UID used for commands executed by OPM will be set to the invoking user's UID instead of the privileged user's UID.
Acceptable Values	Yes/No/AlwaysNo
Default Value	No
SetUserMaskOverride	
Description	The user mask (<code>umask</code>) that will be used for every command executed by OPM. This value must represent a valid octal value between 0000 and 0777.
Acceptable Values	String
Default Value	-
PreserveGroups	
Description	Whether or not the groups list used for commands executed by OPM will be set to the invoking user's groups list instead of the <code>RunAs</code> user's groups list. This does not affect the user's real/effective group id.
Acceptable Values	Yes/No/AlwaysNo
Default Value	No

Parameter	
PreventShellEscape	
Description	Whether or not to prevent a program run by OPM from executing any other programs.
Acceptable Values	Yes/No/AlwaysNo
Default Value	No
AllowedIPAddresses	
Description	Defines the IP/DNS/ hostname on which the privileged command is allowed to be executed. This may be a range of IP addresses, specific addresses, or a combination of them. This parameter is optional.
Acceptable Values	Valid IP address
Default Value	-
ExtendedProtection	
Description	Whether or not extended protection will be enabled.
Acceptable Values	Yes/No/AlwaysNo
Default Value	Yes
RestrictedShell	
Description	Whether or not the command will be executed in a restricted mode. This restriction is only relevant for commands that execute other commands internally like shells or scripts. Note: When working inside a restricted shell, all the restriction parameters are inherited from the shell, except RequireAdditional Authentication and AllowedIPAddresses .
Acceptable Values	Yes/No/AlwaysNo
Default Value	Yes
PasswordPrompts	
Description	Regular expression to identify password prompts in order to hide from the OPM recording, passwords that are typed by the user during a pimsu session.
Acceptable Values	REGEXP
Default Value	[Pp]assword: . *s [Pp]assword: Old [Pp]assword: Enter existing login [Pp]assword: New [Pp]assword: Enter the new [Pp]assword again: Changing password for.*New [Pp]assword: Re-enter new [Pp]assword: Enter choice here: \(current\) UNIX [Pp]assword: New UNIX [Pp]assword: Retype new [Pp]assword: Retype new UNIX [Pp]assword: Vault [Pp]assword: Target [Pp]assword:
AllowWritablePath	
Description	Defines the path of a file or a folder that will be enabled for write operations by the requested command. Specify * at the end of the path to include subfolders and files.
Acceptable Values	String
Default Value	<ul style="list-style-type: none"> ▪ At platform level – * (asterisk) ▪ At account level – empty

Parameter	
DenyWriteablePath	
Description	Defines the path of a file or a folder that will be denied write operations by the requested command. Specify * at the end of the path to include subfolders and files.
Acceptable Values	String
Default Value	<ul style="list-style-type: none"> ▪ At platform level – /etc/*,/bin/*,/usr/bin/*,/usr/local/bin/*,/sbin/*,/usr/sbin/*,/usr/local/sbin/* ▪ At account level – empty

Modifying Command Permissions

After a command has been configured, authorized users can modify commands as well as the command permissions which determine which users and groups can issue or are denied from issuing each command.

Commands that were defined at platform level can only be modified in the platform settings page, whereas commands that were defined for specific accounts are modified in the Account Details page.

To Modify Commands

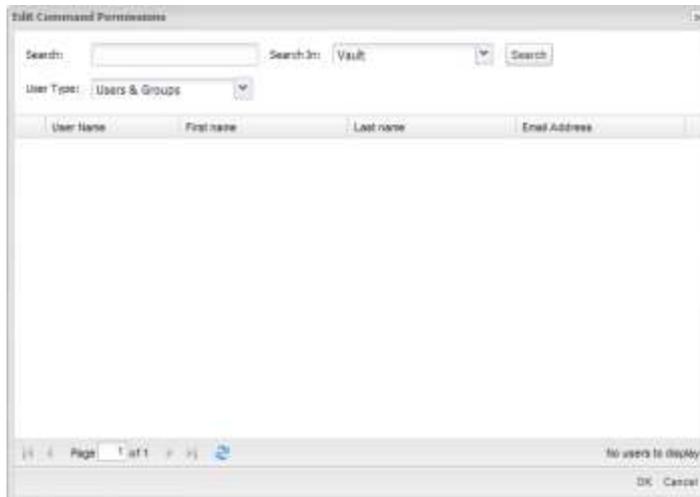
1. In the Commands list, click the **Edit Command** icon on the line of the command to modify; the Edit Command window appears.



2. Modify the command, permissions, and restrictions as necessary, then click **OK**; the command is modified and the Commands list is displayed again.

To Modify Command Permissions

1. In the Commands list, click the **Edit Command Permissions** icon on the line of the command to modify; the Edit Command Permissions window appears and enables you to modify the users and groups who will be able to issue or will be denied from issuing this command.



2. Specify a new search and select a new user or group, then click **OK**; the user or group is modified in the command and is now authorized or denied permission to issue this command and the Commands list is displayed again.

Deleting Commands

When users no longer need to use privileged commands through the OPM, authorized users can delete them from the Commands list.

To Delete Commands

1. In the Commands list, click the **Delete command** icon on the line of the command to delete; the Delete Command window appears and prompts you for confirmation to delete the selected command.
2. Click **Yes**; the command is deleted from the Commands list and users cannot access it with accounts managed by this platform any more.

Defining Privileged Command Groups

Privileged command groups enable users to create lists of one or more commands, and to allow or deny users the ability to perform those commands in one step. Depending on how the command is defined in the command group, users can run these privileged commands with all or specific arguments, or without any arguments.

Users who are members of the Vault Admin group can define privileged command groups and apply them at platform or account level.

To Define a Privileged Command Group

1. Click **ADMINISTRATION** to display the **System Configuration** page, then click **Platform Management** to display a list of supported target account platforms.
2. Select the platform to configure, then click **Edit**; the settings page for the selected platform appears.
3. Expand **UI & Workflows**, and then right-click on **On-Demand Privileges Manager** and select **Add Privileged Commands Groups**; a new section called **Privileged Commands Groups** is added to the OPM parameters and enables you to manage command groups.
4. Right-click on **Privileged Commands Groups** and select **Add Privileged Commands Group**; a new section called **Privileged Commands Group** is added that enables you to create a new command group.
5. In the Name property, specify the name of the privileged command group to create, then press **Enter**.
6. Right-click on the new command group section, and select **Add Privileged Command**; a new parameter for the privileged command is created.
7. In the Name property, specify the command pattern (regular expression) of the privileged command to define, then press **Enter**.
Repeat this step as many times as necessary to define all the commands in this privileged command group.
8. Click **Apply** to save the new privileged command group and stay in the platform settings page,
or,
Click **OK** to save the new privileged command group and return to the System Configuration page.

Deleting Privileged Commands from Privileged Command Groups

Privileged commands can be deleted from a privileged commands group at any time by authorized users.

To Delete a Privileged Command

- Right-click on the Privileged Command to delete, and select **Delete**; the privileged command is deleted immediately from the command group.

Note: There is no message box to prompt confirmation before the privileged command is deleted.

Deleting Privileged Commands Groups

The privileged commands in a commands group can be modified at any time by authorized users. However, when a privileged commands group is deleted, the privileged commands specified in it cannot be used any more unless they are specified in another command group or individually.

To Delete a Privileged Commands Group

1. Right-click on the Privileged Commands Group to delete, and select **Delete**; the following message appears prompting you for confirmation to delete this command group.



2. Click **Yes**; the privileged command group is deleted from the list of Privileged Commands Group.

Configuring the On-Demand Privileges Manager

The OPM can be configured to provide a variety of authentication and display functions. All configurations can be specified by authorized users in the System Configuration page. For more information, refer to *Configuring the System through PVWA*, page 918.

Displaying the Commands List

The following parameters define the columns that will be displayed in the Commands tab in the Account Details pages. These parameters all appear in the **Privileged Session Management UI** parameters in the **Commands** section.

- **Displayed Columns** – The columns specified in this list will appear in the Commands tab. Each parameter that is defined in this section specifies the title of the column, the width of the column in the grid, and the data type of the information. You can also specify whether or not the column will be included in the grid.
- The **SortBy** parameter specifies the default column by which to sort the grid.

Configuring the On-Demand Privileges Manager in Platforms

The following features can be configured in platforms in the **On-Demand Privileges Manager** section .

OPM Recordings

The following parameters define the OPM recording settings.

- **SessionRecorderSafe** – The name of the Safe where all the recordings of privileged sessions enabled by the OPM will be stored. This Safe will be created when the first recording is uploaded to it. The default Safe name is **PIMSuRecordings**.
- **RecordSession** – Whether or not session recording is enabled for this platform. The default value is **Yes**.
- **MaxSessionDuration** – The maximum duration in seconds of the session. The default value is **0** (zero), indicating that the session duration is unlimited.

Authentication

Users can authenticate to the OPM with any of the following authentication methods:

- **Password** – Users are required to provide a password before they can perform privileged commands. The OPM can authenticate the following authentication types:

- Vault password
- LDAP
- RADIUS

In addition, users can authenticate with the authentication method that is configured for them, without forcing a specific method. This is useful when different users in the organization use different authentication methods.

- **SSO** – The OPM relies on the OS that authenticated the user for authentication to the Vault.

To Configure Authentication in the OPM

1. Click **ADMINISTRATION** to display the **System Configuration** page, then click Platform Management to display a list of supported target account platforms.
2. Select the platform to configure, then click **Edit**; the settings page for the selected platform appears.
3. Expand **UI & Workflows**, then expand **On-Demand Privileges Manager**, and then select **Authentication**; the Authentication parameters are displayed with their default values.
4. In **AdditionalAuthenticationMethod**, select the authentication method to use. Choose from the following valid values:
 - **Password**
 - **LDAP**
 - **RADIUS**
 - **Default** – Enables users to authenticate with the authentication method that is configured for them, without forcing a specific method. This is the default value.

This parameter is relevant when the **RequiredAdditionalAuthentication** restriction is set to **Yes**.

5. Click **Save** to save this configuration change,
or,
Click **OK** to save this configuration change and return to the main System Configuration page.
6. For LDAP or RADIUS authentication, configure the following in the OPM's Vault parameters file (Vault.ini) :
 - i. Add the following parameter to enforce a pre-authentication secured session:

```
PREAUTHSECUREDSESSION=YES
```

- ii. Add the following parameters to configure the Vault to trust Self-Signed Certificates:

```
TRUSTSSC=YES
ALLOWSSCFOR3PARTYAUTH=YES
```

For more information about the Vault parameter file, refer to the Privileged Account Security Reference Guide.

To Define Authentication Settings

The following parameter in the **Defaults** section determines whether or not users are required to provide additional authentication before they can run a privileged command.

- **RequireAdditionalAuthentication** – Whether or not additional authentication is required to enable users to authenticate to the Vault. If no additional authentication is required, the system will trust the UNIX host authentication. If additional authentication is required, the additional authentication method is determined by the **AdditionalAuthenticationMethod** parameter in the **Authentication** section. By default, additional authentication is required.

As **RequireAdditionalAuthentication** is a restriction, the default value can be overridden for each command permission at platform or account level. For information about how to modify restriction values, refer to *Restrictions*, page 757.

The following platform parameters in the **Authentication** section define the OPM authentication settings.

- **AdditionalAuthenticationMethod** – The authentication method that will be used if the **RequireAdditionalAuthentication** parameter in the default setting parameters is set to **Yes**. This parameter can be set to Password, LDAP, RADIUS, or Default at platform or account level. The default value is **Default**.
If the **RequireAdditionalAuthentication** parameter in the default setting parameters is set to **No**, but a user specifies a password when they run the pimsu utility, this password will be authenticated even though it is not required.
- **AllowHostAuthenticationWhenVaultIsDown** – Whether or not an OPM command can be invoked without a user password in emergencies (even if **RequiredAdditional Authentication** parameter is set to **Yes**) when there is no connection to the Vault. The default value is **No**.
- **AuthenticationGracePeriod** – The number of seconds that will elapse after users have authenticated to the Vault before the OPM will prompt for their password again. The default value is 300 seconds. If '0' (zero) is specified, users will always be prompted for their password.

Working in a Restricted Shell

Restricted shells enable users to access a fully delegated privileged shell, while enforcing the privileged command features of the OPM. Users can work intuitively in an elevated privileged shell, according to their standard workflow. Each command in this session is validated by the OPM, ensuring command-level access control and audit within the shell session. Preconfigured OPM platforms determine which commands users can perform, enabling enterprises to retain control over user activities in the delegated privileged shell. In addition, the entire shell session is recorded and stored in the Vault, and provides a complete audit of all commands that were performed.

Notes:

- The following cannot be run inside a restricted shell or as a restricted shell:
 - Static executables.
 - Commands marked with setuid flags – This is relevant when the restricted shell is delegated to a user other than root.
- In Linux, restricted shells can only be used if the `/etc/ld.so.preload` file is empty or is not configured.
- In Solaris, when opening a restricted shell, the `LD_FLAGS` and `LD_CONFIG` environment variables are reset.
- In AIX, when opening a restricted shell, the `LDR_FLAGS` and `LDR_CONFIG` environment variables are reset.

Enabling Restricted Shells

All the commands that will be executed in the restricted shell must be added as an ACL in the PVWA. This includes the following types of commands:

- **Commands that users will execute** – Commands that will be executed manually by users in the restricted shell.
- **Commands that run automatically during the shell startup** – Commands that run automatically when the restricted shell starts. This includes commands that run as part of the shell's profile scripts, such as `.profile`, `.bashrc`, etc.

To Enable Restricted Shells

1. Display the Add Command window:
 - To define commands at platform level:
 - i. Click **ADMINISTRATION** to display the **System Configuration** page, then click Platform Management to display a list of supported target account platforms.
 - ii. Select the platform to configure, then click **Edit**; the settings page for the selected platform appears.
 - iii. Expand **UI & Workflows**, and then expand **On-Demand Privileges Manager**; the OPM parameters are displayed with their default values.
 - iv. Click **Commands**; the Command grid is displayed.
 - v. Click **Add**; the Add Command window appears. This window enables you to specify a privileged command group or a specific privileged command.
 - To define commands at account level:
 - i. In the Account Details page of the privileged account to configure, display the Commands tab.
 - ii. Click **Add**; the Add Command window appears.

2. In the Add Command window, specify the required command to restrict. The command can be an independent command or part of command group.
 - To specify a command group:
 - Select **Command group**, then from the drop-down command group list, select the command group that contains the commands you will allow or deny for this user or group.
 - To specify a privileged command:
 - Select **Command**, then specify the privileged command pattern to define in the Vault.

For more information about specifying commands and its definitions, refer to *Adding Privileged Command Definitions*, page 751.

3. Expand the Restrictions list, then click **Add**; a new line is added to the Restrictions grid.
4. From the Name drop-down list, select **RestrictedShell**.
5. From the Value drop-down list, select **Yes**.
6. Click **Next**; the Search for Users & Groups window appears.
7. Search for the user or group that will be able to issue this command:
 - i. In the **Search** edit box, specify all or part of the name of the user or group to search for.
 - ii. From the **Search In** drop-down box, select either the Vault to search in or the name of the external directory to search for the user or group.
 - iii. From the **User Type** drop-down box, select the user type to search for.
 - iv. Click **Search**; a list of users and/or groups that meet these criteria is displayed.
8. Select the user or group to configure for this command, then click **Finish**; the command is displayed in the commands list with the user or group that is authorized to use it.
9. Add and ACL for all the commands in the shell's profile scripts:
 - i. Create a command group for all the commands that run automatically when the restricted shell starts. This includes commands that run as part of the shell's profile scripts, such as `.profile`, `.bashrc`, and can also include any other additional commands.
 - ii. Assign this command group to each user that will run the restricted shell.

Hiding Passwords during Recordings

You can hide passwords that are typed by the user in a pimsu session during recordings by setting the **PasswordPrompts** restriction. This is a regular expression restriction that is used to identify prompts that appear when passwords are typed by the user. When the system finds a match to this regular expression, it omits the line from the OPM session recording thus assures that passwords that were typed by the user will not be recorded.

General Configurations

The general parameters in the main configuration file determine how the OPM will work.

The following table lists the configuration parameters and when they will be applied to the OPM if you change them when the OPM is working.

Parameter	Action
MaxConcurrentRequests DisableExceptionHandling PIMSuCacheLevel PIMSuCacheFile ProviderCacheFolder PIMSuLocalRecordingFolder Port	These parameters will be applied after the OPM is restarted.
LogRetentionOnSizeMB LogRetentionOnTimeIntervalMinutes ShutdownTimeoutSec TcpTimeout CacheDebugLevels DebugLevels ProtocolDebugLevels PIMSuVaultAccessInterval PIMSuDebugLevels OfflineUpdateRetries DefaultDomain UnixUserFormatRegexp CommandTerminationGracePeriod TransactionBeforeCommandExecutionTimeout MaxConcurrentUploaders InteractiveLogonTimeout	These parameters will be applied after the parameters have been refreshed, either automatically according to the AutomaticParmsRefreshInterval parameter or manually with the appprvmgr utility.
PIMSuCacheRefreshInterval AutomaticParmsRefreshInterval OldLogsRetention OfflineUpdateInterval	If these parameters are changed from one positive number to another, they will be applied after the parameters have been refreshed, either automatically according to the AutomaticParmsRefreshInterval parameter or manually with the AppPrvMgr utility. If they are changed from zero (not active) to a positive number (active) or vice versa, they will be applied after the OPM is restarted.

Shared and OPM-specific Configuration

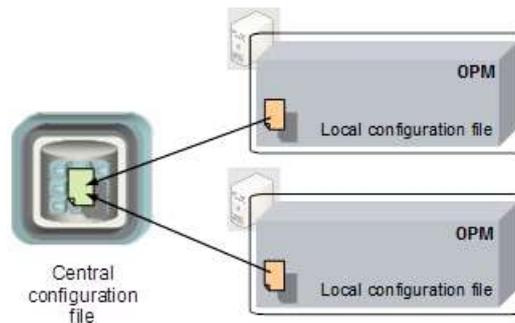
The OPM configuration parameters are stored in two different configuration files, a local file and a central file which is stored in the Vault.

The OPM is configured in the following configuration files:

- **Local configuration file** – This file, called **basic_opm.conf**, specifies the location of the main configuration file in the Vault. By default, the OPM configuration Safe is called AppProviderConf. During installation, this file is copied to the `/etc/opt/CARKaim/conf` folder.
- **Central configuration file** – This file, called **main_opm.conf**. **<platform>.<version>**, contains all the parameters that determine how the OPM will function. During installation, it is copied to the OPM configuration Safe, AppProviderConf.

As the central configuration file is stored in the Vault, several OPMs running on the same type of OS can use the same file. Alternatively, each OPM can use a configuration file that is specifically configured for it, as explained below:

- **Shared Configuration** – Several OPMs running on the same type of OS can access the same central configuration file, which determines how all the OPMs will function.



To Configure Shared Configuration

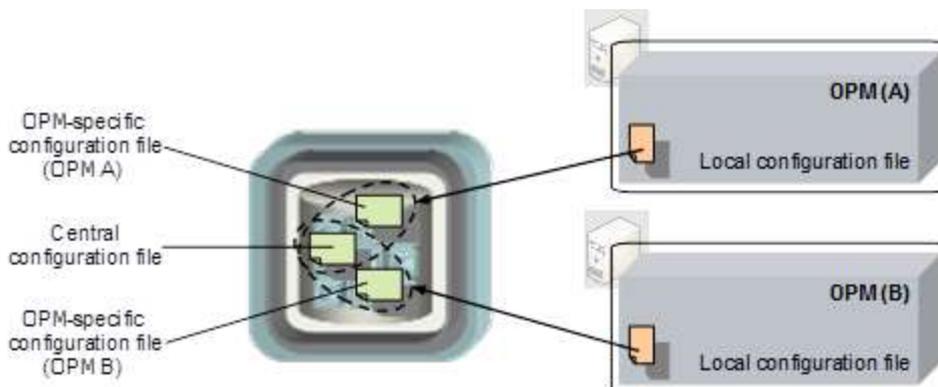
In order to configure Shared Configuration for two OPMs or more, you need to make sure that the local configuration files (**basic_opm.conf**) of all the related OPMs point to same main configuration file. This means that they specify the same Vault, the same configuration Safe, and the same configuration file.

1. Install the first OPM as described in the Privileged Account Security Installation Guide.
2. Install the second OPM. During installation, specify the following:
 - The same **Configuration Safe name** as you specified when you installed the first OPM in the previous step.
 - The same **Configuration file name** as you specified when you installed the first OPM in the previous step.

The installation will not override the **main configuration file** that was created when you installed the first OPM in the previous step.

For more information about specifying the above parameters, refer to the installation instructions.

3. Complete installation as described in the Privileged Account Security Installation Guide.
4. In the Password Vault, make sure the OPM user has the following authorizations on the Safe that contains the central configuration file:
 - List Files
 - Retrieve Files
 - Create Files
 - Update Files
 - Update File Properties
 - Create/Rename Folder
 - Initiate Password Management Operations
 - Initiate CPM Change with Manual Password
 - Rename Files
 - View Audit
 - View Owners
 - Use Password
 - Move Files/Folders
- **OPM-specific Configuration** – A specific configuration file can be created for individual OPMs to override parameter values in the central configuration file. In order to identify this file, its name must specify the OPM that it applies to and it must be stored in the same Safe and in the same folder as the central configuration file. All the parameters in this file override the parameters in the central parameter file, and any parameters that are not specified in the specific OPM file will be taken from the central configuration file.



To Create a OPM-specific Configuration File

1. In the Safe where the central OPM configuration file is stored, by default AppProviderConf, make a copy of the central configuration file.
2. Rename it using the following naming convention:
main_opm.conf.<platform>.<version>.<OPM's Vault username>
3. Open the new configuration file and specify values for the parameters that will be specific to the OPM. Make sure that you specify the section title for the parameters.
4. Delete all the parameters whose values will be taken from the central configuration file.
5. Save the configuration file, close it and store it in the Safe.

6. In the Password Vault, make sure the OPM user has the following authorizations on the Safe that contains the central configuration file:
 - List Files
 - Retrieve Files
 - Create Files
 - Update Files
 - Update File Properties
 - Create/Rename Folder
 - Initiate Password Management Operations
 - Initiate CPM Change with Manual Password
 - Rename Files
 - View Audit
 - View Owners
 - Use Password
 - Move Files/Folders

Each time the OPM service is restarted, the central configuration file and the OPM-specific configuration file are copied to a local cache on the OPM machine. This enables the OPM to operate when the Vault cannot be accessed.

Applying New Configurations when the Vault cannot be Accessed

When the Vault cannot be accessed and the central configuration file cannot be updated, you can create a local main configuration file on the OPM and specify the parameters that will override existing configuration parameters. You can view the current parameters with the `appprvmgr` utility.

When access to the Vault is restored, the local override file is synchronized with the OPM-specific configuration file. If a OPM-specific configuration file has not yet been created in the Vault, it will be created now.

Note: When the Vault is inaccessible, the `AutomaticParmsRefreshInterval` parameter is automatically set to 120. When the Vault becomes accessible again, this value is reset according to the parameter in the main configuration file.

Before creating a local override file, view the parameters that the OPM uses with the following command:

```
appprvmgr showparms 18924
```

For more information about the `appprvmgr` utility, refer to *Managing the OPM*, page 782.

To Create a Local Override File on an OPM

1. On the OPM machine, in the `/etc/opt/CARKaim/conf` folder, create a new text file called `local_overrides_opm.conf`.
2. Open the file and specify the parameters from the main configuration file that will override the existing parameters. Use the same structure as the original main configuration file in the Vault and specify each parameter under the relevant header, for example `[Main]` or `[PIMSuCache]`. For more information, refer to *OPM Main Configuration File* in the Privileged Account Security Reference Guide.

Note: All parameters except the `AutomaticParmsRefreshInterval` parameters can be specified in the local overrides configuration file.

3. Save the `local_overrides_opm.conf` file and close it.

Caching

The OPM can use a local cache to store information that will be supplied to users even when there is a network failure and the Vault cannot be accessed. Information in the cache will be stored encrypted and can only be accessed with the same authentication criteria that is required to retrieve information from the Vault.

The cache contains the following information:

- **Commands List** – This information includes privileged commands and command permissions, and is used to permit users to run privileged commands.
- **Access Control List** – This information includes details about Vault users and access control, and ensures that users can only perform the specific privileged commands that they are authorized for.

All the information in the cache is refreshed and synchronized with the Vault at periodic intervals. Any changes in commands and the passwords required to run them, or Vault configuration will be reflected in the cache after a refresh process, either automatically according to the **PIMSuCacheRefreshInterval** parameter in the main configuration file, or by forcing a refresh process with the **apprvmgr** utility. For more information about the **apprvmgr** utility, refer to *Managing the OPM*, page 782.

The OPM supports two levels of caching. These are specified in the central configuration file in the **PIMSuCacheLevel** parameter, as follows:

- **Memory Only** – The OPM will cache privileged commands and access control information in a memory cache. The OPM will not access the Vault for each request, but will rebuild the cache each time the machine or service restarts.
- **Persistent** – The OPM will cache command lists and access control information in a local secure storage as well as in memory. The name of the cache file is specified in the **PIMSuCacheFile** parameter. This makes the OPM completely independent of network performance and the Vault, and provides full availability to privileged commands.

How does it work?

The OPM maintains a cache that stores command lists and access control, and constantly updates it according to the parameters set in the configuration files. The OPM accesses the Vault regularly to retrieve updated details and to refresh the cache so that users can perform privileged commands according to accurate configurations and authorizations.

The OPM accesses the Vault to retrieve commands and access control details in the following scenarios:

- The time period defined in the **PIMSuVaultAccessInterval** has passed.
- The command and access control information is not stored in the cache. For example, after installing the OPM and configuring it, the command details and corresponding authorizations will be retrieved to the cache the first time a user requests it.
- When the OPM refreshes the cache.

During requests, the OPM does not remove existing commands or access control information from the cache; this information is only removed from the cache during a cache refresh process. This means that if you have changed user's permissions to prevent it from running a specific command, the details will remain in the cache where the user can still run it until the next cache refresh process, after which the user will not be able to run it, or until you initiate a manual refresh process using the `apprvmgr` utility.

Background Cache Refresh Process

A background process refreshes the cache with the information that the OPM has retrieved from the Vault at regular intervals, according to the **PIMSuCacheRefreshInterval** parameter in the main configuration file. As the background refresh process uses the OPM user to access the Vault, the OPM user must have permissions in the Safes where commands and the passwords required to run them reside in the Vault. For more information about building the Vault environment for the OPM, please refer to *Implementing the On-Demand Privileges Manager*, page 745.

The background process manages the following tasks:

- Updating the command and access control details in the cache and synchronizing it with updated details in the Vault.
- Verifying cache integrity to ensure that all the information is accurate and up-to-date.
- Removing any information from the cache in any of the following scenarios:
 - Information has been deleted from the Vault.
 - The command user or the OPM user is no longer authorized to access information that is stored in the cache. (The user does not have the required Safe authorizations to access passwords.)

The background process also changes the password of the OPM's user in the Vault and updates its user credentials files every 24 hours. This is not configurable.

The OPM retrieves information from the Vault and stores it in the cache, which is constantly updated by the background process. Whenever a user initiates a privileged session, they receive the most recent information.

The cache is refreshed periodically from the Vault, depending on the following parameters in the main configuration file:

- **PIMSuCacheRefreshInterval** – How frequently in seconds the cache will be refreshed from the Vault.
- **PIMSuVaultAccessInterval** – The number of seconds that information can be stored in the cache before it expires. If the Vault is inaccessible, commands and other information that reside in the cache will be returned. In order to force access to the Vault and bypass the cache, specify **PIMSuVaultAccessInterval=0**.
- **AutomaticParmsRefreshInterval** – How frequently in seconds the main configuration file on the OPM machine will be refreshed from the Vault. This parameter refreshes the configuration file specified in the `VaultParmsFile` parameter in the OPM basic configuration file, as well as the OPM-specific configuration file if it exists. For more information about OPM-specific configuration, refer to *Shared and OPM-specific Configuration*, page 770.

It is recommended to specify a value for the **PIMSuVaultAccessInterval** parameter that is higher than the value of the **PIMSuCacheRefreshInterval** parameter to prevent information from expiring in the cache, as the background refresh will always update the cache before records expire.

These parameters can only be modified in the central configuration file in the Vault.

If the Vault is inaccessible, the OPM will not try to access the Vault until the background cache refresh process succeeds in accessing the Vault, whether to refresh the cache or the main configuration file.

When the Vault is inaccessible, each OPM request will be handled directly by the cache until the periodic tasks, such as the background refresh task, detects that the connection to the Vault is active again. For more information, refer to *Issuing Privileged Commands When There Is No Direct Connection to the Vault*, page 777.

Implementing the Local Cache

Whenever the OPM receives a request to run a privileged command, it refers to the local cache for the command list instead of accessing the Vault directly. This provides users with constant access, regardless of network performance and Vault accessibility. The background process described above accesses the Vault regularly to refresh the local cache so that the OPM will always permit users to perform privileged commands according to accurate access control information.

Implementation Scenarios

The following table demonstrates four different implementation scenarios based on different cache related configurations:

Scenario	Description	Cache level	Sample VaultAccess Interval (sec)	Sample CacheRefresh Interval (sec)
"Cache Only"	The default OPM behavior. The OPM handles requests by accessing the cache, other than the first request by any user for which it accesses the Vault.	Memory/ Persistent	240	180
"Cache First"	The OPM handles requests by first accessing the cache, unless the cache refresh interval has expired and the cache needs to be refreshed, in which case, the OPM will access the Vault.	Memory/ Persistent	180	240
"Vault First"	The OPM immediately accesses the Vault to handle requests. If the OPM cannot access the Vault, it will refer to the cache.	Memory/ Persistent	0	180

Issuing Privileged Commands When There Is No Direct Connection to the Vault

The OPM enables users to run privileged commands during periods of time when the Vault is not available, while constantly tracking and monitoring all user activities. This facilitates uninterrupted workflows, increasing productivity and providing a focused and seamless business environment.

The OPM acts as a 'local server' that securely caches privileged commands and their properties and access control details, and provides immediate access to these privileged commands, independent of network performance. As the OPM periodically refreshes its cache from the Vault, ensuring that the information it stores is constantly up-to-date and accurate, this information is as reliable as possible when the Vault cannot be accessed directly. The OPM also generates recordings and audit logs that track activity with privileged commands so that there is complete accountability for each command request by every user, and monitoring logs that register OPM activity and status. These are maintained on the OPM when the Vault cannot be accessed, and uploaded to the Vault as soon as the connection between the OPM and Vault is restored.

The OPM cache offers the following features while the Vault is not available:

- **Secure cache mechanism** – Privileged commands can be stored encrypted in a secure cache on the OPM machine, together with access control information which the OPM requires to authenticate requests to run privileged commands. The OPM can permit these requests directly from the cache.

For more information about the different levels of caching and how to configure the OPM's local cache, refer to *Caching*, page 773.

- **Recordings** – Recordings of privileged sessions are maintained in the local recording folder on the OPM machine. Their metadata is also stored in the OPM cache, and is uploaded to the Vault as soon as there is an active connection.
- **Audit and Monitoring** – The Vault log, maintained in the OPM cache, tracks all account activity. This log contains information about the privileged commands audit, and their corresponding session recordings.

For more information about the OPM logs and configuring them, refer to *Auditing*, page 783.

Authenticating Users

When there is no active connection to the Vault, users can be authenticated by the information stored in the cache.

- **SSO authentication** – If the OPM is configured to authenticate users by SSO authentication, users will be authorized to perform privileged commands by the information in the cache as seamlessly as if the Vault were active.
- **Non-SSO authentication** – If the OPM is configured to authenticate users by a non-SSO authentication method, users will not be able to authenticate unless there is an active connection to the Vault or if the **AllowHostAuthenticationWhenVaultIsDown** parameter for their platform is set to **Yes**.

Updating OPM Activity in the Vault

By default, the OPM stores audit logs and recordings in the Vault immediately. However, when there is no active connection to the Vault, the OPM switches to an offline mode and updates a dedicated offline cache where it stores the logs and metadata for recordings until the Vault connection becomes active again and they can be uploaded into the Vault.

This offline cache is managed by a background process called the **Offline Manager**, which regularly checks the cache for new audit logs or recording files to upload to the Vault. The Offline Manager tries to access the Vault regularly according to the **OfflineUpdateInterval** and **OfflineUpdateRetries** parameters.

These parameters can be configured in the OPM main configuration file:

- **OfflineUpdateInterval** – Determines the interval in seconds between every cycle of the Offline Manager that is responsible for executing the offline operations saved in the offline cache. The default value is **1800** seconds.
- **OfflineUpdateRetries** – Determines the maximum number of retries by the Offline Manager in order to execute an offline operation that could not be performed in the Vault. The default value is **600** retries.

According to the value of the **OfflineUpdateInterval** parameter, the Offline Manager tries to perform all the offline operations that have been stored in the cache by the OPM. The Offline Manager tries to perform each offline operation until it reaches the maximum number of retries that is configured for each operation.

The combination of the above parameters ensures that, by default, the offline cache can store audit logs and recordings for approximately ten days when the Vault is not available.

Integrating with UNIX Centralized User Management Products

The OPM can be configured to support users that are configured in third party user management applications, such as NIS or AD-Bridge products. These users may be end users who use the pimsu utility or privileged users who execute privileged commands. The OPM supports both types of users and any combination between them as if they were local users.

Since privileged accounts (e.g. root) can be domain users, the corresponding accounts in the Vault for privileged users will be configured differently from local privileged accounts. For example, in a local account, the Address property specifies the local IP address, whereas in a UNIX domain user's account it can specify the domain name.

In order to be able to run privileged commands on behalf of a privileged account that is managed in domain, the domain name has to be specified in the account's Address property. As a result, when these users specify their name in the pimsu command, they must specify their full name, including the domain name, as shown below:

```
pimsu -u unixdom\user1
```

The following parameters in the OPM main configuration file enable the OPM to support Unix domain users:

- **UnixUserFormatRegexp** – This parameter specifies the following information:
 - **Domain Index** – The index of the submatch that describes the Unix domain.
 - **User Index** – The index of the submatch that describes the username.
 - **Regxp format** – A regular expression that describes the format of the user.

This is shown in the following example:

```
UnixUserFormatRegexp=1,2,(.*)\\(.*)
```

Using the above example, a user called unixdom\user1 will match the regular expression `(.*)\\(.*)` and will match two parameters. The first match is the domain name, **unixdom**, and the second match is the user name, **user1**.

- **DefaultDomain** – This parameter specifies the name of the domain that will be used as a possible value for the Address property of a privileged account, as shown in the following example:

```
DefaultDomain=unixdom
```

Using the above example, all users and accounts that are mapped transparently from a centrally managed UNIX domain will be allocated **unixdom** as the default domain in the account.

If this parameter is specified, it overrides the domain name that was obtained according to the UnixUserFormatRegexp parameter.

Configuration Guidelines:

NIS:

Configure the **DefaultDomain** parameter and set the NIS server address, so that the OPM will be able to locate the privileged accounts. This address should be the same as the address specified in the corresponding password object in the Vault.

AD-Bridge Product:

Configure the **UnixUserFormatRegexp** parameter so that the OPM will be able to locate the privileged accounts as well as the end users who execute privileged commands.

The follow table summarizes the configuration required for AD-Bridge or NIS integration:

Use case:	UNIX without Central User Management	NIS	AD-Bridge Product
Running pimsu for root user	pimsu (without -u) As the root account is defined locally, OPM will find it according to the local IP.	Pimsu (without -u) As the root account can be defined centrally in the NIS, OPM will find it according to the DefaultDomain parameter or the local IP.	Pimsu (without -u) As the root account is defined locally, OPM will find it according to the local IP.
Running pimsu with alternate privileged account	pimsu -u oracle OPM will search the oracle account according to the local IP.	Pimsu -u oracle As an oracle account can be defined centrally in the NIS, OPM will find it according to the DefaultDomain parameter or the local IP.	Pimsu -u DOMAIN\oracle OPM will find the oracle account according to the UnixUserFormatRegexp * parameters or the local IP.
Running pimsu when SSO configuration is enabled	The end user that will be authenticated is a logged on Unix user (e.g. Paul).	The end user that will be authenticated is a logged-on Unix user (e.g. Paul) which can be defined in the NIS.	The end user that will be authenticated is logged in by the AD-Bridge product. The format of this user is similar to Windows format, (e.g. MyDomain\Paul) The name will be parsed according to UnixUserFormatRegexp *. The corresponding Vault user is 'Paul', without the domain name.

Disaster Recovery

The OPM benefits from Disaster Recovery features which provide seamless productivity during a failover. For more information about Disaster Recovery in the Vault, refer to *CyberArk Disaster Recovery Vault*, page 885.

Transparent Failover

As soon as the Production Vault cannot be reached by the OPM, the failover process begins in the DR Vault transparently, and no human intervention is required.

The IP address of both the Vault and the DR Vault can be specified in the Vault.ini configuration file. When the OPM cannot reach the Vault specified by the first IP address, it transfers automatically to the Vault specified by the second IP address, which is the DR Vault.

To Configure Transparent Failover

1. In the Vault.ini file, in the **Address** parameter, specify the IP addresses of the Vault and the DR Vault, separated by commas, as shown in the following example:

```
Address=1.1.1.102,1.1.1.232
```

The above example indicates that the IP address of the Production Vault is 1.1.1.102 and the IP address of the DR Vault is 1.1.1.232.

2. Add the **SwitchVaultAddressTimeOut** parameter.

This parameter specifies the number of seconds that the OPM will try to access additional Vault IP addresses after the initial timeout to the current Vault, specified in the **Timeout** parameter, expires.

If this parameter is not added, the default value of three seconds will be applied.

3. Save the Vault.ini file and close it.

Replicating OPM Users' Passwords

As each OPM user requires a credential file to authenticate to the Vault, it is essential that the credential files in the DR Vault are always identical to those on the Production Vault. At regular intervals, the OPM automatically initiates a password change in the Vault and in the corresponding credential file for the OPM. In order for the OPM user in the DR Vault to access the Vault and continue working seamlessly in a Disaster Recovery situation, the user's new credentials must be replicated to the DR Vault whenever they are changed.

This is configured by the following parameter in the CreateCredFile utility:

- **DisableSyncPasswordToDR** – Whether or not passwords in user credential files will be replicated to all DR sites before they are replaced. The default value of this parameter is 'No', which makes sure that user credential files on all DR sites (if they exist) are synchronized with the Production Vault and that users will be able to continue working with the Vault seamlessly after a failover. If this parameter is changed to 'Yes', passwords will be replaced in credential files regardless of whether or not they have been replicated to all DR sites.

Accessing the Password Vault

The OPM user requires a user credential file to access information in the Password Vault and retrieve it so that the requesting user can issue a privileged command. The following parameters in the basic configuration file indicate the Password Vault where accounts are stored as well as the location of the OPM's user credential file.

- **VaultFile** – The full pathname of the Vault.ini file from where accounts will be retrieved.
- **CredFile** – The full pathname of the OPM's credential file used to access the Password Vault.

Managing the OPM

The **appprvmgr** utility enables you to manage the OPM while it is still working, eliminating downtime and restarts during maintenance procedures. This utility enables you to do the following tasks:

- **View parameters** – You can view the parameters that determine how the OPM currently functions.
- **Refresh parameters** – You can refresh the OPM configuration parameters.
- **Refresh the local cache** – You can refresh the OPM's local cache.

During installation, this utility is copied to the /opt/CARKaim/bin folder on the OPM machine.

It has the following syntax:

```
appprvmgr.exe <command> [command parameters] [/port <port>] [/?]
```

This utility has the following syntax:

Command	Specifies
ShowParms	Displays the configuration parameters that determine how the OPM currently functions, regardless of the configuration files where these parameters are specified (central, OPM-specific, or local overrides configuration files). These parameters do not include the platform configuration parameters.
RefreshParms	Refreshes the configuration parameters that determine how the OPM currently functions after any changes have been made in any of the configuration files.
RefreshOfflineCache	Forces the OPM to refresh its offline cache by running any pending operations.
RefreshCache	Forces an immediate cache refresh. This command can be used to refresh the AIM and/or the OPM cache. If neither of the following command parameters are specified, both caches will be refreshed.
AIM	The AIM cache will be refreshed.
OPM	The OPM cache will be refreshed.
Port	The port that is used to access the OPM.
--?	Lists the available options.

If the OPM completes the command with warnings or errors, a relevant message will appear. Check the OPMConsole or the OPMTrace log files for more information.

Note: If the trace debug levels are set to a low level or to zero, set the debug levels to a higher level and then rerun the utility.

The following example shows the command that can be issued to display the configuration parameters that are applied to the OPM:

```
>appprvmgr showparms
```

The following example shows the command that can be issued to update the OPM cache immediately:

```
>appprvmgr refreshcache OPM
```

Auditing

Every request issued at a command line to run a privileged command is recorded in the Vault audit log. These activities can be viewed in the Activities Log under the following activity groups:

- Privileged Accounts Access Activities
- Privileged Accounts Management Activities

For more information, refer to *The Privileged Account Security Solution Reports*, page 340.

In addition, the entire privileged session that is run after the privileged command is authenticated is recorded by the OPM and is uploaded into the Vault where it can be accessed by authorized users. The location of the recording files is specified by the **SessionRecorderSafe** parameter in the On-Demand Privileges Manager parameters in the platform that is associated with the account used to initiate the privileged session.

Monitoring

In order to monitor OPM activity and status, the OPM creates the following log files:

- **OPMConsole.log** – This file contains informational messages about the OPM, such as ‘Server is starting’ and ‘Server is shutting down’. This log is meant for system administrators who monitor the status of the OPM. Errors that refer to OPM function and user authentication are included in this log. All the messages that are written in the console log file are also written in the UNIX system log. For more information, refer to *Monitoring the Unix System Log*, page 786.
- **OPMTrace.log** – This file contains errors and trace messages. The types of messages that are included depend on the debug levels specified in the main configuration file in the following parameters:
 - **DebugLevels** – Sets the debug level of the OPM. You can set several values, separated by commas.

Debug level	Indicates
0	No messages will be written to the trace log. This is the default debug level.
1	OPM errors will be written to the trace log.
2	OPM trace messages will be written to the trace log.
3	OPM CASOS errors will be written to the trace log.
4	OPM CASOS activities and trace messages will be written to the trace log.
5	OPM background refresh trace messages will be written to the trace log.

- **ProtocolDebugLevels** – Sets the debug level of the protocol layer. You can set several values, separated by commas.

Debug level	Indicates
0	No messages will be written to the trace log. This is the default debug level.
1	Protocol errors will be written to the trace log.
2	Protocol trace messages will be written to the trace log.

- **CacheDebugLevels** – Sets the debug level of the cache. You can set several values, separated by commas.

Debug level	Indicates
0	No messages will be written to the trace log. This is the default debug level.
1	Cache errors will be written to the trace log.
2	Cache trace messages will be written to the trace log.

- **PIMSuDebugLevels** – Sets the debug level of the OPM activity. You can set several values, separated by commas.

Debug level	Indicates
0	No messages will be written to the trace log. This is the default debug level.
1	OPM errors will be written to the trace log. This has the same effect as setting the DebugLevels parameter to 1.
2	OPM command transmission trace messages will be written to the trace log.
3	OPM background refresh trace messages will be written to the trace log.
4	OPM execute command service trace messages will be written to the trace log.
5	The OPM objects created during execute command service trace messages will be written to the trace log.
6	All commands and activities performed using extended protection, prevent shell escape and restricted shell restrictions will be written in the preload.log.<pid> file in the /opt/CARKaim/lib folder. Do not activate this trace level without consulting with your CyberArk support representative.

Monitoring the CyberArk OPM Service

The OPM service can be monitored by standard enterprise monitoring tools that enable you to track its status. This enables you to know when the OPM is active and responsive to command elevation requests, and when a problem occurs that prevents the OPM from functioning properly and which will cause it to stop.

Monitoring the Unix System Log

OPM messages can be viewed in the Unix System Log. They are identified by their source name, which is “Cyber-Ark OPM”. These messages include status messages, such as ‘Server is starting’ and ‘Server is shutting down’, as well as errors that occur because the OPM cannot supply the password due to authorization or configuration issues.

The Unix System Log can be configured to write OPM messages to the messages file in addition to any other daemon facility messages. The following table displays the options for the OPM:

Facility	Priority
daemon	Info
	Err
	Warning

Configuring the System Log for the OPM

1. In the system log configuration file, `syslog.conf`, add the following line:

```
<facility>.<priority>[;<facility>.<priority>;...] <file path>
```

2. Restart the **syslog** service.

The following example will write all daemon facility messages to the `/var/log/messages` file. This includes information, errors and warnings.

```
Daemon.err;daemon.warning;daemon.info /var/log/messages
```

Configuring Auditing and Monitoring Log Files

The following parameter in the basic configuration file specifies the location of the OPM log files:

- **LogsFolder** – The full pathname of the folder where the OPM will store the local logs file.

The following parameters in the main configuration file specify when the log file will be archived in the Vault

- **LogRetentionOnSizeMB** – The size in MB of the log files when they are moved to the ‘old’ subfolder of the Logs folder. New log files will be started automatically.
- **LogRetentionOnTimeIntervalMinutes** – The number of minutes after which the log files are moved to the ‘old’ subfolder of the Logs folder. New log files will be started automatically.

The following parameter in the main configuration file specifies when old log files will be deleted.

- **OldLogsRetention** – The number of days that trace and console log files will be saved, after which they will be deleted. Audit log files will not be deleted. The OPM will automatically search every hour for log files to delete; this cannot be configured.

To prevent old log files from being deleted, specify 0 (zero).

Administrating the On-Demand Privileges Manager

This section describes several basic administration procedures that are required to manage the OPM and commands.

Renaming the OPM User

1. Stop the CyberArk On-Demand Privileges Manager service (opmsrv).
2. In the Password Vault, rename the OPM user. Specify any name that meets your enterprise requirements.
3. Specify a new password for the OPM user.
4. On the OPM machine, create a new user credential file for the OPM user with the CreateCredFile utility. Use the same file name, but specify the OPM's new username and password. For more information, refer to *Appendix B: Creating User Credential Files*, page 956.
5. If the OPM is configured to maintain a PIMSu persistent cache, delete the cache file. The location of the cache file is specified in the PIMSuCacheFile parameter in the main_OPM configuration file.
6. Delete all the general cache files that were created by the OPM. The location of these files is specified in the CacheFolder parameter in the main_opm configuration file.
7. If the OPM uses a OPM-specific configuration file, rename the file suffix to specify the new OPM user name.
8. Start the CyberArk On-Demand Privileges Manager service (opmsrv).
9. Make sure that the OPMuser has been renamed successfully:
 - i. Open the OPMConsole.log, and make sure that the first lines contain the following messages:

```
APPAP035I Application Password Provider [Provider_Name] on machine
[IPAddress] version [Version_number] is up [PROVIDER_MODE]
APPAP032I Main parameters file [ConfFile] was loaded successfully
```

These messages indicate that the OPM started successfully.
 - ii. From a command line, using the pimsu utility, run a privileged command. If the OPM has been renamed and configured correctly, the command will be performed successfully.

Relocating the OPM to a Different Machine

The following steps describe how to move the OPM and its cache to a different machine, as follows:

1. Install the OPM on the new machine, as described in the Privileged Account Security Installation Guide.
2. Specify the same OPM user name that was used in the previous machine:

In Solaris:

- At the OPM user name prompt, specify the same OPM user name that was used in the previous machine.

In Linux/AIX:

- In the OPM user name field in the aimparms file, specify the same OPM user name that was used in the previous machine.
 - If you specify a different OPM user, the OPM cannot use the caches that were used on the previous machine. In this situation, the OPM must be installed as described in the Privileged Account Security Installation Guide, and the following instructions are not relevant.
3. If the OPM on the previous machine was configured to use a persistent cache, copy the cache file to the location on the new machine that is specified in the PIMSuCacheFile parameter in the main_opm configuration file.
 4. Copy all the general cache files that were created by the OPM to the location on the new machine that is specified in the CacheFolder parameter in the main_opm configuration file.
 5. Copy the OPM user's credential file from the previous machine to the location on the new machine that is specified in the CredFile parameter in the basic_opm configuration file.

As the user credential file specifies security restrictions, such as the OPM machine's IP, if any of the specified restrictions have changed, a new user credential file will be required. For more information, refer to *Appendix B: Creating User Credential Files*, page 956.

6. Make sure that you have reconfigured the OPM environment correctly:
 - i. Start the CyberArk On-Demand Privileges Manager service (opmsrv).
 - ii. Open the OPMConsole.log, and make sure that the first lines contain the following messages:


```
APPAP035I Application Password Provider [Provider_Name] on machine
[IPAddress] version [Version_number] is up [PROVIDER_MODE]
APPAP032I Main parameters file [ConfFile] was loaded
successfully
```

 These messages indicate that the OPM started successfully.
 - iii. From a command line, using the pimsu utility, run a privileged command. If the OPM has been moved and configured correctly, the command will be performed successfully.

Clearing the Persistent Cache

1. Stop the CyberArk On-Demand Privileges Manager service (opmsrv).
2. Delete the cache file. The location of the cache file is specified in the PIMSuCacheFile parameter in the main_opm configuration file.
3. Start the CyberArk On-Demand Privileges Manager service (opmsrv).

Changing a Configuration Safe

1. In the basic configuration file, specify the following:
 - The location in the Vault of the main configuration file:
 - **AppProviderParmsSafe** – The name of the new Safe where the main configuration file is stored.
 - **AppProviderVaultParmsFolder** – The name of the folder in the configuration Safe where the main configuration file is stored.
 - **VaultParmsFile** – The name of the main configuration file.
 - The local copy of the shared configuration file:
 - **LocalParmsFileFolder** – The folder on the application machine where the main configuration file will be stored.
2. In the Password Vault, make sure that the OPM user has the following authorizations on the new Safe that contains the main configuration file:
 - Use accounts
 - Retrieve accounts
 - List accounts
 - Add accounts
 - Update password value
 - Update password properties
 - Rename accounts
 - Create/Rename folders

Service Administration for Solaris

The OPM service will be started automatically when the Solaris machine is initialized in run levels 2 or 3. You can start and stop the service manually as described below.

Only one instance of the OPM can run on any machine.

Starting the Service

To start the OPM service, use the following command:

```
/etc/init.d/opmsrv start
```

Note: You must have root permissions to use this command.

Stopping the Service

To stop the OPM service, use the following command:

```
/etc/init.d/opmsrv stop
```

Note: You must have root permissions to use this command.

Checking the Service Status

To check the OPM service status, use the following command:

```
/etc/init.d/opmsrv status
```

Service Administration for Linux

The service will be started automatically when the Linux machine is initialized in run levels 2 – 5. You can start and stop the service manually as described below.

Only one instance of the OPM can run on any machine.

Starting the Service

To start the service, use the following command:

```
/etc/rc.d/init.d/opmsrv start
```

Note: You must have root permissions to use this command.

Stopping the Service

To stop the service, use the following command:

```
/etc/rc.d/init.d/opmsrv stop
```

Note: You must have root permissions to use this command.

Checking the Service Status

To check the service status, use the following command:

```
/etc/rc.d/init.d/opmsrv status
```

Service Administration for AIX

The service will be started automatically when the AIX machine is initialized in run levels 2 – 9. You can start and stop the service manually as described below.

Only one instance of the OPM can run on any machine.

Starting the Service

To start the service, use the following command:

```
/etc/rc.d/init.d/opmsrv start
```

Note: You must have root permissions to use this command.

Stopping the Service

To stop the service before rebooting or shutting down the system, do the following:

1. Add the **/etc/rc.shutdown** file, if it doesn't exist already.
2. In the above file, add the following line: **/etc/rc.d/init.d/aimprv stop**

Note: The **/etc/rc.shutdown** file will be called when you run the shutdown command.

Password Upload Utility

The Password Upload utility works with the CyberArk Password Vault to create password objects from a passwords list and store them in the Vault. This makes the Vault implementation process quicker and more automatic.

This chapter introduces you to the Password Upload utility and guides you through setup and implementation. It includes the following topics:

- Implementing the Password Upload Utility
- Running the Password Upload Utility

Implementing the Password Upload Utility

The Password Upload utility uploads multiple password objects to the Password Vault, making the Vault implementation process quicker and more automatic. This utility works by uploading passwords and their properties into the Password Vault from a pre-prepared file, creating the required environment, when necessary. It is run from a command line whenever a password upload is required.

Creating the Vault Environment Automatically

The Password Upload utility initiates the Vault environment required to store passwords in the Safe and start working with them. This includes creating new Safes, adding the CPM user as a Safe owner, and sharing the Safe with the Password Vault Web Access.

Creating New Safes

The Password Upload utility uses Template Safes to create Safes automatically with the properties that are specified in the Template Safes. You can create different types of Template Safes, depending on your requirements. When the utility uploads passwords into the Vault, if the specified Safe doesn't exist, the utility will create a new Safe based on the Template Safe that is specified in the password file. If a Template Safe is not specified, a new Safe will be created, based on the default Template Safe that is specified in the utility configuration file.

In order to create a new Safe based on a Template Safe, the user running the utility requires the following authorizations in the Vault:

- The 'Add Safes' user authorization
- Ownership of the Template Safe with at least one authorization

Adding the CPM User as a Safe Owner

The Password Upload utility adds a CPM user automatically to new and existing Safes to which it uploads passwords, with the following authorizations:

- View Audit
- View Safe Members
- Retrieve accounts
- List accounts
- Add accounts
- Update password value
- Update password properties
- Access Safe without Confirmation
- Unlock accounts (dependent on the parameters specified in the configuration file)
- Manage Safe (dependent on the parameters specified in the configuration file)

The name of the CPM user is specified in the password file.

In order to add the CPM user to existing Safes, the user running the utility requires the above authorizations in the Safe as well as the following authorization:

- Manage Safe Members

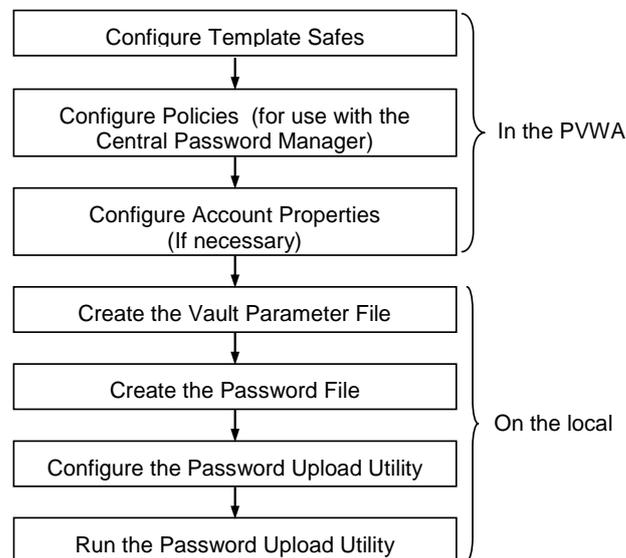
Sharing the Safe with the Password Vault Web Access

The Password Upload utility automatically shares new and existing Safes to which it uploads passwords with the Password Vault Web Access gateway account whose name is specified in the utility configuration file. This enables users to access passwords through the Password Vault Web Access as soon as they have been uploaded to the Safe.

In order to share existing Safes with the gateway account, the user running the utility requires the following authorization in the Safe:

- Manage Safe

The following diagram shows the procedure to follow in order to enable the utility to upload password objects successfully.



In the PVWA:

1. Create the Safes where the passwords will be stored. For more information, refer to *Adding Safes in the PVWA*, page 65.
2. If this Safe will be used as a Template Safe:
 - If this Safe will be used as a Template Safe for all the new Safes that will be created automatically when the utility uploads the password list, in the utility configuration file, in the **DefaultTemplateSafe** parameter, specify the default template Safe that will be used to create new Safes.
 - If different Template Safes will be used for different password files, specify the name of the relevant Template Safe in the password file.

Note: The name of the Template Safe only needs to be specified the first time the non-existent Safe appears.

For more information about Template Safes, refer to *Creating New Safes*, page 793.

Note: This utility only supports Safe, folder, and file names in English. Make sure that all Safe and folder names are in English.

3. If you created Safes manually, give the user that will run the utility Safe ownership of all the Safes specified in the password file, with the following authorizations:
 - Add accounts
 - Update password properties
 - Update password values
 - Access Safe without confirmation – In existing Safes that require confirmation from authorized users before they can be accessed (dual control)
4. In the ADMINISTRATION, in Platform Management, configure the target account platform that will determine the type of password that is allowed and how frequently it must be changed. Each platform has a unique platform name which will be specified in the password file for each password object.

For more information about platforms, refer to *Adding New Platforms*, page 110.

On the machine where the utility is installed:

5. In the utility installation folder, open the Vault parameter file and specify the parameters of the Vault into which the password objects will be uploaded. For more information, refer to *Vault Parameter File* in the Privileged Account Security Reference Guide.
6. If you want to run the utility automatically, so that you do not have to supply the user name and password, create a user authentication file for the user who will run the utility. For more information, refer to *Appendix B: Creating User Credential Files*, page 956.
7. In the utility installation folder, open the password file and specify the password objects and their properties to upload to the Vault, then save the file in Comma Separated Values (CSV) format. For more information, refer to *Creating the Password File*, page 796.
8. In the utility installation folder, open the configuration file and do the following:
 - Specify the parameters that will enable the utility to upload the password file to the Vault.

Set the following parameter:

```
CPMUserAdminRights=yes
```

For more information, refer to *Configuring the Password Upload Utility*, page 799.

9. At a command line prompt, run the Password Upload utility.

The following example would run the utility according to a configuration file called Conf.ini. As no path is specified, the file is stored in the utility installation folder.

```
> PasswordUpload Conf.ini
```

For more information, refer to *Running the Password Upload Utility*, page 801.

Creating the Password File

Password parameters that will be uploaded to the Vault are stored in a text file as Comma Separated Values (CSV). The first line in the file defines the names of the password properties as specified in the Password Vault. Every other line represents a single password object and its property values, according to the properties specified in the first line.

Note: This utility only supports Safe, folder, and file names in English. Make sure that the filename is in English and that all Safe, folder, and file names match the exact case of those in the Vault.

The following password properties are required for every password object that will be uploaded to the Vault:

Parameter	Description
Password_name	The name of the Password object.
Safe	The name of the Safe where the password object will be stored.
Folder	The name of the folder where the password object will be stored.
Password	If the password object is new, a password must be specified. To upload a new password object with a blank password, specify "NO_VALUE".

A password property, whose value is not specified in the password values, will not be specified in the password object when it is uploaded to the Vault.

To Save a Password File in Excel

You can create a password file in Excel and save it in CSV format so that it can be uploaded to the Vault. Each column in the Excel file represents a different password property.

1. In the utility installation folder, open the sample password file and specify the values of the passwords that will be uploaded to the Password Vault when the utility is run.

Note: Do not change the order of the first 6 columns in the password file (Password_name, TemplateSafe, CPMUser, Safe, Folder, Password).

2. Save the file in CSV format.

Specifying Passwords in the Password File

Passwords that will be changed automatically by the CPM require the following additional password properties.

Parameter	Description
Platform name	The Platform Name parameter of the platform that will be applied to this password, and is specified in the platform.
UserName	The name of the user on the remote machine who this password belongs to.
Address	The address of the Vault where the password will be changed (IP or DNS).

Other password parameters are optional. For a complete list of password object properties that are created when the Password Vault is installed, refer to *Appendix A: Account Properties*, page 924.

Adding Password Properties without a Value

Some password properties do not require a value, but can be added to the password object when it is uploaded to the Vault.

- In the password property value, specify **NO_VALUE**; the password property will be added to the password object, but a value will not be assigned to it.

Deleting Password Properties

A password property can be deleted from an existing password object.

- In the password property value, specify **DELETE**; when the password object is uploaded to the Vault, the password property will be deleted from the password object.

Updating Existing Password Objects

Both passwords and properties in existing password objects can be updated through the password file.

- In the password file, specify the new value for the password or the password property to update. Password or property values that will not be changed should be left empty; when the utility uploads the password and password properties to the Vault, existing password objects will be updated.

A configuration parameter in the utility configuration file must specify that properties in existing password objects can be updated. For more information, refer to *Configuring the Password Upload Utility*, page 799.

Adding Comments to the Password File

Lines that are marked as comments will not be uploaded to the Password Vault.

- To mark a line as a comment, at the beginning of the line, type hash (#).

Example:

The following sample password file displays a header line with two passwords to upload to the Password Vault.

```

Password_name,TemplateSafe,CPMUser, Safe, Folder, Password, DeviceType, PolicyID,
UserName, Address, CPMDisabled, ResetImmediately
Operating System-UnixSSH-1.1.1.250-Root, ExclusivePasswordsTemplate,
PasswordManager, UnixPasswords, Root, asdf, Operating System, UnixSSH, Root,
1.1.1.250,, NO_VALUE
Operating System-Windows-1.1.1.227-Administrator,, , WindowsPasswords,
Root\Domains, 1234, Operating System, Windows, Administrator, 1.1.1.227, NO_VALUE

```

Password 1:

The first password object that will be uploaded is for use on an **Operating System** device and will be managed by the **UnixSSH** platform. This password is called **Operating System-UnixSSH-1.1.1.250-Root** and will be stored in the **UnixPasswords** Safe in the **Root** folder. If this Safe does not exist, it will be created according to the **ExclusivePasswordsTemplate** Safe. The CPM user called **PasswordManager** will be added to the Safe with all the authorizations required to enable him to manage the passwords within. This Safe will be shared with the gateway account specified in the 'GWAccounts' parameter in the configuration parameter file. The password is **asdf**. This password is intended for the **Root** user on the machine whose host IP is **1.1.1.250**. The **CPMDisabled** property is not specified and therefore the password will be managed by the CPM. The **ResetImmediately** value has not been specified, but the property will be specified in the password object, and the password will be changed by the CPM during the next cycle.

Password 2:

The second password object that will be uploaded is for use on an **Operating System** device and will be managed by the **Windows** platform. This password is called **Operating System-Windows-1.1.1.227-Administrator** and will be stored in the **WindowsPasswords** Safe in the **Root\Domains** folder. If this Safe does not exist, it will be created according to the default Template Safe specified in the 'DefaultTemplateSafe' parameter in the configuration parameter file. As no CPM user is specified, the CPM user will not be added as a user to the Safe. The password is **1234**. This password is intended for the **Administrator** user on the machine whose host IP is **1.1.1.227**. The **CPMDisabled** property value has not been specified, but the property will be specified in the password object, and the password will not be changed by the CPM until this property is removed. The **ResetImmediately** property has not been specified and will not be added to the password object.

Configuring the Password Upload Utility

The Password Upload utility is configured through a parameter file that contains references to parameter files and to specific parameters that determine the utility's functionality.

A sample configuration file is included in the package that contains the Password Upload utility.

For a complete list of the parameters in the Password Upload utility's configuration file, refer to the Privileged Account Security Reference Guide.

Updating Existing Password Objects

The Password Upload utility can update existing password object properties and passwords in the Vault according to the properties specified in the CSV file.

- In the utility configuration file, specify the following parameter:

```
UpdateIfExists=Yes
```

If `UpdateIfExists=No`, neither passwords nor password properties can be updated by the utility.

Creating Missing Folders in the Vault

If the password object properties in the password file specify a folder in the Vault that does not exist, the Password Upload utility can create the new folder and create the password object in that folder.

- In the utility configuration file, specify the following parameter:

```
CreateMissingFolders=Yes
```

Managing Errors

If an error occurs when the Password Upload utility is uploading a password object from the password file, the utility can either abort the upload process or skip to the next password object to upload in the password file. In both cases, an error will be written to the error log.

- To abort the upload process if a password object cannot be uploaded, specify the following parameter:

```
StopOnError=Yes
```

- To continue uploading the next password object in the password file, specify the following parameter:

```
StopOnError=No
```

Example:

Below is a sample configuration file:

```
#-----
# Mandatory parameters
#-----
Os=windows
VaultFile=vault.ini
PasswordFile=passwords.csv
DefaultTemplateSafe="Default Template"
CPMUserAdminRights=yes
AllowFullImpersonationSharing=no
GWAccounts=PVWAGWAccounts

#-----
# Optional parameters
#-----
SessionId=1
CredFile=user.ini
LogFile=UploadPasswords.log
ErrorLogFile=ErrorLog.log
UpdateIfExists=yes
StopOnError=no
CreateMissingFolders=yes
VerboseMode=yes
DebugMode=no
```

The above sample parameter file specifies the name of the Vault parameter file, **vault.ini**. As a pathname is not specified, the utility will look for it in the same folder that the utility is running from. The list of password objects to upload are stored in the **passwords.csv** file in the same folder as well.

If the Safe specified in the CSV file does not exist, and no specific Template Safe is defined, the Safe called **Default Template** will be used as the Template Safe. If the CPM user is specified in the CSV file, it will be added to the new Safe with the **Manage Safe** authorization, which will enable him to manage Safes in Exclusive Passwords mode. The new Safe will be shared by the PVWAGWAccounts gateway accounts group and impersonation will be set to **Enable access to impersonated users with additional Server authentication**. As a result, users who log onto the Vault through the Password Vault Web Access will be required to supply a username and password (Vault, Radius or LDAP authentication).

The utility will allocate Session ID number '1' to this session.

The **user.ini** credentials file is specified, indicating that the user running the utility will be able to log onto the Vault automatically, without any human intervention.

All activities will be saved in a log file called **UploadPasswords.log**, while error messages will be saved in a file called **ErrorLog.log**. Neither of these parameters specifies a pathname, indicating that they will also be saved in the same folder as the utility.

The utility will replace existing passwords or password object properties with passwords or password properties specified in the password file that is being uploaded. If, for any reason, an error occurs and a password object cannot be uploaded to the Vault, the utility will write an error message in the ErrorLog.log file and continue uploading the next password object in the password file. If the password object properties in the password file specifies a folder in the Vault that does not exist, the utility will create that folder and store the new password object in it.

The `VerboseMode` parameter determines that when this utility runs, the user will be able to see how the upload process develops by viewing constant messages, confirmations, and errors on the screen.

When this configuration file is used to run the utility, the debug mode will not be activated.

Running the Password Upload Utility

The Password Upload utility is a command line utility that has the following usage:

```
PasswordUpload <Configuration Filename>
```

Parameter	Description
Configuration Filename	The name of the configuration file that contains references to the password file to upload, and parameters that determine the utility functionality. This configuration file is described in detail in <i>Configuring the Password Upload Utility</i> , page 799.

Before running the utility, make sure that the user who will run it is an owner of existing and Template Safes that are specified in the password file with the appropriate authorizations. For more information, refer to *Creating the Vault Environment Automatically*, page 793.

To Run the Password Upload Utility

- At the command line prompt, run the **PasswordUpload** utility.
 - If you do not specify a user credentials file, you will be prompted for the user name and authentication of the Vault user running the utility.
 - If you specify the user credentials file, you will not be prompted for user authentication. For more information about creating user credentials files, refer to *Appendix B: Creating User Credential Files*, page 956.

Event Notification Engine

The Event Notification Engine (ENE) sends email notifications about Privileged Account Security solution activities automatically to predefined users. It is installed automatically as part of the Vault server installation as a service.

This chapter introduces you to the ENE and guides you through the following processes:

- Enabling the ENE
- Configuring the ENE
- Logging

Enabling the ENE

After installing the Vault, the ENE must be enabled so that you will be able to receive email notifications about the Vault activities.

Note: After the ENE has been configured, the ENE setup wizard will only be enabled if the SMTP address is set to 1.1.1.1. To rerun the ENE setup wizard, in the Notification Settings page, set the SMTP address to 1.1.1.1 then re-invoke the ENE setup wizard.

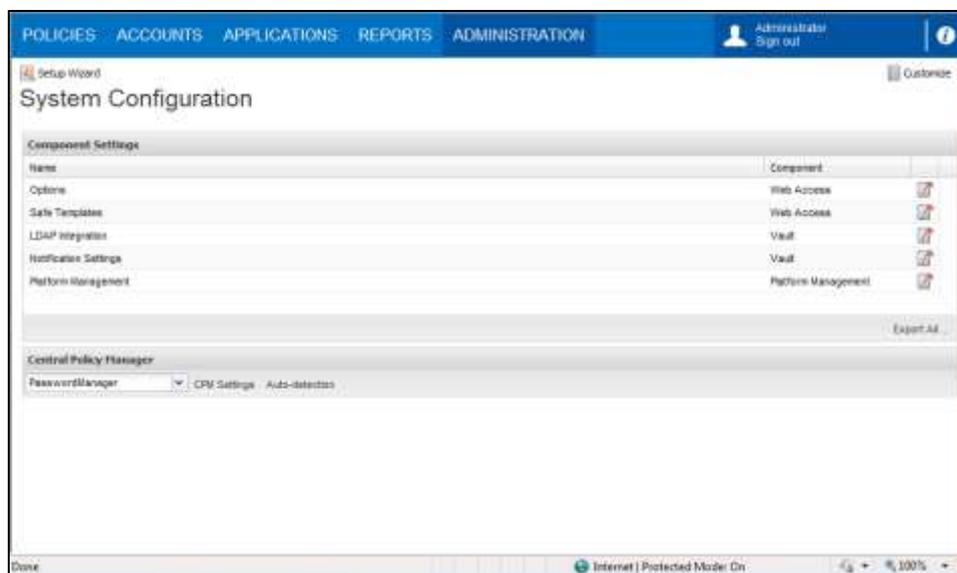
Before Enabling the ENE

1. Log onto the PrivateArk Administrative Client as an administrator user.
2. Make sure that the business email address of the user who will issue ENE notifications is specified in their user properties. This user must belong to the **Vault Admins** group. By default, this is the **Administrator** user.

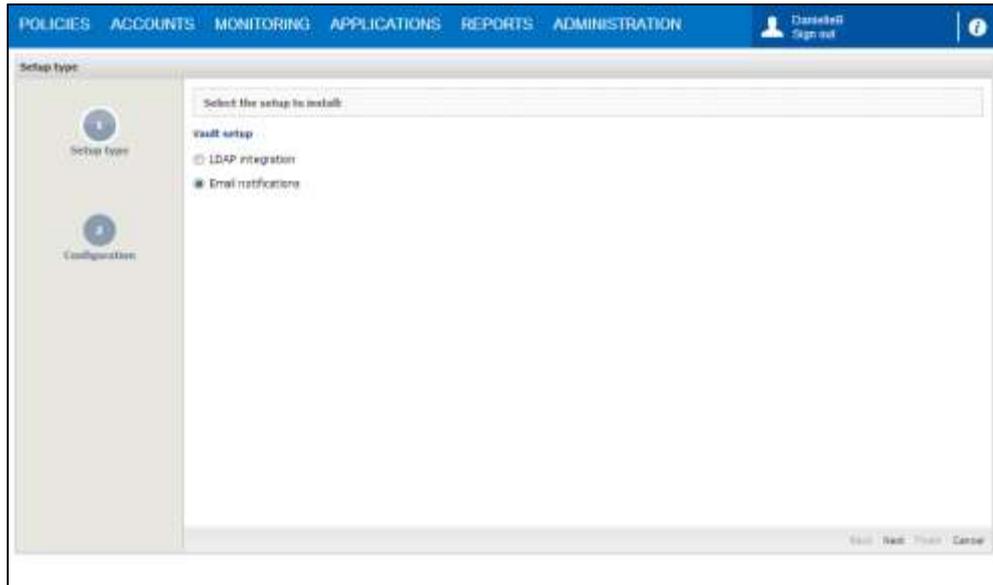
To Enable the ENE

The ENE is installed as part of the Vault server installation as a service called **Cyber-Ark Event Notification Engine**. After Vault server installation or upgrade, do the following to enable the service:

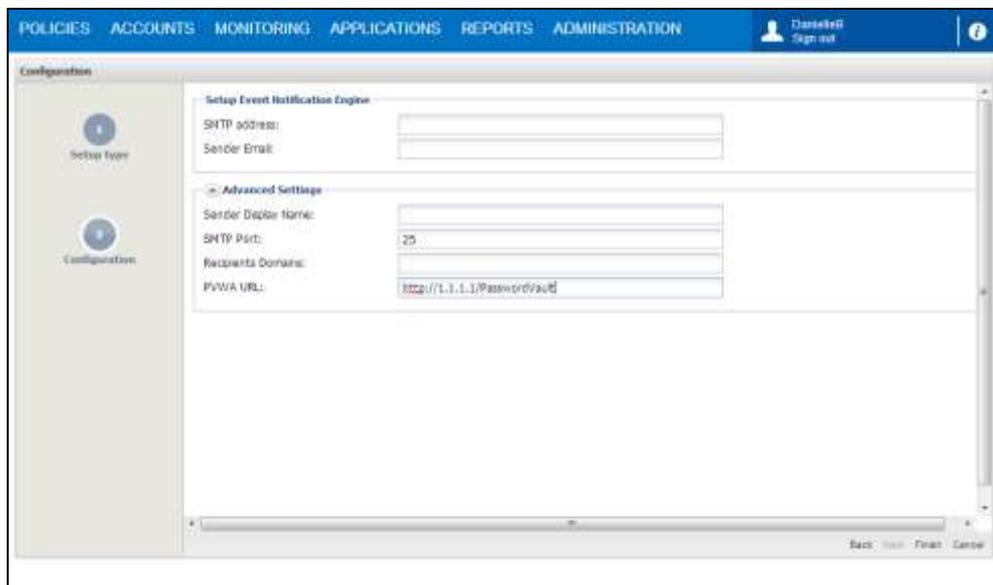
1. Log onto the PWWA as an administrator user. Make sure that this user belongs to the **VaultAdmins** group so that you have the required permissions to enable ENE notifications.
2. Enable the Event Notification Engine:
 - i. In the System Configuration page, click **Setup Wizard**.



The Setup Configuration wizard displays the Vault setup page.

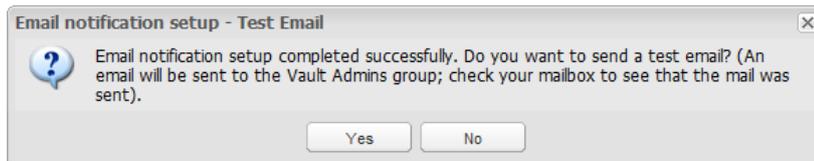


- ii. Select **Email notifications**, then click **Next**; the Configuration page appears.



- iii. In the **Setup Event Notification Engine** area, specify the following details:
- **SMTP address** – The IP address of the SMTP server. You can specify multiple IP addresses for high availability implementations. Separate multiple IP addresses with commas.
 - **Sender Email** – The mail address that will appear as the notification sender.
- iv. In the **Advanced Settings** area, specify the following optional details:
- **Sender Display Name** – The name that will appear as the sender's name.
 - **SMTP Port** – The port through which the ENE will send notifications.
 - **Recipients Domain** – The name of the domain where the recipient's email account exists.
 - **CA-PWVABaseURL** – The URL of the machine where the PWA is installed (e.g. https://www.myserver.com)

- v. Click **Finish**; the initial ENE configuration is saved and the Email notification setup message appears.



- vi. Click **Yes**; a test email is sent to the members of the Vault Admins group.

Configuring the ENE

In order for the ENE to send out notifications according to your requirements, the following must be configured:

- **Templates** – The templates determine the appearance and the content of notifications that will be sent.
- **Recipients** – Users who will receive notifications. Recipients can be specified as Safe owners according to Safe authorizations, Vault groups, or individual recipients.
- **Activities** – The activity that will initiate notifications.

Implementing Dynamic References

Each notification contains list of dynamic variables that can be used as references in the email template. Some of the dynamic variables are defined per event, and some other are fixed for all notifications.

These references can be further manipulated and enriched in the NotificationTemplateMapFile parameters.

The list below describes the list of dynamic variables that are generated by the different events:

- **General variables** – The following variables can be used in all the notifications generated by the ENE.

Variable name	Description
CA-IssuerUserName	The user name of the user who generated the notification.
CA-SafeName	The name of the Safe where the notification was generated.
CA-IssuingDate	The date when the notification was generated.
CA-IssuerFirstName	The first name of the user who generated the notification, as specified in the issuer's user information. If this information does not exist or cannot be retrieved, the notification will replace the variable with Sir or Madam.
CA-IssuerLastName	The last name of the user who generated the notification, as specified in the issuer's user information.
CA-IssuerEmailAddress	The email address of the user who generated the notification.
CA-IssuerPhoneNumber	The phone number of the user who generated the notification.

- **Vault event variables** – The following variables can be used in Vault event notifications generated by the ENE.

Variable name	Description
Request notifications:	
CA-RequestDateFrom	The initial date specified in the request.
CA-RequestDateTo	The end date specified in the request.
CA-RequestType	The type of access specified in the request. Either single or multiple access is allowed.
CA-RequestReason	The reason for the requested operation.
CA-RequestId	The unique Id of the request.
CA-ObjectFullPath	The full path of the object for which the request was created.
CA-RequestConfirmerReason	The reason why the request was confirmed.
CA-RequestIssuerUserName	The user name of the original request issuer.
CA-RequestConfirmerEmail Address	The email address of the user who is authorized to confirm the request.
CA-RequestConfirmerPhone Number	The phone number of the user who is authorized to confirm the request.
CA-RequestTicketingSystem	The ticketing system specified in the request.
CA-RequestTicketId	The unique ID of the ticket specified in the request.
License notifications:	
CA-LicenseUsageUser TypeName	The licensed user type that is about to be exceeded. This will either specify a particular user type or “total users”.
CA-LicenseUsageExisting Users	The current number of existing users.
CA-LicenseUsageLicensed Users	The maximum number of licensed users.
CA-LicenseExpirationDate	The date on which the Vault license will expire.
File/Account notifications:	
CA-UploadFullFileName	The name of the uploaded file.
CA-UploadDate	The date of the upload.
CA-UploadingUserName	The name of the user who uploaded the file.
CA-UploadFolder	The folder to which the file was uploaded.
CA-UploadFolderCifsLink	Folder to which the password or file was uploaded, with left “Root” omitted and right backslash added.

Variable name	Description
Backup/DR notifications:	
CA-ReplicationUserName	The name of the DR replicator user.
CA-MinutesSinceLast SuccessfulReplication	The number of minutes since the last successful DR replication.
CA-LastSuccessful ReplicationTime	The time of the last successful DR replication.

- **Password use events** – The following variables can be used in password use event notifications generated by the ENE.

Variable name	Description
CA-EventDateTime	The date and time when an account was accessed.
CA-UsedBy	The name of user who accessed the account
CA-Reason	The reason for accessing the account
CA-ObjectFullPath	The full path of the account that was accessed
CA-PVWABaseUrl	The base URL of the PVWA from the system options
CA-ObjectName	The name of the account that was accessed
CA-FolderName	The folder where the account is stored
CA-SafeName	The name of the Safe where the account is stored

- **Automatic password management events** – The following variables can be used in automatic password management events notifications generated by the ENE.

Variable name	Description
Account disabled notifications:	
CA-UsageSuffix	The suffix of the service account associated with this account. If this account is a service account, this value contains 'usage'.
CA-TaskName	The name of the task performed. Options are verification, reconciliation, change, and unlocking.
CA-FailReason	The reason the task failed.
CA-TaskReason	The reason the task was performed.
CA-ObjectFullPath	The full path of the account on which the task was performed.
CA-SafeName	The name of the Safe where the account is stored.
CA-FolderName	The name of the folder where the account is stored.
CA-ObjectName	The name of the account that was used.
CA-PVWABaseUrl	The base URL of the PVWA from the system options.
CA-EventDateTime	The date and time when the task was performed.

Variable name	Description
Account verification notifications:	
CA-UsageSuffix	The suffix of the service account associated with this account. If this account is a service account, this value contains 'usage'.
CA-FailReason	The reason the task failed.
CA-ObjectFullPath	The full path of the account on which the task was performed.
CA-SafeName	The name of the Safe where the account is stored.
CA-FolderName	The folder where the account is stored.
CA-ObjectName	The name of the account that was used.
CA-PVWABaseUrl	The base URL of the PVWA from the system options.
CA-EventDateTime	The date and time when the task was performed.
Notifications prior to account expiration:	
CA-ExpireDate	The date when the account will expire.
CA-ChangeDate	The date when the account will be changed.
CA-SafeName	The name of the Safe where the account is stored.
CA-FolderName	The folder where the account is stored.
CA-ObjectName	The name of the account that was used.
CA-PVWABaseUrl	The base URL of the PVWA from the system options.
CA-ObjectFullPath	The full path of the account on which the task was performed.
Unreleased password notifications:	
CA-PVWABaseUrl	The base URL of the PVWA from the system options.
CA-SafeName	The name of the Safe where the account is stored.
CA-FolderName	The folder where the account is stored.
CA-ObjectName	The name of the account that was used.
CA-ObjectFullPath	The full path of the account on which the task was performed.
Used password notifications:	
CA-EventDateTime	The date and time when the task was performed.
CA-UsedBy	The name of user who accessed the account.
CA-Reason	The reason for accessing the account.
CA-ObjectFullPath	The full path of the account on which the task was performed.
CA-PVWABaseUrl	The base URL of the PVWA from the system options.
CA-ObjectName	The name of the account that was used.
CA-FolderName	The folder where the account is stored.
CA-SafeName	The name of the Safe where the account is stored.

Users can manipulate variables in the NotificationTemplateMapFile parameters by modifying the values of the parameters listed in the following table. These parameters enable you to add or change the contents of variables that are referred to in notifications, giving you more control over the content of notifications.

After defining these variables, add the name of the variable in the relevant notification in the ENE Template.

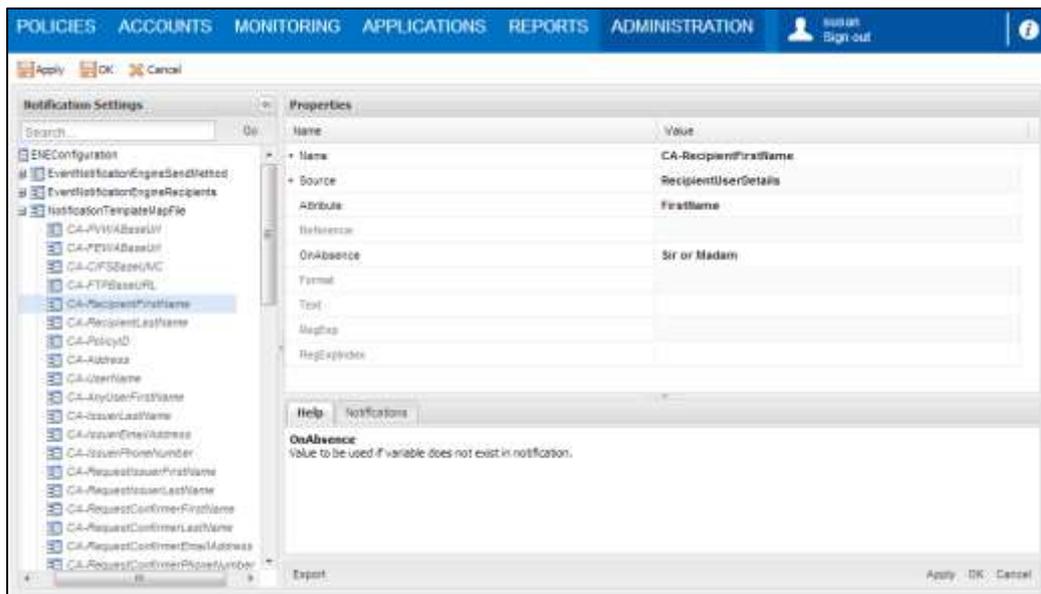
Each variable is defined with the following information:

Parameter	Description
Name	The name of the variable as it will be used in notification templates defined in the Template.xml file. All parameter names must be prefixed with CA- .
Source	The source of the information used in the variable. Possible sources are: <ul style="list-style-type: none"> ▪ RecipientUserDetails – Variables can be based on details from a recipient's user account. ▪ Category – Variables can be based on a file category defined in the Vault. The Reference attribute points to the file from which we want to fetch the category value. By default, the reference points to the file in the operation that prompted the notification. ▪ UserDetails – Variables can be based on details from a user account. The Reference attribute can be used to point to another variable that contains the username from where the user details will be retrieved (by using the Reference parameter). By default, the Reference attribute points to the issuer's username ▪ Application – Variables can be based on other variables that are created by CyberArk products. For example, variables that are used for CPM notifications. The main service account of this kind of variables is to format the text of the variables or to add "OnAbsence". ▪ Const – Variables can be based on static text defined in the 'Text' parameter. For more information, see below.
Attribute	The attribute of the source used in the variable. Any attribute that is defined in the Vault for the specified source can be used. Possible attributes are: <p>RecipientUserDetails HomePage, Email1, Email2, Email3, Cellular, Department, Fax, HomeCity, HomeCountry, HomePhone, HomeState, HomeZip, JobCity, JobCountry, JobState, JobStreetJobTitle, JobZip, FirstName, LastName, MiddleName, Notes, Organization, Pager, Profession, WorkPhone.</p> <p>UserDetails HomePage, Email1, Email2, Email3, Cellular, Department, Fax, HomeCity, HomeCountry, HomePhone, HomeState, HomeZip, JobCity, JobCountry, JobState, JobStreetJobTitle, JobZip, FirstName, LastName, MiddleName, Notes, Organization, Pager, Profession, WorkPhone.</p> <p>Application Any parameter that appears in CyberArk events. For example, CA-SafeName.</p> <p>Category Any valid file category that has been specified for the account or file.</p>
Reference	Points to another variable that contains the information that will be retrieved.
OnAbsence	Specifies text that is used in the notification if the variable cannot be replaced with a value.

Parameter	Description
Format	The format that will be used to display the value retrieved to replace the variable. Possible formats are: <ul style="list-style-type: none"> ▪ Text – The variable value will be specified as free text. This is the default format. ▪ Date – The variable value will be specified in date format. ▪ Time – The variable value will be specified in time format. ▪ DateTime – The variable value will be specified in date and time format. ▪ Url – The variable value will specify URL links.
Text	Defines the text that will be displayed if the source is Const .
RegExp	Specifies the regular expressions that will be used. This parameter is only relevant if the Format parameter specifies RegExpFilter .
RegExpIndex	Specifies the value that will be used from the list of expressions listed in the RegExp directive. This parameter is only relevant if the Format parameter specifies RegExpFilter .

To Define Recipient User Details

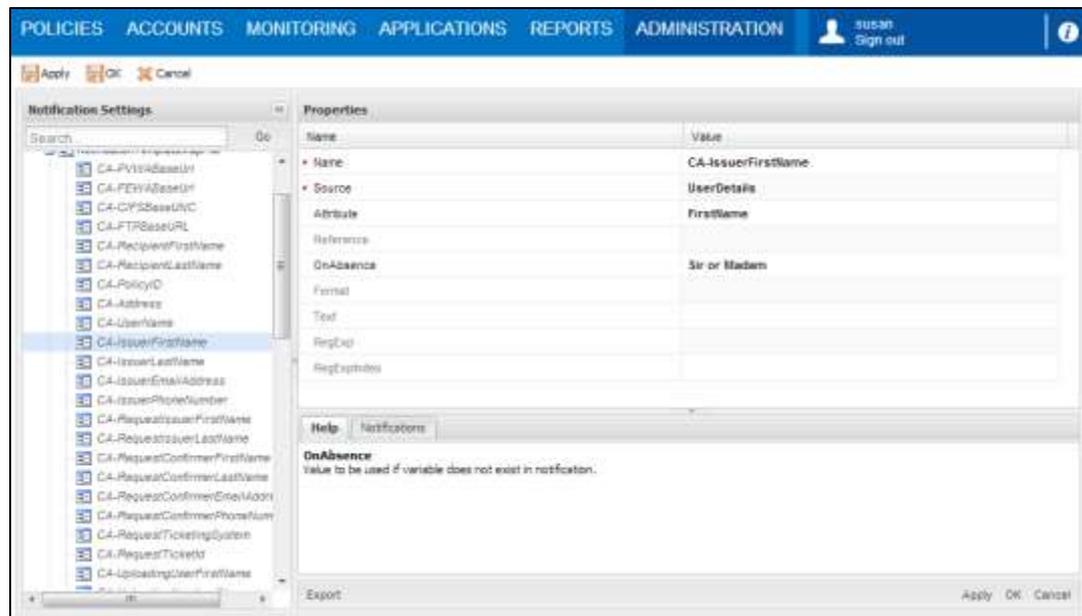
Information can be retrieved from an attribute specified in a recipient user's details, as shown in the following example:



This example defines a new dynamic variable called **CA-RecipientFirstName**. The information for this variable will be taken from the First Name specified in the recipient user account's details. If this information does not exist or cannot be retrieved, the notification will replace the variable with **Sir or Madam**.

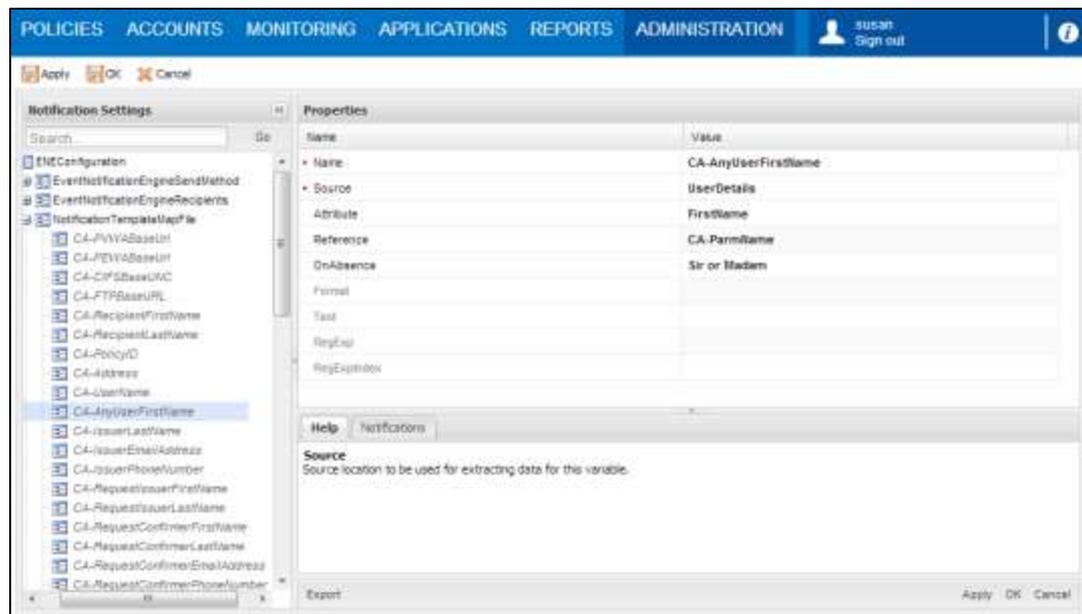
To Define User Details

Information can be retrieved from an attribute specified in an issuer's user details, as shown in the following example:



This example defines a new dynamic variable called **CA-IssuerFirstName**. The information for this variable will be taken from the First Name specified in the issuer user account's details. If this information does not exist or cannot be retrieved, the notification will replace the variable with **Sir or Madam**.

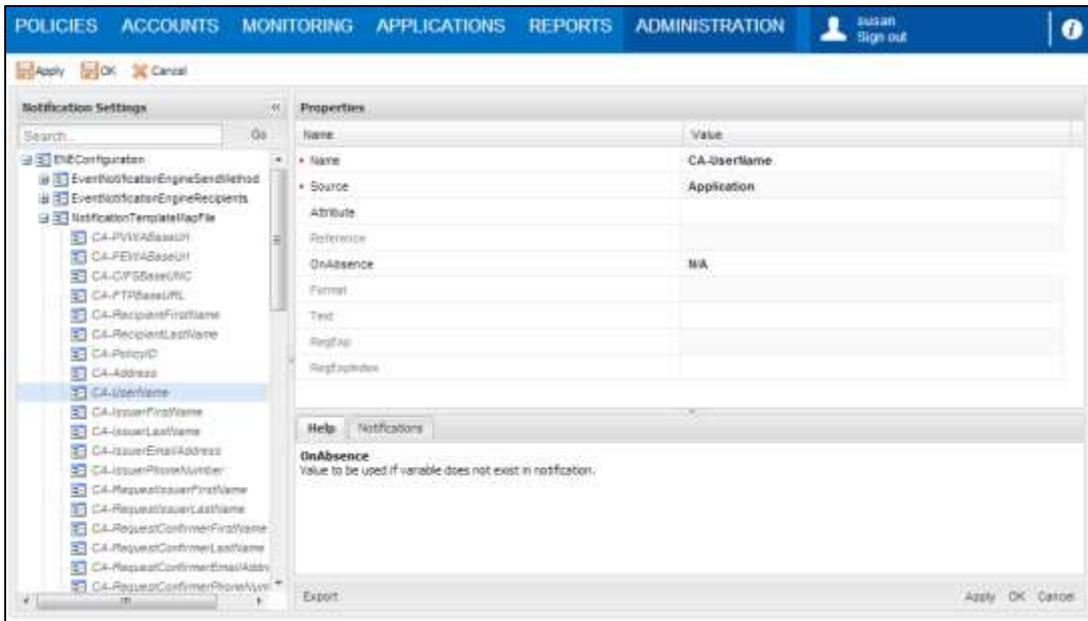
To retrieve information from a user whose information is specified in a different parameter, in the Reference parameter, specify the name of the parameter where the information will be found, as shown in the following example:



This example retrieves the username of the user that is specified in the **CA-ParamName** parameter.

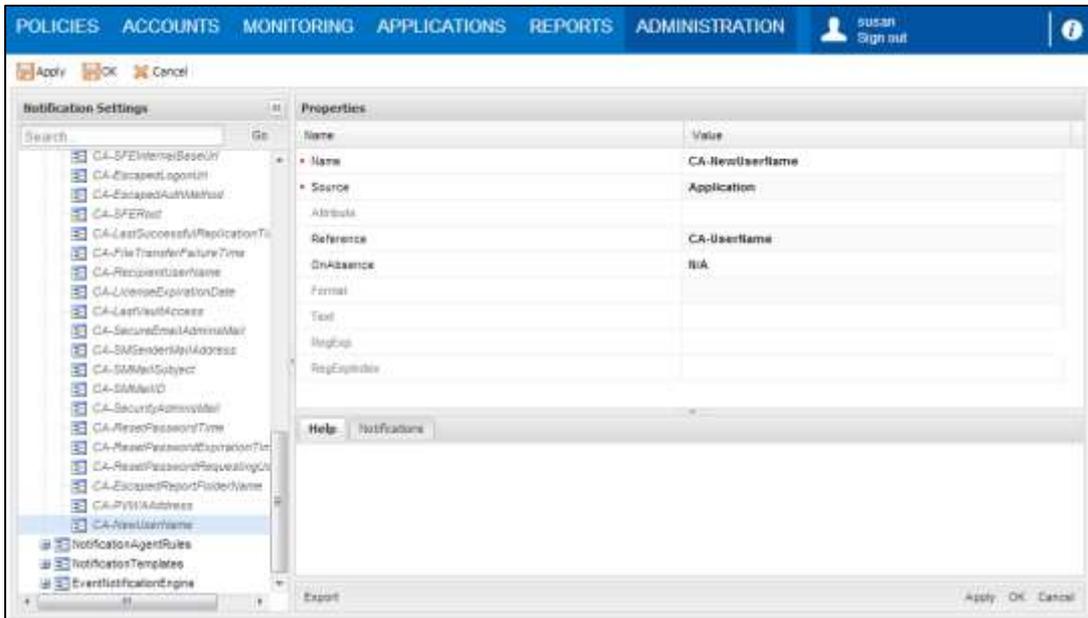
To Define Details of a CyberArk Component Event

Information can be retrieved from an event performed by a CyberArk component, as shown in the following example:



This example defines a new dynamic variable called **CA-UserName**. The information for this variable will be taken from the events generated by each activity performed by a CyberArk component. If this information does not exist or cannot be retrieved, the notification will replace the variable with **N/A**.

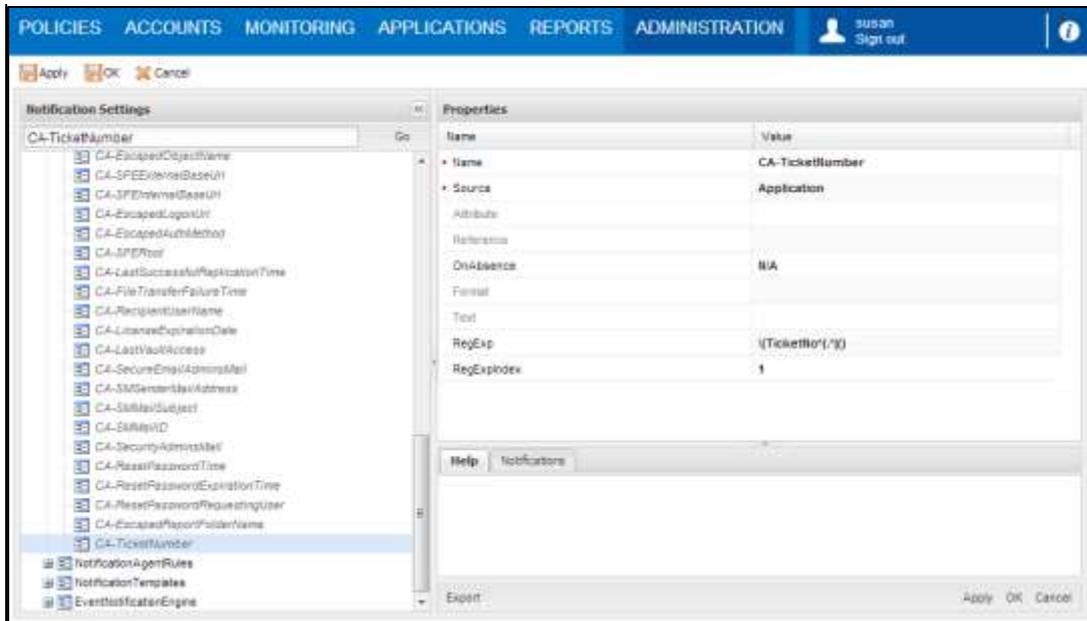
To retrieve information from another parameter, in the Reference parameter specify the name of the parameter where the information will be found, as shown in the following example:



This example creates a parameter named **CA-NewUserName** which contains the same value as the “CA-UserName” parameter. By using one of the formats above, you can alter the new parameter data, for example, by applying the RegExp filter.

To Define a Regular Expression

The **RegExpFilter** retrieves data from variables that use Regular Expressions. Additional directives indicate the information to display in the notification. The following example shows the parameters that are specified in this type of variable.



This example defines a new dynamic variable called **CA-TicketNumber**. The information for this variable will be taken from an event performed by a CyberArk component. If this information cannot be retrieved, **N/A** will be displayed instead of a relevant value. However, if a value is retrieved, it will be displayed as a regular expression using the first value in the RegExp directive, as indicated by the RegExpIndex directive. This example will return the value of a ticket number.

Configuring Email Notification Templates

A set of templates determines the content of notifications that are sent to recipients. The notification settings contain the default templates that the ENE uses for notifications whenever an event occurs. You can customize these templates and create additional custom templates. The ENE supports up to 1024 templates.

These templates can be modified to users' specific needs, so that they contain information that meets enterprise standards.

The default templates contain the following notification templates:

- **Request notifications** – Templates for Safe and password request notifications, as well as confirmation, rejection, and deletion notifications.
- **New file notifications** – A template for when a new file is stored in the Safe.
- **Password expiration notifications** – Templates for password and account group expiration notifications and upcoming expiration notifications.
- **Automatic management notifications** – Templates for auto-management and verification failure notifications
- **Password use notifications** – Templates for notifications when a privileged password is used.
- **Backup/DR notifications** – A template for when backup or DR replications are not performed successfully.
- **Auto-detection notifications** – Templates for notifications following auto-detection processes.
- **Report notifications** – Templates for notifications about report generation.
- **Monitoring notifications** – A template for notifying users when a Privileged Account Security solution component isn't active for a predetermined length of time.

The URLs specified in ENE templates must use the correct case. If the name of the PVWA was changed during installation, the URLs in the ENE templates must be changed accordingly.

To Modify Email Notification Templates

1. Click ADMINISTRATION, then in the **System Configuration** page click **Notification Settings**; the Notification Settings page appears.
2. Select **Notification Templates**; then specify the following parameter:
 - **Charset** – The character set that will be used for the notification. The default value is **utf-8**. For a list of standard character sets, refer to: <http://www.iana.org/assignments/character-sets>.
3. Specify the general parameters that will be applied to all the templates in this file.
 - **GeneralHeader** – The header that will appear in each notification.
 - **GeneralFooter** – The footer that will appear in each notification.

These parameters can be overridden by specifying the same parameters in individual templates.

4. Select the template to customize. You can modify the default content of the selected template to customize the text that will appear in notifications. You can also add or remove parameters to create your own custom template.

Some of the parameters used in templates are specified in the NotificationTemplateMapFile parameters which you can use for reference. This list specifies the parameter names that can be used in templates, and their source in the Vault. In addition, these parameters specify a default value to use in the template if the parameter value does not exist. For a full list of dynamic references that can be used in notifications, refer to *Implementing Dynamic References*, page 805.

You can modify the following template properties:

- **ID** – The unique identifier of the template. This identifier is used in the Rules.xml configuration file to specify the template that will be used to create and send notifications.
- **Type** – The format type of notification, which is Text.
- **Charset** – You can set a character set that will be used for this specific template, overriding the character set specified in the general parameters at the top of the configuration file.
- The **Subject** parameter specifies the subject of the notification. This text will appear as the subject of the notification. The subject can be specified in either of the following ways:
 - **Fixed text** – The subject can be specified by fixed text, as shown in the following example:

The screenshot shows a 'Properties' window with a table of configuration parameters. The 'Subject' field is highlighted with a red oval and contains the text 'Notification: Safe access request'.

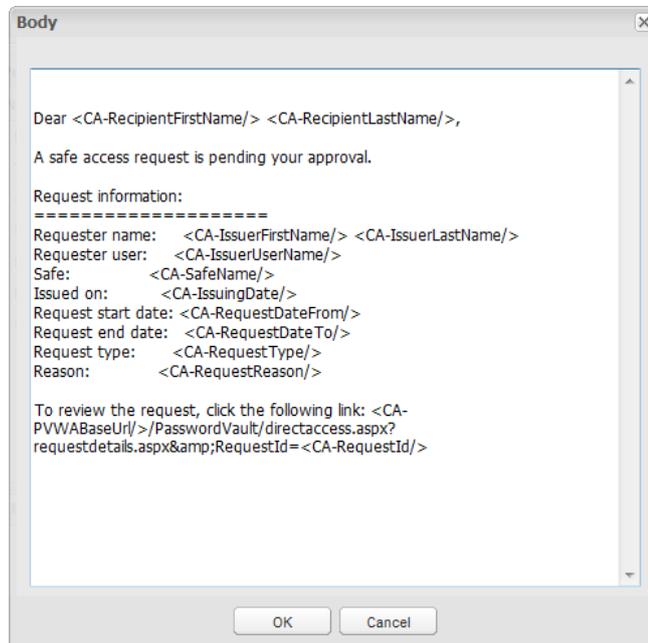
Name	Value
ID	1
Type	Text
Charset	
Header	
Subject	Notification: Safe access request
Body	Dear <CA-RecipientFirstname> <CA-RecipientLastname>, A safe ...
Footer	

- **Dynamically** – The subject can be specified dynamically by specifying dynamic references as described in *Implementing Dynamic References*, page 805. For example:

The screenshot shows a 'Properties' window with a table of configuration parameters. The 'Subject' field is highlighted with a red oval and contains the text 'Notification for <CA-RecipientFirstname> <CA-RecipientLastname>'.

Name	Value
ID	1
Type	Text
Charset	
Header	
Subject	Notification for <CA-RecipientFirstname> <CA-RecipientLastname>
Body	Dear <CA-RecipientFirstname> <CA-RecipientLastname>, A safe ...
Footer	

- To customize the header for this template and to override the header that is specified in the general parameters at the top of the configuration file, specify the **Header** for this template.
- The **Body** section determines the content of the notification. This information comprises text and dynamic references to information that is generated when activities occur in the Vault.



Note: Do not change the dynamic references manually. The ENE will replace them with specific details about the detected Vault activity when the notification is created.

For a full list of dynamic references that can be used in notifications, refer to *Implementing Dynamic References*, page 805.

- To customize the footer for this template and to override the footer that is specified in the general parameters at the top of the configuration file, specify the **Footer** for this template.
5. Click **Apply** to save the new parameter values and stay in the Notification Settings page,
or,
Click **OK** to save the new parameter values and return to the System Configuration page.

To Create New Email Notification Templates

1. In the Notification Settings page, select **Notification Templates**; a list of existing templates is displayed.
2. Either copy an existing template and change its parameters and properties or create a completely new template with default settings:
 - To add a new template:
 - i. Right-click **Notification Templates**, then in the pop-up menu select **Add Template**; a new template is created.
 - ii. Modify existing parameters and properties and/or create new ones for this template.

- To add a new template based on an existing template:
 - i. Right-click on the template to copy, then in the pop-up menu select **Copy**.
 - ii. Right-click on **Notification Templates**, then in the pop-up menu select **Paste Template**; the existing template is added to the list of templates using the same name as the original template.
 - iii. In the new template, change the ID name and specify a unique name for that template.
 - iv. Modify existing parameters and properties and/or create new ones for this template.
- 3. Click **Apply** to save the new template and stay in the Notification Settings page, or,
Click **OK** to save the new template and return to the System Configuration page.

Configuring Recipients

In order for the ENE to send notifications about activities in the Vault, it must be able to identify recipients and activities, and make the connection between the two. This configuration process comprises the following two steps:

- **Step 1: Defining Recipients** – For more information, refer to *Defining Recipients below*.
- **Step 2: Connecting Activities and Recipients** – For more information, refer to *Connecting Activities and Recipients, page 824*.

Defining Recipients

The ENE uses the list of recipients defined in the EventNotificationEngineRecipients parameters to determine which users will receive notifications.

You can specify users according to Safe ownership or email addresses, as follows:

- **Safe ownership** – Notifications can be sent to users according to Safe ownership. In addition, several options enable you to be more specific about recipients:
 - The person who stored the password in the Vault can be excluded.
 - You can specify only members of a specific group.
- **Fixed email addresses** – Notifications can be sent to a fixed list of email addresses, regardless of their Safe ownership.
- **Other specifications** – Other individuals or groups of users can be specified.

Types of recipients can be defined in any of the following ways:

- **User name** – The name of a specific user in the Vault.
- **Group name** – The name of a group in the Vault.
- **Vault query** – A set of criteria that determine one or more users or groups in the Vault.
- **Application** – An application will set the recipient's user name as part of the application's event data.
- **Email** – A specific email, regardless of whether or not the user is defined in the Vault.
- **Dynamically** – A user, group, or email can be taken from a specified parameter value.

In addition, specific users can be excluded from the recipients list even if they meet the criteria specified in any of the above definitions.

For Vault users, the ENE uses the email address specified in the `SendMethod` parameter in the `EventNotificationEngineSendMethod` parameters. The default value for this parameter is `Email2`, which refers to the user's Business Email address that is specified in the User's properties in the Vault.

For users who are defined in an external directory, the ENE uses the email address set in the LDAP server.

To Define Recipients

1. Click **ADMINISTRATION**, then in the **System Configuration** page click **Notification Settings**; the Notification Settings page appears.
 2. Expand **EventNotificationEngineRecipients**; the list of existing recipients is displayed.
 3. Right-click **EventNotificationEngineRecipients**, then in the pop-up menu select **Add Recipient**; a new recipient is created.
 4. In the **Recipient** parameter, in the **ID** property, specify the unique identifier of the group of recipients to define. This identifier is used in the `NotificationAgentRules` to specify the group of recipients that will receive notifications.
 5. Define the recipients.
 - To define a specific Vault user, group, or email address:
 - i. Right-click the new recipient, then in the pop-up menu select **Add NamedRecipientObject**; a new named recipient object is created.
 - ii. Specify the following properties:
 - **Type** – The type of recipient. Specify **User**, **Group**, or **eMail**.
 - **Name** – The name of the Vault user or group, or the recipient's email address.
- Note:** You can define up to 256 named recipient objects, although it is more efficient to define a Vault query if notifications will be sent to several users or groups.

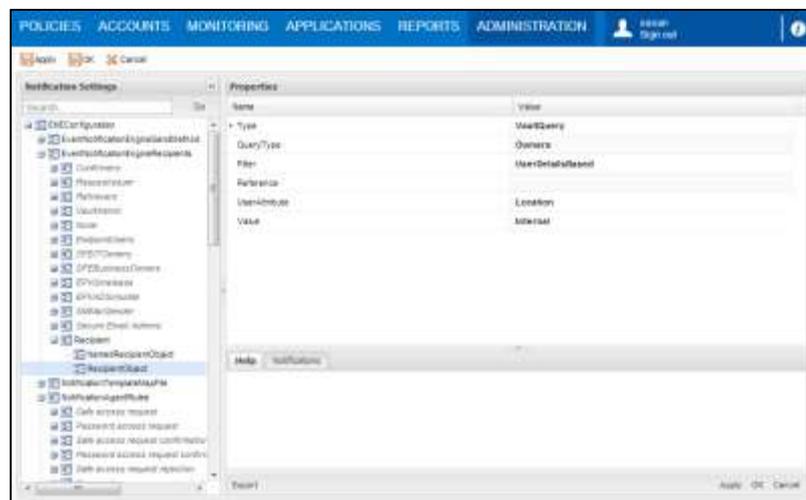
- To add a recipient according to specific criteria – Enables you to add recipients according to Safe ownership, permissions, or activity.
 - i. Right-click the new recipient, then in the pop-up menu select **Add RecipientObject**; a new recipient object is created.
 - ii. Define the notification recipients in any of the following ways:
 - **To define a Vault query** – Define the criteria by which users and groups will be identified:
 - a. In the **Type** parameter, specify **VaultQuery**.
 - b. In the **QueryType** parameter, specify one of the following values:
 - **Owners** – Notifications will be sent to Safe owners. The number of recipients can be filtered by Safe authorizations, as described below.
 - **Requestor** – Notifications will be sent to the user who performed the activity that initiated the notification.
 - c. In the **Filter** parameter, specify one of the following values:
 - **PermissionBased** – Safe owners will be filtered according to their permissions in the Safe, depending on the specified value, as listed in the table below:

Value	Description
UsePassword	The 'Use accounts' permission
Retrieve	The 'Retrieve accounts' permission
List	The 'List accounts' permission
Create	The 'Add accounts' permission
Update	The 'Update password value' permission
UpdateObjectProperties	The 'Update password properties' permission
InitiateCPMChange	The 'Initiate CPM password management operations' permission
InitiateCPMChangeWithManualPassword	The 'Specify next password value' permission. This authorization is only relevant if 'InitiateCPMChange' is specified.
Rename	The 'Rename accounts' permission
Delete	The 'Delete accounts' permission
Unlock	The 'Unlock accounts' permission
Administer	The 'Manage Safe' permission
ManageOwners	The 'Manage Safe Members' permission
Backup	The 'Backup Safe' permission
ViewAudit	The 'View audit' permission
ViewOwners	The 'View Safe Members' permission
Confirm	The 'Authorize password requests' permission

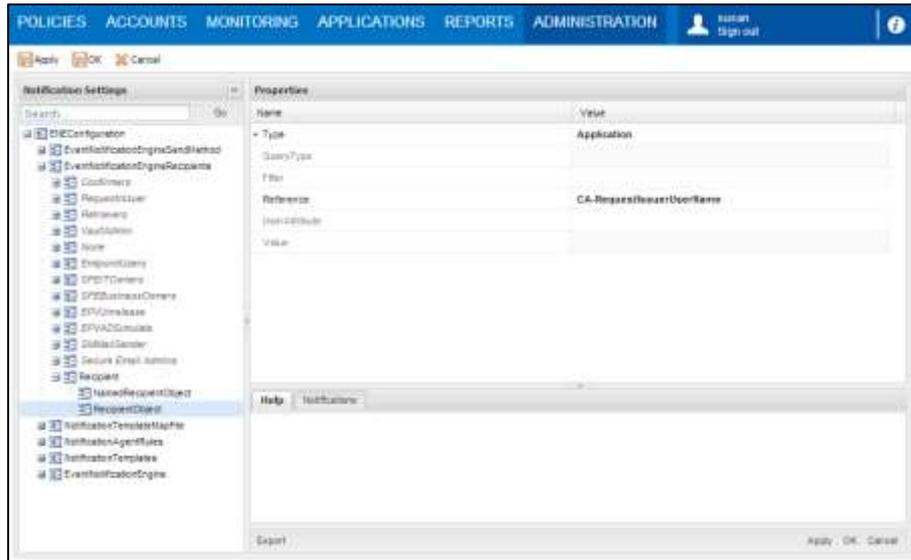
Value	Description
AccessWithout Confirmations	The 'Access Safe without Confirmation' permission
CreateFolder	The 'Create folders' permission
DeleteFolder	The 'Delete folders' permission
MoveFrom and MoveTo	The 'Move accounts/folders' permission
ValidateContent	The 'Validate Safe Content' permission

- **UserDetailsBased** – Safe owners will be filtered according to their Location in the user hierarchy, depending on the specified value.

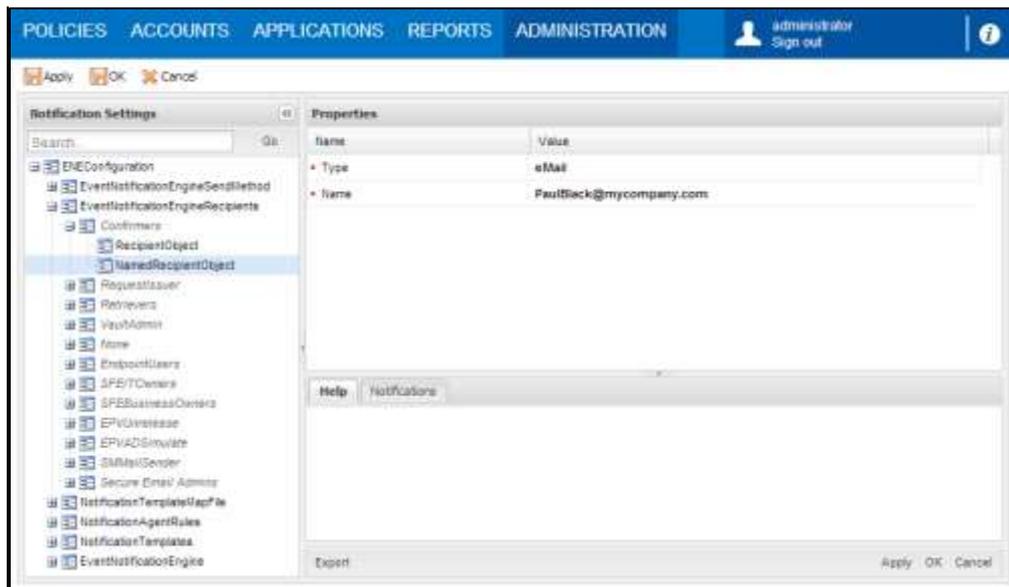
The following example shows a query that will identify Safe owners in a Location called 'Internal':



- To enable an application to define a recipient:
 - a. Create a **RecipientObject** parameter.
 - b. In the **Type** parameter, specify **Application**.
 - c. In the **Reference** parameter, specify the variable that the application will replace with a recipient's user name. In the following example, the application will set the user name that corresponds to the 'CA-RequestIssuerUserName' variable.

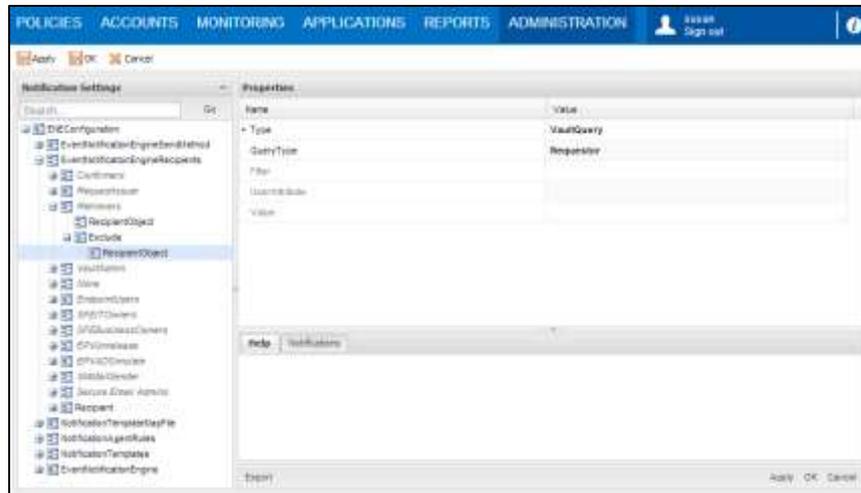


- To define an email recipient:
 - a. Create a **NamedRecipientObject** parameter.
 - b. In the **Type** parameter, specify **eMail**.
 - c. In the **Name** parameter, specify the recipient's email, as shown in the following example.



- To exclude recipients - Specify any users to exclude from the recipients list, even if they meet the criteria specified in the previous step.
 - i. Right-click the new recipient, then in the pop-up menu select **Add Exclude**; a new recipient object is created.

- ii. Right-click the new recipient object, then in the pop-up menu select one of the following:
 - **Add RecipientObject** – Define a recipient object to exclude specific users:
 - **Vault Query** – Define a Vault query to exclude the user whose activity initiated the notification, as shown in the following example.



- **eMail** – Specify an email address to exclude it from the list of recipients, as shown in the following example.
 - **Add NamedRecipientObject** – Define a specific user or group to exclude from the list of recipients:
 - **Type** – The type of recipient to exclude. Specify either **User** or **Group**.
 - **Name** – The name of the Vault user or group.
6. Click **Apply** to save the modified recipients list and stay in the Notification Settings page,
or,
Click **OK** to save the recipients list and return to the System Configuration page.

To Modify Existing Recipients

1. Click **ADMINISTRATION**, then in the **System Configuration** page click **Notification Settings**; the Notification Settings page appears.
2. Expand **EventNotificationEngineRecipients**; the list of existing recipients is displayed.
3. Select the recipient to customize and change the relevant properties. You can modify the default values of any of the recipient properties and add or remove recipients to create a list of recipients that meets your enterprise requirements and procedures.
4. Click **Apply** to save the modified recipients list and stay in the Notification Settings page,
or,
Click **OK** to save the recipients list and return to the System Configuration page.

To Create New Recipients

1. In the Notification Settings page, select **EventNotificationEngineRecipients**; a list of existing recipients is displayed.
2. Either copy an existing recipient and change its parameters and properties or create a completely new recipient:
 - To add a new recipient:
 - i. Right-click **EventNotificationEngineRecipients**, then in the pop-up menu select **Add Recipient**; a new recipient is created.
 - ii. Specify the relevant parameters as described in *To Define Recipients*, page 819.
 - To add a new recipient based on an existing recipient:
 - i. Right-click on the recipient to copy, then in the pop-up menu select **Copy**.
 - ii. Right-click on **EventNotificationEngineRecipients**, then in the pop-up menu select **Paste Recipient**; the existing recipient is added to the list of recipient using the same name as the original recipient.
 - iii. In the new recipient, change the ID name and specify a unique name for that recipient.
 - iv. Modify existing parameters and properties and/or create new ones for this recipient.
3. Click **Apply** to save the modified recipients list and stay in the Notification Settings page,
or,
Click **OK** to save the recipients list and return to the System Configuration page.

Connecting Activities and Recipients

The `NotificationAgentRules` list contains sets of rules that determine the activities that initiate notifications and the groups of recipients that will be notified. These rules can either be used as they appear in the default `NotificationAgentRules` list, or they can be customized to meet your organizational requirements.

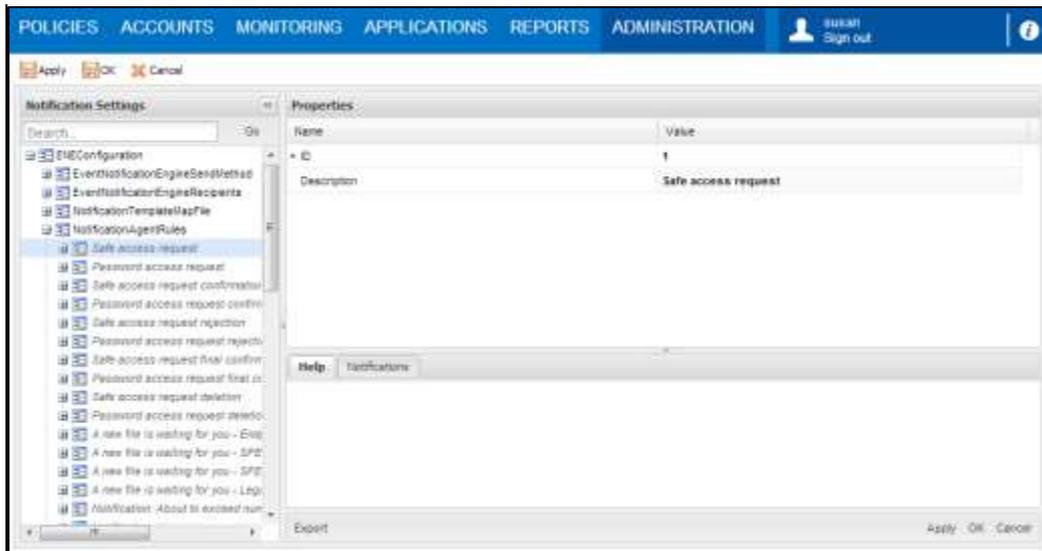
These rules specify criteria which filter events that occur in the Vault and initiate notifications.

Note: The parameters in the `NotificationAgentRules` list must be specified in the order listed below.

To Connect Activities and Recipients

1. Click ADMINISTRATION, then in the **System Configuration** page click **Notification Settings**; the Notification Settings page appears.
2. Select **NotificationAgentRules**.
3. For each rule, specify the following properties to define the rule's **general information**:
 - **ID** – The unique ID of the rule.
 - **Description** – A description of the rule.

The following example defines the general information for rule #1. This rule will issue notifications when a Safe access request is created in the PVWA.



4. For each rule, specify the rule's **filter criteria**. This section defines the filter criteria that will initiate a notification process.
 - **Source** – The source of the activity that will initiate a notification process. This filter is defined by the following property:
 - **ID** – The unique ID of the CyberArk component or implementation. This must not be changed.

The following table lists the SourceIDs and the implementation or component they represent:

SourceID	Component/Implementation
5	IBV/SDV implementations
10	Privileged Account Security implementations
15	Vault server

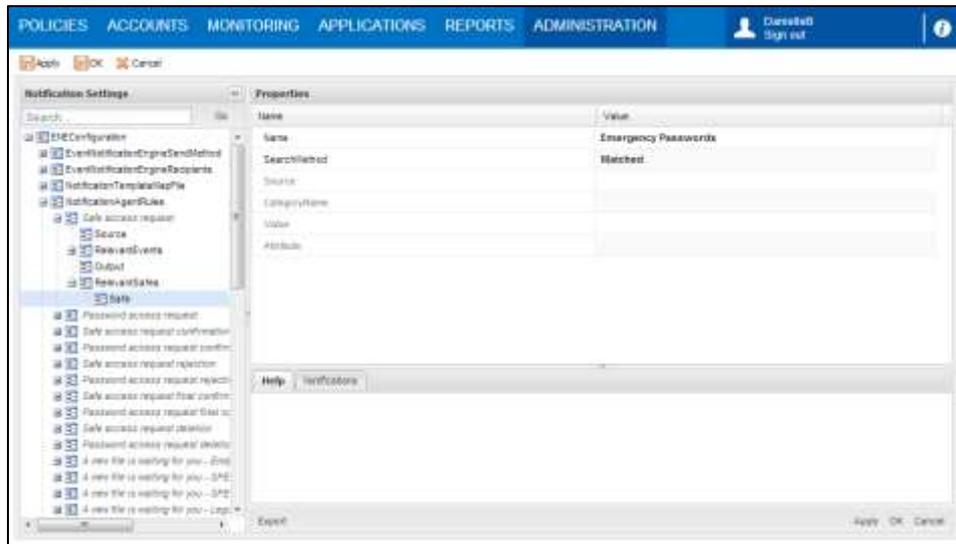
- **Client** – The Privileged Account Security client where the activity took place. This filter is defined by the following property:
 - **ID** – The unique ID of the CyberArk component where activities will take place to prompt this notification rule. This must not be changed.
- **RelevantSafes** – Identifies Safes to search for activities that will initiate notifications. The filter for each Safe is defined by the following properties:
 - **Name** – The name of the Safe to search for activity.

- **SearchMethod** – Specify either the full Safe name or a Safe pattern according to the SearchMethod value, as described in the table below:

Parameter	Description
Value	The full name of the Safe or part of it. If the SearchMethod is 'RegExp', a wildcard can be specified.
SearchMethod	The search method that will be used to identify the Safe. Specify one of the following values: <ul style="list-style-type: none"> ▪ Matched – The search will identify the exact Safe name that is specified. ▪ RegExp – The search will identify the specified Safe name as a regular expression. ▪ BitwiseAnd – The search will identify the Safe by performing a bitwise match of the Safe option field with the specified value. This search method is used with the Safe details specified in the Value and Attribute properties, described below.

By default, all Safes that are owned by the Notification Engines group will be searched for activity. This parameter, therefore, is optional and only needs to be specified to issue notifications about activities in a particular Safe.

In the following example, the RelevantSafes parameters specify the Safe and the Search method that will be searched for activity in this rule using the SafeName parameter:

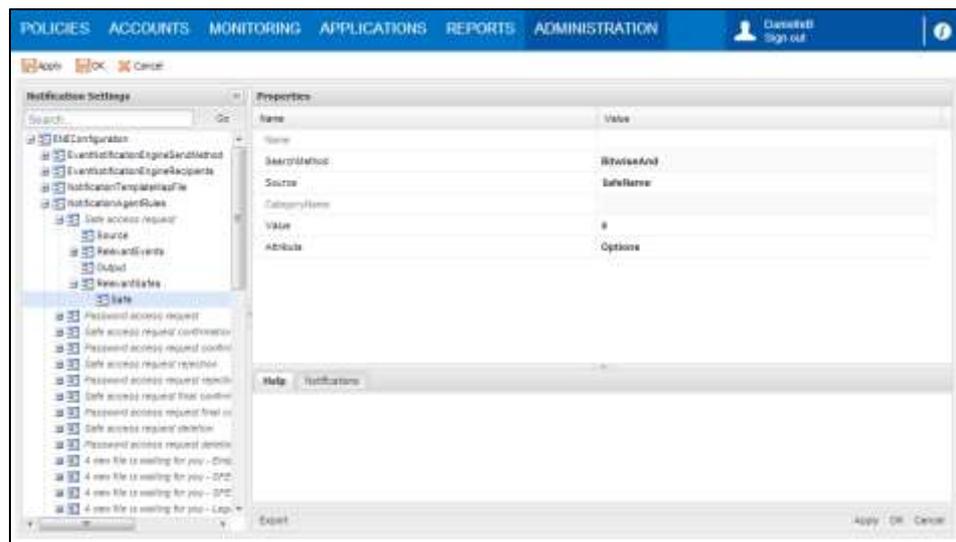


- **Source** – Defines the type of search which determines the rest of the parameters to specify. The Source parameter has the following possible values:

Value	Description
SafeName	Identifies Safes by their name or part of it.
SafeDetails	Identifies Safes by one or more properties.
CategoryExists	Identifies Safes by a file category.

- **SafeDetails** – Safe file options that identify the Safe, using the following properties:
 - **SearchMethod** – Performs a bitwise match of the Safe option field with the value to identify the Safe. The valid value is **BitwiseAnd**.
 - **Value** – A number that represents the bit mask of the option field. The valid value is **Number**.
 - **Attribute** – The attribute name to take from the Safe details.
 - Note:** This parameter refers to internal Safe properties and it must not be changed. The valid value is **Options**.

In the following example, the RelevantSafes parameter specifies the Safes whose option fields bitwise matches the value of 4:



- **RelevantFolders** – The Folder parameter specifies the name of a folder, or part of it, for which notifications will be sent when a file is stored in the Safe. Rules that do not specify this parameter will send notifications for events that occur in all folders in the Safe.

The Folder name is defined by the following properties:

- **Name** – The name, or part of it, of the folder to search for activity. Make sure you specify the precise name of the folder, with the correct uppercase and lowercase characters.
- **SearchMethod** – The search method that will be used to identify the Safe. Specify **RegExp**.
- **Source** – Determines that the search is by folder names. Specify **FolderName**.

Specify more than one Folder parameter in a rule to determine multiple parts of a folder name. For example,

- **To specify a single filter** – Add one Folder parameter and specify 'Important' in the Name property to send notifications to all folders whose name contains the word **Important**.
- **To specify multiple filters** - Add one Folder parameter and specify Important in the Name property, then add another Folder parameter and specify **with** in the Name property to send notifications to all folders whose name contains both **Important** and **with**.

- **RelevantEvents** – The EventType parameter specifies the unique ID of the Vault activity that initiates a notification. Rules that do not specify this parameter will send notifications for all events that occur in the Safe.

Any of the following EventType IDs can be specified:

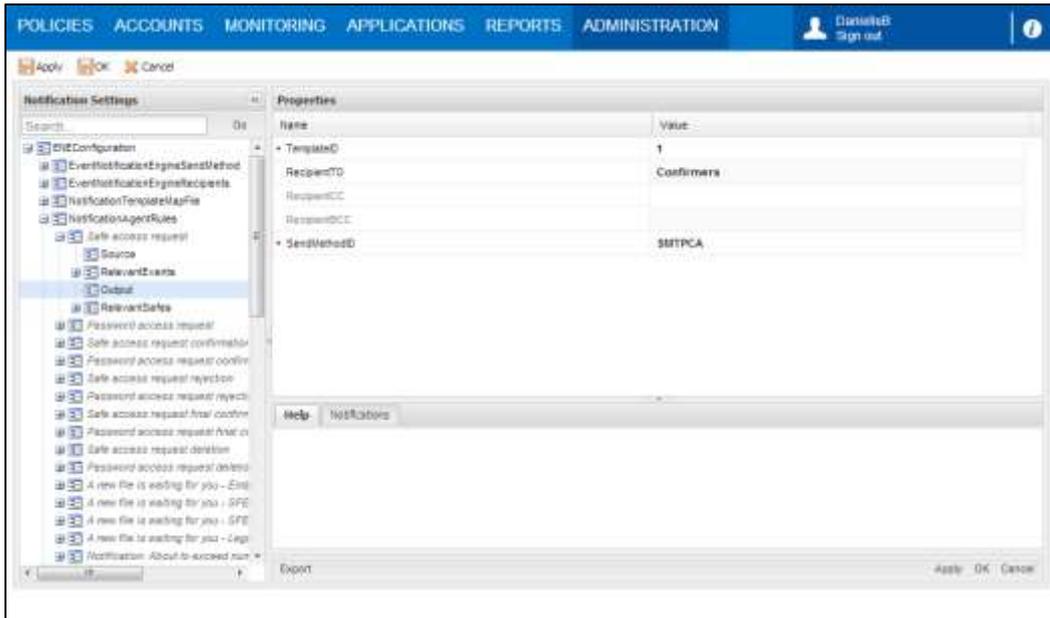
Activity	SourceID	EventTypeID
A request to open a Safe has been created	10	1
A request to retrieve a password or file from the Vault has been created	10	2
Confirmation from an authorized user has been received in response to a request to open a Safe	10	3
Confirmation from an authorized user has been received in response to a request to retrieve a password or file from the Vault	10	4
A request to open a Safe has been rejected by an authorized user	10	5
A request to retrieve a password or file has been rejected by an authorized user	10	6
The final confirmation from an authorized user has been received in response to a request to open a Safe	10	7
The final confirmation from an authorized user has been received in response to a request to retrieve a password or file	10	8
A request to open a Safe has been deleted	10	9
A request to retrieve a password or file has been deleted	10	10
A password is about to expire and will be changed automatically by the CPM	10	15
A password will expire soon	10	16
An account group is about to expire and will be changed automatically by the CPM	10	17
An account group will expire soon	10	18
A password will no longer be auto-managed	10	19
A password could not be verified	10	20
A privileged password has been used	10	21
An account must be changed and released	10	22
An automatic detection process has failed	10	23
An automatic detection process has finished successfully	10	24
An automatic detection process detected a new unmanaged account	10	25
An automatic detection process simulation has failed	10	26
An automatic detection process simulation has finished successfully	10	27

Activity	SourceID	EventTypeID
The maximum number of a specific type of licensed CyberArk user will be reached soon	15	51
The maximum number of total licensed CyberArk users will be reached soon	15	52
A password or file has been stored in the Vault	15	101
The CyberArk DR replication has not been completed successfully during the predefined period of time.	15	151
The CyberArk backup replication has not been completed successfully during the predefined period of time.	15	152
The CyberArk Vault license is about to expire.	15	153
A scheduled report was generated successfully	10	201
A scheduled report could not be generated	10	202
A report that was defined and generated immediately was generated successfully	10	203
A report that was defined and generated immediately could not be generated	10	204

For more information about these events, refer to the Privileged Account Security Reference Guide.

5. For each rule, specify the rule's **Output** options. This section defines who the notification will be sent to, what it will contain, and how it will be sent.
 - **TemplateID** – This parameter specifies the unique ID of the template in the NotificationTemplates list that determines the content of the notification. For more information about configuring templates, refer to *Configuring Email Notification Templates*, page 815.
 - **Recipients** – Specify the ID of the recipients group that is defined in the EventNotificationEngineRecipients list. For more information, refer to *Defining Recipients*, page 818. Use any of the following parameters:
 - **RecipientTO** – Notifications will be sent to users specified in this group.
 - **RecipientCC** – A copy of notifications will be sent to users specified in this group.
 - **RecipientBCC** – A blind copy of notifications will be sent to users specified in this group.
 - **SendMethodID** – This parameter specifies the ID of the protocol used to send notifications. This is defined in the EventNotificationEngineSendMethod parameters. For more information, refer to *Enabling the ENE*, page 803.

In the following example, the notification for this rule will use template #1 in the NotificationTemplates list, and it will be sent to the users defined in the 'Confirmers' recipient group in the EventNotificationEngineRecipients list, with a copy to the users defined in the 'Vault Administrators' recipient group. This notification will be sent using the protocol defined with the 'SMTPCA' ID in the EventNotificationEngineSendMethod parameters.



6. Specify any Safes to **Exclude** from the activities search:
 - **Safe** – The Safe parameter specifies the Safe(s) to exclude from the recipients list, using the Safe parameters described above.
 In the following example, the 'InternalPasswords' Safe will be excluded from the search for activities, and no notifications will be sent for activities in this Safe.
7. Click **Apply** to save the modified rules list and stay in the Notification Settings page,
 or,
 Click **OK** to save the rules list and return to the System Configuration page.

Configuring Vault Notifications

Notifications can be issued for specific activities that take place in the Vault, and Safe owners, as well as specified recipients can be notified of these activities. These notifications are configured by the `VaultEventNotifications` parameter in the `DBParm.ini` file.

- **Requests** – The following values enable the ENE to send notifications for requests to access a Safe, a password, or a file:
 - **NotifyOnNewRequest** – A notification will be sent whenever a request is created. This notification is enabled by default.
 - **NotifyOnConfirmRequest** – A notification will be sent whenever a request is confirmed by an authorized user.
 - **NotifyOnRejectRequest** – A notification will be sent whenever a request is rejected by an authorized user. This notification is enabled by default.
 - **NotifyOnConfirmRequestByAll** – A notification will be sent when confirmation from all authorized users is received for a request. This notification is enabled by default.
 - **NotifyOnDeleteRequest** – A notification will be sent whenever a request is deleted. This notification is enabled by default.
- **File activities** – The following parameter enables the ENE to send notifications for file activities in the Vault:
 - **NotifyOnStoreObject** – A notification will be sent when a new password or file is stored in the Vault.
- **Licensing** – The following notifications about licensing will be sent to recipients:
 - **License usage** – Licensing notifications will be sent automatically to recipients each time a license message is issued to the ITALog. For more information, refer to *Managing the CyberArk License*, page 868.
 - **License expiration** – A notification will be sent before the Vault license expires. By default, this notification is sent seven days before the license expires, and then every day after that until the license is renewed. The number of days before sending a notification prior to license expiration can be configured by the **NotificationPriorLicenseExpiration** parameter in `DBParm.ini`.
- **Backup and DR** – Notifications will be sent automatically to recipients whenever backup and DR replications are not completed successfully. For more information, refer to *Managing the CyberArk License*, page 868.

Configuring CPM Notifications

Notifications can be issued for passwords that are managed automatically by the CPM, and recipients can be notified of password management activities according to platform. These parameters are specified in the platform and can be used to override the ENE configuration.

To Configure CPM Notifications

1. Log onto the PWWA as an Administrative user.
2. Click **ADMINISTRATION** to display the **System Configuration** page, then click Platform Management to display a list of supported target account platforms.
3. Select the platform to configure, then click **Edit**; the settings page for the selected platform appears.
4. Expand **Automatic Password Management**, and then select **Notifications**; the CPM parameters are displayed with their default values.
5. In the **Password Change** parameters, specify the following parameter:
 - **DaysNotifyPriorExpiration** – The number of days before a password is changed that a notification will be sent to recipients. Optionally, you can specify a re-notification interval that determines the number of days between notifications for the same password expiration and a re-notification period that determines the period of time during which these notifications will be sent. After the re-notification period, no more notifications will be sent for this password expiration.
6. In the **Notification** parameters, specify the following parameters:
 - **NFNotifyPriorExpiration** – Whether or not specified recipients will receive notifications before a password expires.
 - **NFPriorExpirationRecipients** – An optional parameter that lists the email addresses that notifications will be sent to. This list overrides the recipients defined in the EventNotificationEngineRecipients list.
 - **NFFromHour** – The hour when notification will begin.
 - **NFToHour** – The hour when notification will end.
 - **NFInterval** – The interval in minutes between the notification tasks.
 - **NFNotifyOnUnreleasedPasswords** – Whether or not specified recipients will receive notifications when an account is not released after the time defined in MinValidityPeriod. This parameter is not relevant if the platform is a group platform.
 - **NFOnUnreleasedPasswordRecipients** – The email addresses of users who will receive notifications when an account is not released after the time defined in MinValidityPeriod.
 - **NFNotifyOnPasswordDisable** – Whether or not specified recipients will receive notifications when a password is disabled.
 - **NFOnPasswordDisableRecipients** – An optional parameter that lists the email addresses of users who will receive notifications when a password is disabled. This list overrides the recipients defined in the EventNotificationEngine Recipients list.
 - **NFNotifyOnVerificationErrors** – Whether or not specified recipients will receive notifications when a password verification process results in an error.

- **NFOnVerificationErrorsRecipients** – An optional parameter that lists the email addresses of users who will receive notifications when a password verification process results in an error. This list overrides the recipients defined in the EventNotificationEngineRecipients list.
- **NFNotifyOnPasswordUsed** – Whether or not specified recipients will receive notifications when a password is used. These notifications will only be sent under the following conditions:
 - The password is used from the Password Vault Web Access interface
 - The user who retrieves the password must have at least one Safe authorization other than 'Manage Safe' on the Safe where the password was used.(e.g. Retrieve passwords)

The PVWA application user must be a member of the PasswordManager_Info Safe, with the following authorizations:

- Manage Safe
- Retrieve accounts

Note: If more than one CPM is installed, the PVWA application user must be a member of each of the PasswordManager_Info Safes.

- **NFOnPasswordUsedRecipients** – An optional parameter that lists the email addresses of users who will receive notifications when a password is used. This list overrides the recipients defined in the EventNotificationEngineRecipients.

For more information about these parameters, refer to the Privileged Account Security Reference Guide.

7. Click **Apply** to apply the configuration changes immediately,

or,

Click **OK** to save the changes and display the System Configuration page.

Logging

In order to monitor ENE activity and status, the following log files are created in the Event Notification Engine installation folder:

- **ENECConsole.log** – This file contains informational messages and errors that refer to ENE function. This log is meant for the system administrator who needs to monitor the status of the ENE.
- **ENETrace.log** – This file contains errors and trace messages that can be used for troubleshooting. The types of messages that are included depend on the debug levels that are specified in the EventNotificationEngine.ini configuration file. In the following parameters:
 - **ControllerDebugLevel** – This parameter defines the controller debug level.
 - **CollectorDebugLevel** – This parameter defines the collector debug level.
 - **ParserDebugLevel** – This parameter defines the parser debug level.
 - **SMTPSenderDebugLevel** – This parameter defines the SMTP sender debug level.

The amount of information written in the ENETrace.log is determined by the following trace levels:

Trace level	Indicates
1	Only exceptions will be written in the trace log.
2	Trace messages will be written in the trace log.
3	Vault connectivity errors will be written in the trace log. This trace level is only available for the ControllerDebugLevel parameter.
4	Vault connectivity debug and activity logging will be written in the trace log. This trace level is only available for the ControllerDebugLevel parameter.

New log files are created in either of the following scenarios:

- **Each time the ENE is started** – New log files are created each time the ENE is started. When the ENE stops running, the log files are timestamped and stored in the 'Event Notification Engine\Logs\Old' folder so that they do not overwrite existing log files.
- **When the log files reach 50 MB** – When the log files reach 50 MB, they are timestamped and stored in the 'Event Notification Engine\Logs\Old' folder and new log files are created.

Recording ENE Activities in the Event Viewer

In addition to the above log files, the Event log records activities that are performed by the ENE until the EventNotificationEngine parameters are retrieved from the Vault and the log files are created according to the specified parameters. This enables users to track all the activities carried out by the ENE from the moment it starts working.

In order to identify ENE components that performed activities, the following prefix is added to messages in the Event log:

- Cyber-Ark ENE

Operating the CyberArk Vault

The CyberArk Vault is installed with a minimal graphical user interface that enables you to perform various Server operations. These include starting the Server which enables Users to access the Vault, stopping the Server and viewing the Server activities log.

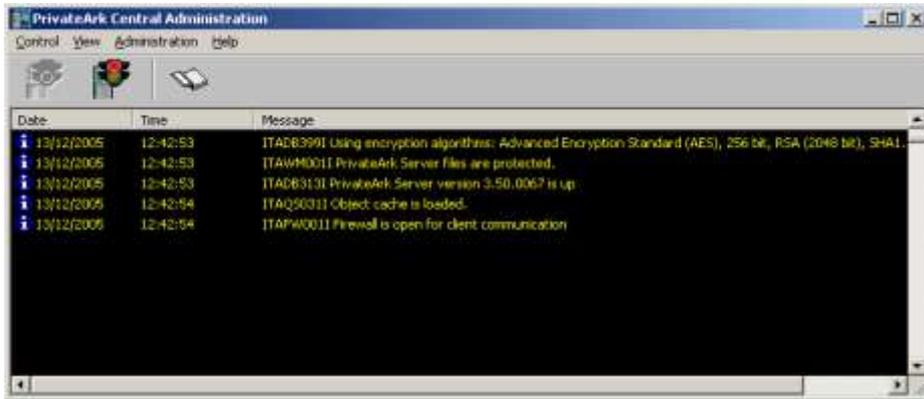
This chapter introduces you to system administration on the CyberArk Vault server machine and describes how to administrate the CyberArk Vault server.

This chapter includes the following sections:

- Working with the Server Interface
- Viewing the Server Log
- Managing the Vault Workload
- Specifying Administration Settings
- Monitoring the Vault
- Managing the CyberArk License
- Information Recovery
- CyberArk HA Vault Cluster
- CyberArk Disaster Recovery Vault

Working with the Server Interface

The Server interface can only be installed on the Server host.



It is possible to operate the Server from a remote terminal. For more information, refer to *Remote Administration for the Vault/DR Vault*, page 848.

The Server interface enables the following operations:

- Starting the Server, which then begins operating as a Windows service.
- Stopping the Server.
- Displaying the Server log.
- Administrating the Server settings, which include the Server IP address and the location of Safes stored in the Vault.

Note: The current Server log records are automatically displayed in the PrivateArk Central Administration window. To view historical log records, click the *Show Log* button in the toolbar.

Starting the Server

As a service, the Server can be configured to start automatically or manually.

The Server Key is required to start the Server. There are two ways of providing the Server Key upon startup, as follows:

- The Server Key can be permanently installed on the Server host. This enables you to configure the Server to start automatically.
- The Server Key can be stored on a removable media, such as a disk or CD. This means that the disk or CD must be in place in order to start the Server and can be removed after the Server has started.

To Start the Server Automatically

- In the Services dialog box, configure the “PrivateArk Server” service to start automatically.

Note: For the Server to start automatically, the Server Key must be permanently installed on the Server terminal.

To Start the Server Manually

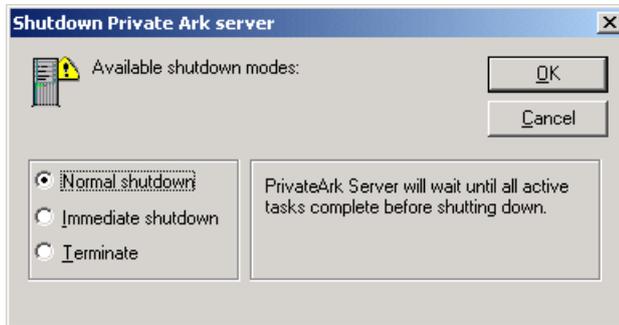
Note: Dbmain.exe is located in the Server installation directory.

- As a service:
 1. Insert the Operator CD or disk which contains the Server Key and the Public Recovery Key.
 2. In the PrivateArk Server Interface, click **Start** on the toolbar,
or,
In the Services dialog box, select the PrivateArk Server service, and click **Start**,
or,
From the command line, type:
`dbmain.exe service db`
 3. Remove the Operator CD or disk, and place it back in its safe location.
- As a console application:
 1. Insert the Operator CD or disk which contains the Server Key and the Public Recovery Key.
 2. At the command line, type: `dbmain.exe console db`
 3. Remove the Operator CD or disk, and place it back in its safe location.

The Server is generally started as a console application when troubleshooting the CyberArk Vault.

Stopping the Server

- As a service:
 1. In the PrivateArk Server interface, click **Stop** in the toolbar; the Shutdown PrivateArk Server window appears.



Type of Shutdown	Indicates ...
Normal	Wait for active tasks to complete before stopping the Vault. This is the default.
Immediate	Force active tasks to complete before stopping the Vault.
Terminate	Stop the Vault without completing active tasks.

2. Select the type of shutdown to carry out, then click **OK**; a message box appears requiring confirmation for shutdown.
 3. Click **OK** to confirm that you wish to close the PrivateArk Server; the Server will shutdown and will not process any more requests.
- Or,
1. Display the Services window, and select the PrivateArk Server service.
 2. From the **Action** menu, click **Stop**.
- As a console application:
 1. At the command line, type: `dbmain.exe console db`
 2. Press **Ctrl** and **C** together; the PrivateArk Server terminates normally.

Viewing the Server Log

The Server log displays all the activities that have been carried out in the CyberArk Vault over a specified period of time.

To Display the PrivateArk Log

- From the **View** menu, select **PrivateArk Log**; the Server generates the log and displays it.

To Find an Item in the Log

- To find a specific item in the log, from the **Options** menu, select **Find**, then **Find in log**; the Find in PrivateArk log window appears.



- In the Find what edit box, type the text to find.
- In the Filter section, select any required filter specifications.
- In the Direction section, select the direction in which to begin the search, then click **Find Next**; the Server carries out the search, and highlights the line containing the specified text.

The types of messages that are included in the server log depend on the **DebugLevel** specified in the DBParm.ini. You can set several values, separated by commas.

Trace name	Level	Indicates
PE	1	A service start and end. This is an elementary trace level which is usually activated.
PE	2	Special cases.
PE	3	Includes messages related to activities performed during the FilesList service. This trace level is specifically for FilesList performance issues.
PE	4	Includes messages related to activities performed during the AddSafe service. This trace level is specifically for timing issues related to impersonated users during an AddSafe transaction.
PE	5	Includes messages related to activities performed during atomic bundle transactions.
PE	6	Special cases.
PE	7	Special cases related to error ITATS093E.

Trace name	Level	Indicates
PE	8	Includes messages related to the UserBlock instantiation during a service execution.
PE	9	Special cases for HandleInactiveApplications.
PE	10	Includes messages related to errors that occurred during the DB locks mechanism.
PE	13	Includes messages related to ENE events which are invoked from Vault services.
PE	16	Detailed information regarding the Clear Safe History process.
UI	1	Includes messages related to the UserBlock instantiation during a service execution.
UI	2	Includes messages related to the UserBlock instantiation during a service execution.
UI	8	Includes messages related to the UserBlock instantiation during a UI action.
PE	10	Includes messages related to errors that occurred during the DB locks mechanism.
PE	13	Includes messages related to ENE events which are invoked from Vault services.
SYSLOG	1	Includes messages related to decisions made about Syslog jobs.
SYSLOG	2	The xml output will be added to the trace log for each Syslog written.
SYSLOG	3	Detailed network related messages of the syslog mechanism.
SYSLOG	4	Detailed information regarding initiated and finished syslog operations.
DM	12	Special cases which are related to Backup files (dumps & binary logs) parsing when running restore transactions via PARestore.
DM	13	The query will be written to the trace for each MySQL query.
DM	14	The bind parameters will be written to the trace for each MySQL query.
CRYPT	1	Special case for Cryptolib.
PERF	1	Detailed Vault services debug
LDAP	14	Detailed messages regarding LDAP provisioning and authentication
LDAP	15	Detailed messages regarding LDAP provisioning and authentication
COMM	1	Detailed networking related messages
COMM	2	Detailed networking related messages
COMM	3	Detailed networking related messages
COMM	4	Detailed metadata secure channel replication related messages

Trace name	Level	Indicates
CONNPOOL	1	Detailed networking related messages
CONNPOOL	2	Detailed networking related messages
AUTH	1	Detailed information regarding Logon operations
AUTH	2	Detailed information regarding Radius logon operations
INFRABLSERVICE	1	Detailed information regarding Vault services executed by the Logic Container component
INFRABLSERVICE	2	Detailed information regarding Vault services executed by the Logic Container component

Managing the Vault Workload

The Vault server serves multiple clients concurrently. The number of concurrent transactions and the priorities on allocating Vault tasks can be controlled and configured by relevant parameters.

Controlling Vault Concurrency Level

The maximum number of concurrent tasks is controlled by the **TasksCount** parameter in the DBParm.ini. The default value for this parameter is 20.

Customers can increase this number in implementations where strong hardware or multiple processors are installed, or in cases of mass file transfer activities.

Note: Modifying this parameter may have a significant effect on the Vault response time. Any change must be followed by careful monitoring.

Controlling Vault Tasks Allocation

The Vault allows users to define how Vault server tasks are allocated between different Vault interfaces and users, resulting in optimized functionality.

The following parameters in DBParm.ini enable you to control the Vault task allocation process.

- **DedicatedTasksAllocation** – Defines how many concurrent Vault transactions are dedicated to specified Vault interface IDs.
- **MaxTasksAllocation** – Defines the maximum number of concurrent Vault transactions that can be used for specified Vault interface IDs.
- **DedicatedTasksAllocationByUser** – Defines how many concurrent Vault transactions are dedicated to specified users.
- **MaxTasksAllocationByUser** – Defines the maximum number of concurrent Vault transactions that can be used for specific users.

These parameters use the following syntax:

<Number of Vault tasks>(<Vault interface IDs/users>):FromHour-ToHour

This syntax is described below in detail:

Syntax	Description
Number of Vault tasks	The number of concurrent Vault server transactions that will be assigned to the specified Vault interface IDs or users. The overall number of concurrent tasks that the Vault can manage is defined in the TasksCount parameter, also in DBParm.ini. If the total number of Vault tasks specified in these parameters does not reach the number of concurrent tasks specified in the TasksCount parameter, these tasks can be used by any Vault interface or user.
Vault interface ID/User	The Vault interface ID that will be handled by the Vault server, or the name of the user whose tasks will be handled by the Vault server. Vault interface IDs and users can be specified in more than one workload management definition. If more than one Vault interface ID or user is specified in the same workload management definition, the number of concurrent transactions is shared between the specified Vault interface IDs or users. In addition to Vault interface IDs and users, you can also define tasks that will be allocated for internal functions.
FromHour and ToHour	The time from when and until when the Vault server will apply the specified workload management. These hours are specified using 0 (zero) to indicate midnight through to 23 to indicate 11pm. You can specify workload management processes that begin on one day and end during the next day. For example, 20-8 indicates that the process will begin at 8pm and will continue until 8am the following morning.

The following examples show how each parameter can be specified to define Vault workloads:

Example 1 – Allocating dedicated tasks

The following example assures that five tasks out of the total number of specified concurrent tasks will be dedicated to online user access through the WINCLIENT/PVWA interfaces. This definition is valid between 8am and 6pm.

```
DedicatedTasksAllocation=5 (WINCLIENT, PVWA) : 8-18
```

Example 2 – Allocating maximum tasks

The following example determines that between 8am and 8pm, a maximum number of eighteen tasks will be allocated for batch CPM processing.

```
MaxTasksAllocation=18 (CPM) : 8-20
```

Example 3 – Allocating dedicated tasks per user

The following example determines that between 8am and 5pm, one task is dedicated for the user called Paul. The rest of the tasks can be used by any other users.

```
DedicatedTasksAllocationByUser=1 (Paul) : 8-17
```

Example 4 – Allocating maximum tasks per user

The following example determines that between 7am and 7pm, a maximum number of five concurrent tasks will be allocated to Paul and Lisa.

```
MaxTasksAllocationByUser=5 (paul, lisa) : 7-19
```

Example 5 – Allocating tasks for internal functions

The following example shows how tasks can be dedicated to internal functions such as clearing Safe history. This example reserves 3 concurrent tasks at any time for internal functions.

```
DedicatedTasksAllocationByUser=3(Internal)
```

Managing Long Transactions

The CyberArk Vault can be configured to automatically identify and manage Vault database transactions that have been running for an extended period of time, using the **MonitorLongTransactions** parameter in DBParm.ini.

This parameter sets policies that determine how long-running transactions are managed. These policies include a user type or database query type that identifies the transactions to monitor, and a series of thresholds that determine how these transactions are monitored and whether they are managed automatically or manually.

Managing Long Transactions Automatically

The MonitorLongTransactions parameter defines a series of policies for different types of database transactions and users who perform them. These policies determine when alerts are written in the Vault log file and can also terminate transactions that have been running for a certain length of time. The order of the policies determines the priorities that are applied during monitoring. All parameters are mandatory.

This parameter uses the following information to set policies:

```
MonitorLongTransactions=<UserType/QueryPattern>,<Message  
threshold>,<Repeated message threshold>,<Terminate transaction  
threshold>
```

These values are described in the table below:

Value	Determines ...
UserType/ QueryPattern	<p>Determines how the platform will identify transactions. Specify either a specific user type or a regular expression that identifies a database transaction.</p> <ul style="list-style-type: none"> To set a user type, specify a 'U' prefix before specifying the name of the user type in quotation marks. For example, U"EPVUser". To set a valid regular expression of a database transaction, specify an 'R' prefix before specifying the regular expression in quotation marks. For example, R"SELECT".
Message threshold	<p>The number of seconds that a monitored transaction must run for it to be considered an exception to the platform. An alert is written in the italog file for transactions that are alive for longer than the specified number of seconds.</p> <p>Specify '-1' to disable monitoring for the user type or query pattern in this platform, and to cancel the other parameters specified in this platform.</p>

Value	Determines ...
Repeated message threshold	The frequency in seconds that additional alerts will be written in the italog file to indicate that a monitored transaction is still alive. Specify a value greater than 60 seconds. Specify '-1' to display messages according to the number of seconds specified for the message threshold.
Terminate transaction threshold	The number of seconds after which a 'kill' command will be executed for live transactions. Specify '-1' to prevent the kill command from being executed.

The following example shows how to set the MonitorLongTransactions parameter:

```
MonitorLongTransactions=U`built-
inadmins`,60,180,240,U`EPVUser`,100,300,-1
R`flush tables`,100,300,-1
```

The above example sets the following three policies:

- Transactions run by built-in administrators – After these transactions have been running for 60 seconds, an alert is written in the italog file. When transactions have been running for an additional 180 seconds, another alert is written in the italog file. Transactions that have been running for yet another 240 seconds, will be terminated by a kill command.
- Transactions run by EPVUser user types – After transactions that are run by EPV users have been running for 100 seconds, an alert is written in the italog file. When these transactions have been running for another 300 seconds, another alert is written in the italog file. As -1 is specified for the kill command, these transactions will not be terminated.
- Database transactions that match the 'flush tables' regular expression – After transactions that match the 'flush tables' regular expression have been running for 100 seconds, an alert is written in the italog file. When these transactions have been running for another 300 seconds, another alert is written in the italog file. As -1 is specified for the kill command, these transactions will not be terminated.

The order of these policies indicates that transactions initiated by built-in administrators will be managed according to the first platform, even if they contain 'flush tables' as a regular expression. Likewise, transactions initiated by EPVUser user types will be managed according to the second platform, even if they contain 'flush tables' as a regular expression. Only transactions that contain 'flush tables' as a regular expression and are not run by built-in administrators or EPVUser user types will be managed according to the third platform.

By default, the following platform is set in the DBParm.ini:

```
MonitorLongTransactions=U`EPVUser`,120,600,-1
```

This platform ensures that all transactions initiated by EPVUser users that have been running for 120 seconds, will be written in the italog file, and another alert will be written in the italog file after these transactions have been running for 600 seconds. These transactions will not be terminated automatically.

For information about terminating transactions manually, refer to *Managing Long Transactions Manually*, below.

Managing Long Transactions Manually

The `CAVaultManager` utility enables you to manually terminate transactions that have been running for a long time, using the `CAVaultManager TerminateDBTransaction` command.

This command terminates a specific transaction. If the transaction is no longer alive, an error will be issued.

CAVaultManager TerminateDBTransaction

`CAVaultManager` uses the `TerminateDBTransaction` parameter to provide the following option:

```
CAVaultManager  TerminateDBTransaction
                  /DBTransactionID
```

This usage is explained in the following table and example:

Option	Description
<code>/DBTransactionID</code>	The unique transaction ID of the long transaction. This ID appears in the alert message that is written in the italog file when the transaction is identified by the <code>MonitorLongTransactions</code> parameter in <code>DBParm.ini</code> .
<code>/?</code>	Displays the list of options available with this utility.

To Terminate a Transaction

1. In the italog file, in the alert that refers to the long-running transaction, copy the transaction's unique ID.

For example, the following alert in the italog file refers to a transaction whose unique ID is 24.

```
19/01/2011 11:37:24 ITADB484S DB process 24 Exceeded regular
expression flush tables time policy (running 109 seconds).
```

2. Run the `CAVaultManager.exe TerminateDBTransaction` command. If you do not specify the transaction's unique ID in the command, you will be prompted for it, as shown in the following example:

```
CAVaultManager.exe TerminateDBTransaction
Database Transaction ID [mandatory] ==> 24
```

3. This command will identify the transaction and terminate the transaction, then display the following confirmation:

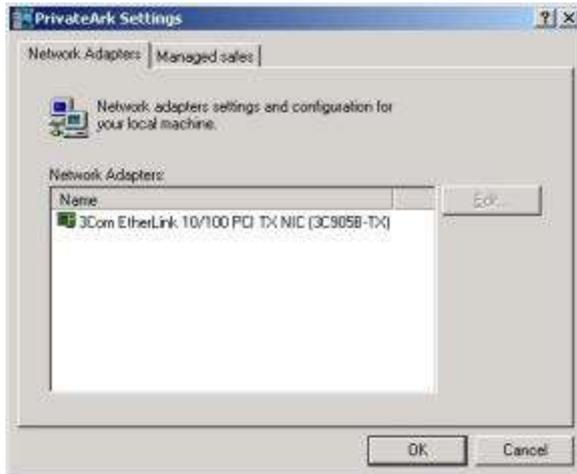
```
CAVLT212I Transaction was killed successfully.
```

Specifying Administration Settings

The Settings window enables you to view and modify the network adapter configurations and the Safe directories used by the CyberArk Vault.

To View the Administration Settings

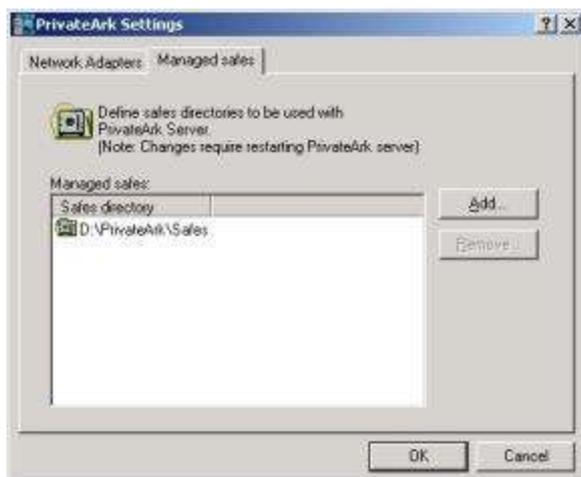
1. From the **Administration** menu, select **PrivateArk Settings**; the PrivateArk Settings window appears.
2. In the **Network Adapters** tab, you can view a list of your Network Adapters.



3. To configure an adapter, select it, then click **Edit**; the Set Network Adapter configuration window appears.



4. Specify the Adapter type, then click **OK**.
5. Click the **Managed safes** tab to display the directories in which Safes are stored. Safe directories that have been erased appear in red.



6. To add a Safe directory, click **Add**; the Browse window appears enabling you to select a directory in which to store more Safes.
7. To remove a Safe directory, select the directory to remove, then click **Remove**.
8. Click **OK** to close the PrivateArk Settings window.
9. Restart the Server for the changes to take effect.

Accessing the Vault through a Gateway Account

A user, defined as a Gateway account, enables users to access privileged accounts and files through an application, such as the PVWA. Users who are logged onto the application can access any Safe that is shared with the gateway account.

A gateway account enables the following types of impersonation:

- **Provide full impersonation** – Enables an application to log a user onto the Vault after authenticating to a third party server, such as RSA SecurID or LDAP. For example, users trying to access the Vault are automatically authenticated to a third party server with no manual intervention. This impersonation is transparent to the user.
- **Provide partial impersonation** – Enables an application to allow another application to access the Vault. For example, the PVWA sends a partial impersonation token to the PSM which uses it to access the Vault.
- **Provide impersonation with additional Server authentication** – Enables an application to log a user onto the Vault after authenticating to a third party server as well as authenticating to the Vault with a username and password. For example, users trying to access the Vault are automatically authenticated to a third party server with no manual intervention, and are then required to provide their Vault username and password.

Monitoring the Vault

Remote Administration for the Vault/DR Vault

The CyberArk Vault Remote Control feature enables users to carry out several operations on Vault components from a remote terminal. It comprises two elements – the Remote Control Agent and the Remote Control Client. The Agent is installed as part of the Vault component, on the Server and the Disaster Recovery Server.

Configuring the Remote Control Agent Manually

The Remote Control Agent can be configured automatically during installation. If, however, you did not configure it during installation, you can configure it afterwards using the following instructions:

1. After Vault installation, in the agent parameter file in the Server installation folder fill in the following parameters:

- **RemoteStationIPAddress** – The IP address of the remote computer from where instructions will be received to carry out operations on the Vault. You can specify up to 3 IP addresses, separated with commas. For example,
`RemoteStationIPAddress=1.1.1.220,1.1.1.225,1.1.1.250`
- **RemoteAdminPort** – The port through which the remote control instructions will be received. This port number must be different from the Vault's defined open ports. The CyberArk port for Remote Control is 9022.
- **ExtensionComponentList** – The full pathname of the Vault component DLL file that the remote agent will load. This file is in the Vault installation folder. For example,

```
ExtensionComponentList="C:\Program Files
(x86)\PrivateArk\Server\PARVaultAgent.dll,C:\Program Files
(x86)\PrivateArk\Server\PARENEAgent.dll"
```

For more information, refer to *Remote Control Agent Parameter File* in the Privileged Account Security Reference Guide.

2. Create the password for the agent.
 - i From the Vault server installation folder, run the following command to create the password for the agent and encrypt it.


```
Paragent setpassword
```

 The encrypted password is stored automatically in the location specified by the `UserCredentialsPath` parameter in the agent parameter file.
 - ii In the EPV-Internal Safe, create a password object called **RCAgent** and store the agent password in it for safekeeping.
3. From the Control Panel, display the available services, and start the **PrivateArk Remote Control Agent** service. Change it from a manual service to automatic.

Configuring the Remote Control Client

The Remote Control Client is a utility that runs from a command line interface and carries out tasks on a Vault component where the Remote Control Agent is installed. It does not require any Vault components to be installed on the same computer, including a PrivateArk Client.

The Remote Control Client can work with several agents, providing controlled flexibility and streamlined administration facilities.

The Remote Control utility is called PARClient. Its details and usage are listed below.

```
PARClient <remote machine[/password]>
          [/Port <port number>]
          [/CreatePassFile <file name>]
          [/UsePassFile <file name>]
          [/StateFileName <file name>]
          [/Q]
          [/C "command"]
          /?
```

Parameter	Description
<remote machine [/password]>	The name or IP address of the remote machine where the Remote Control Agent is installed and the Agent's password is stored. The password exists as an encrypted password in a separate file that is referenced in the RemoteUserCredentials parameter in PARAgent.ini.
/Port <port number>	The port number from where the remote machine will receive control commands. The default is 9022.
/CreatePassFile <file name>	Creates a file that will contain an encrypted password. This can replace the '/password' parameter and is a more secure way of storing the password.
/UsePassFile <file name>	Use the file that contains the encrypted password.
/StateFileName <file name>	Use when several users work simultaneously. The file contains internal information about each user's session. Without this switch, the client uses a default state file, called parclient.dat.
/Q	Quiet mode. The utility does not ask for confirmations before carrying out an operation.
/C	Run a single command and exit. If a command is specified with a space, use "" (quotation marks) around the command. This is shown in the examples below.
/?	Lists the available options.

The Remote Control utility can be used for administrative tasks on the Password Vault and the DR Vault. It contains several general commands that are the same for each component. In these commands, use the following names to specify each component:

Component	Name
Vault server	Vault
Disaster Recovery Vault	PADR
Event Notification Engine	ENE

In addition, the Remote Control utility contains commands that are component-specific.

Managing the Vault, DR Vault, and ENE from a Remote Location

The following table displays the commands that can be used with the PARClient utility to manage the Vault, DR Vault and ENE from a remote physical location.

Note: When the Start, Stop, and Status commands are used to control a CyberArk Vault Cluster, all nodes in the cluster are affected.

In a clustered environment, the Microsoft Cluster automatically starts cluster resource services that have stopped. To control the CyberArk services, use the Microsoft Cluster Administrator tool and not the Remote Control Client.

Parameter	Description
General local commands:	
Help <component>	Show all the available commands, or extended Vault help.
Exit	Exit the remote administration session.
Vault management commands:	
List <component>	Lists manageable Vaults on local and remote machines.
[/Modules]	Supplement information about all the Vault modules.
[/PARComponents]	Lists remote control components on local and remote machines.
[/Local]	Lists only local components.
[/NoVersion]	List all components without version information.
With Vault	Set the default Vault for next commands.
Reboot	Reboots the remote machine.
Vault commands:	
Start Vault	Start a Vault on the remote machine.
/Last	Starts the Vault with the last known good configuration files.
Stop Vault	Stop a Vault on the remote machine.
/Normal	Wait for active tasks to complete before stopping the Vault. This is the default.
/Immediate	Force active tasks to complete before stopping the Vault.
/Terminate	Stop the Vault without completing active tasks.
Restart Vault	Restarts a Vault on the remote machine.

Parameter	Description
Status Vault	Show activity status of a Vault on the remote machine.
GetLog Vault	Show the Vault log messages on remote machine. When this command is used without the /TimeFrom option, all unread messages will be retrieved.
/TimeFrom <ddmmyyy:hhmm>	The date and time to display records from.
/Lines <number>	Maximum number of log records to display.
/LastOffsetRead	Offset in the log file where the last successful read process was completed.
/LastTimeRead	The timestamp of the last successful read operation.
GetParm Vault	Retrieves a configuration parameter from the specified component.
<parm name>	The name of the parameter to retrieve. This may be one of the following: DefaultTimeout, MTU, SecurityNotification, DebugLevel, DisableExceptionHandling
[/Group <group>]	The name of the group.
[/SubComponent <subcomponent>]	The name of the subcomponent.
SetParm Vault	Sets a configuration parameter on the Vault for one of the following parameters: DefaultTimeout, MTU, SecurityNotification, DebugLevel, DisableExceptionHandling Note: In a high-availability cluster environment, this parameter will only change the configuration parameter on the active node.
<parm name>= <parm value>	Parameter name and value to set.
[/Group <group>]	Optional logical group name.
[/SubComponent [subcomponent]	Optional subcomponent name.
[/Temporary /Permanent /Immediate]	The parameter update will be effective according to one of the following options (respectively): <ul style="list-style-type: none"> ◆ until the Vault restarts ◆ after the Vault restarts ◆ immediately
[/Default]	Set default value to parameter.
Do Vault "command"	Execute an extended Vault command on the remote machine.
DR Vault commands:	
Start PADR	Start a DR Vault on the remote machine.
Stop PADR	Stop a DR Vault on the remote machine.
Restart PADR	Restarts a DR Vault on the remote machine.
Status PADR	Show activity status of a DR Vault on the remote machine.

Parameter	Description
GetLog PADR	Shows the Disaster Recovery Vault log file, PADR.log, on the remote machine.
/TimeFrom <ddmmyyy:hhmm>	The date and time to display records from.
GetParm PADR	Retrieves a configuration parameter from the DR Vault.
<parm name>	The name of the parameter to retrieve. This may be one of the following: EnableCheck, EnableReplicate, EnableFailover, EnableDBsync, FailoverMode For more information about the above parameters, refer to <i>DBParm.ini</i> in the Privileged Account Security Reference Guide.
SetParm PADR	Sets a configuration parameter on the Vault for one of the following parameters: DefaultTimeout, MTU, SecurityNotification, Debug, DebugLevel, DisableExceptionHandling Note: In a high-availability cluster environment, this parameter will only change the configuration parameter on the active node.
<parm name>	Sets a configuration parameter on the Disaster Recovery Vault for one of the following parameters: EnableCheck, EnableReplicate, EnableFailover, EnableDBsync, FailoverMode
/permanent	The configuration parameter specified in SetParm will take effect after the component is restarted.
Do PADR "command"	Execute an extended DR Vault command on the remote machine.
ENE management commands:	
List <component>	Lists manageable ENEs on local and remote machines.
[/Modules]	Supplement information about all the ENE modules.
[/PARComponents]	Lists remote control components on local and remote machines.
[/Local]	Lists only local components.
[/NoVersion]	List all components without version information.
ENE commands:	
Start ENE	Start the ENE service.
Stop ENE	Stop the ENE service. Before stopping, the ENE service will send out notifications for all the activities that it has already recognized.
Status ENE	Show activity status of the ENE service on the remote machine.
GetLog ENE	Show the ENE log file on the remote machine.
/LogFile Trace/Console	Whether the ENE log file will be ENETrace.log or ENEConsole.log.

The following examples show several tasks that you can perform from a remote physical location.

Examples of Remote Administration on the Vault

The following example logs the user on to the Vault and returns its status. In this example, the password of the Remote Control Agent is 'Asdf1234'.

```
>parclient 1.1.1.250/Asdf1234 /c "status vault"
```

The following example returns the current CPU usage on the Vault.

```
>parclient 1.1.1.250/Asdf1234 /c getcpu
```

In the next example, the user sets the 'DefaultTimeout' parameter in the DBParm.ini file in the Vault.

```
>parclient 1.1.1.250/Asdf1234 /c "SetParm Vault DefaultTimeout=30 /Immediate"
```

Examples of Remote Administration on the DR Vault

The following example logs the user onto the DR Vault where they start the DR service:

```
>parclient 1.1.1.250/Asdf1234 /c "start PADR"
```

In the next example, the user returns the status of the DR Vault.

```
>parclient 1.1.1.250/Asdf1234 /c "status padr"
```

Examples of Remote Administration on the ENE

The following example logs the user onto the Vault where they start the ENE service. In this example, the password is 'Asdf1234'.

```
>parclient 1.1.1.250/Asdf1234 /c "start ENE"
```

In the next example, the user returns the current status of the ENE service.

```
>parclient 1.1.1.250/Asdf1234 /c "status ENE"
```

The above command should show you that the ENE service is active.

Monitoring the Vault and the DR Vault from a Remote Location

Remote Monitoring enables you to receive a variety of information from the Production Vault and the DR Vault from a remote machine. This information includes operating system and Vault information, as follows:

Operating System information:

- CPU, memory, and disk usage
- Event notifications
- Service status

Component-specific information:

- Password Vault and DR Vault status
- Password Vault and DR Vault logs

The following table displays the commands that can be used with the PARClient utility to retrieve this information.

Parameter	Description
GetCPU	Displays the current CPU usage percentage on the remote machine.
GetDiskUsage	Displays the free disk space in MB and in percentage on the remote machine.
ServiceStatus	Shows the activity status of a system service on the remote machine. The service must be listed in the AllowedMonitoredServices in the remote agent parameter file. This parameter accepts wildcards.
/ServiceName <service name>	The name of the service on the remote machine. These services must also be specified in the remote agent parameter file. When the service name contains a space, the name must be contained within a slash and quotation marks, as follows: \"<service name>\"
GetMemoryUsage	Shows the physical and swap memory status.
GetOSLog	Shows the operating system log messages. When this command is issued without the /TimeFrom option, only unread records will be displayed.
/Name <os log name>	The name of an existing operating system log. Valid values are: Application, Security, System.
[/TimeFrom <ddmm yyyy:hhmm>]	The date and time to display records from.

Examples:

In the following example, the user has accessed the PARClient folder through a command line prompt, and logs onto the Vault. The user specifies the password of the Remote Control Agent which, for this example, is 'Asdf1234'.

```
>parclient 1.1.1.250/Asdf1234
```

Now the user can retrieve the current CPU usage.

```
PARCLIENT>getcpu
```

The user can monitor the activity of the Event Log service on the Password Vault machine. The Event Log service must be listed in the `AllowedMonitoredServices` in the remote agent parameter file.

```
PARCLIENT>servicestatus /servicename "Event Log"
```

The user can also find out how much free disk space there is in MB on the Vault machine.

```
PARCLIENT>GetDiskUsage
```

When the user has finished monitoring the Vault activities, they can exit the Remote Control Client utility:

```
PARCLIENT>exit
```

Even if a user does not exit the utility, after the Remote Control Client has been idle for 30 seconds, the user is required to enter the agent's password again in order to carry out any commands. This maintains the security level of the Vault operators.

Configuring Remote Monitoring

The Remote Monitoring uses SNMP to send Vault traps to a remote terminal. This enables users to receive both Operating System and Vault information, as follows:

Operating System information:

- CPU, memory, and disk usage
- Event log notifications
- Service status

Component-specific information:

- Password Vault and DR Vault status
- Password Vault and DR Vault logs

CyberArk provides two MIB files (for SNMP v1 and SNMPv2) that describe the SNMP notifications that are sent by the Vault. These files can be uploaded and integrated into the enterprise monitoring software. These MIB files are included on the Privileged Account Security Installation CD:

- CYBER-ARK-MIB-V1.txt – Used to implement SNMP v1.
- CYBER-ARK-MIB-V2.txt – Used to implement SNMPv2.

To Configure Remote Monitoring

1. In the remote control agent configuration file, specify the following parameters:
 - **AllowedMonitoredServices** – The name of the system services that can be monitored from a remote location. The name of the service must be specified as it appears in the Service Name field in the Service Properties window.For example,

```
AllowedMonitoredServices="PrivateArk Database,PrivateArk Logic Container"
```

- **MonitoredEventLogNames** – The names of the event logs of activities that have taken place since the Server started, such as Application, Security, and System.
 - **SNMPHostIP** – The IP address of the remote computer where SNMP traps will be sent. Separate multiple IP addresses with a comma.
 - **SNMPTrapPort** – The port through which SNMP traps will be sent to the remote computer. You can specify either port 161 or 162. The default port is 162.
 - **SNMPTrapInterval** – The number of seconds that pass between notifications. The default value is **30**.
 - **SNMPCommunity** – The name of location where the SNMP traps originated.
 - **SNMPVersion** – The SNMP version that will be used to send SNMP notifications. Specify any of the following values:
 - v1 – The Vault will support SNMPv1 with a unique OID for each trap.
 - v2 – The Vault will support SNMPv2. This is the default value.
 - Compatibility – The Vault will send SNMP notifications using the format used in Vault versions prior to version 5.0.
2. Specify the following parameters to enable users to receive SNMP notifications.

These values comprise interval in seconds between checks and the percentage in usage that would initiate a notification, as shown in the following example:

```
SNMPTrapsThresholdCPU=30,80
```

The above example indicates that CPU usage will be checked every 30 seconds, and an SNMP trap will be sent if the usage is 80% or greater.

- **SNMPTrapsThresholdCPU** – A SNMP trap will be sent when the specified percentage of used CPU resource is reached. The default value is **200,90** and the Threshold value is optional.
 - **SNMPTrapsThresholdPhysicalMemory** – A SNMP trap will be sent when the specified percentage of used physical memory is reached. The default value is **200,90** and the Threshold value is optional.
 - **SNMPTrapsThresholdSwapMemory** – A SNMP trap will be sent when the specified percentage of used swap memory is reached. The default value is **200,90** and the Threshold value is optional.
 - **SNMPTrapsThresholdDiskUsage** – A SNMP trap will be sent when the specified percentage of used disk is reached. The default value is **200,85** and the Threshold value is optional.
 - **SNMPTrapsThresholdServiceStatus** – A SNMP trap will be sent at specified intervals when CyberArk product services or additional services that are configured using AllowedMonitoredServices parameter are down. The default value is **200**.
3. From the Control Panel, display the available services, and restart the **PrivateArk Remote Control Agent** service.

For more information about the Remote Control Agent parameter file, refer to *Remote Control Agent Parameter File* in the Privileged Account Security Reference Guide.

Notifications for High Resource Consumption

SNMP v2 traps can be configured to send notifications that indicate consistent high resource consumption, rather than each time there is a peak in usage. This includes high CPU usage, memory consumption, and disk usage. In addition, single notifications can be sent to the general state of the resource, for example, a notification is sent when resource consumption is consistently high over a predefined period of time, and then again when consumption is reduced.

Additional values in the remote configuration parameters determine how many times high consumption must be detected in order to consider it consistent.

- **Retries** – The number of times that high consumption is detected, after which it is considered consistently high.
- **RetriesIntervals** – The number of seconds between each retry.
- **State-full** – Whether or not a notification will be sent after the specified number of retries is reached and another will be sent when high consumption has been reduced and is no longer consistent.

The following example shows the values that can be added to the **SNMPTrapsThresholdCPU** parameter to send notifications for consistent high CPU usage:

```
SNMPTrapsThresholdCPU=Interval,Threshold,[Retries,RetriesIntervals,State-full]
```

- Specify the values that define how frequently notifications will be sent, as shown in the following examples:

Example 1:

```
SNMPTrapsThresholdCPU=30,80,3,30,No
```

The above example indicates that CPU usage will be checked every 30 seconds. If the usage is 80% or greater over three checks that are carried out at 30 second intervals, a notification will be sent indicating that high CPU consumption is consistent. Additional notifications that indicate high CPU consumption will be sent every three checks until the CPU usage goes down to below 80%.

Example 2:

```
SNMPTrapsThresholdCPU=30,80,3,30,Yes
```

This example is the same as the first example, except for the last value which indicates when notifications are sent. Additional checks will be performed every 30 seconds and another notification will be sent when CPU usage goes down below 80%, at least three times at 30 second intervals, to notify that the CPU usage isn't above the threshold anymore.

These values can be sent for the following parameters:

- `SNMPTrapsThresholdCPU` – By default, this is set to 200,90,3,30,YES
- `SNMPTrapsThresholdsPhysicalMemory` – By default, this is set to 200,90,3,30,YES
- `SNMPTrapsThresholdsSwapMemory` – By default, this is set to 200,90,3,30,YES
- `SNMPTrapsThresholdsDiskUsage` – By default, this is set to 200,85,3,30,YES
- `SNMPTrapsThresholdServiceStatus` – By default, this is set to 200,3,30,YES

For more information about these parameters, refer to *Configuring Remote Monitoring*, page 855.

Filtering Log Messages according to Severity

Log messages can be filtered in order to focus the notifications that you receive according to severity and reduce the number of SNMP traps that are sent through SNMP servers.

To Filter Log Messages

1. In `PARAgent.ini`, add the following parameters:
 - **LogMessagesFilterRegexp** – A list of filters that determines which messages will be sent through the SNMP server, as long as they are not listed in the `ExcludedLogMessagesFilterRegexp` parameter.
 - **ExcludedLogMessagesFilterRegexp** – A list of filters that determines which messages will not be sent through the SNMP server. This list overrides the list in `LogMessagesFilterRegexp`.

In the following example, the `LogMessagesFilterRegexp` parameter determines that all log messages will be sent through the SNMP server. However, the `ExcludedLogMessagesFilterRegexp` parameter overrides the previous parameter and determines which messages will not be sent.

```
LogMessagesFilterRegexp=.*
ExcludedLogMessagesFilterRegexp=(ITA|PARE|PADR|CAS).*I
```

2. Restart the PrivateArk Remote Control Agent service.

Monitoring Privileged Account Security Solution Components

You can monitor the components in your Privileged Account Security implementation in order to ensure that they are active. If the Vault discovers that the specified components are not active, notifications can be sent to members of the **Vault Admins** group to inform them and enable them to restore these components as soon as possible.

The Vault monitors user types for specific users and makes sure that they are actively accessing the component they are configured for. This feature is configured in the `DBParm.ini` and enabled in the PrivateArk Administrative Client.

To Configure Component Monitoring

In `DBParm.ini`, add the following parameters:

- **ComponentNotificationThreshold** – A series of values that define components to check for activity, whether or not notifications will be sent if the components are inactive, and the frequency of these notifications.

- **ComponentMonitoringInterval** – The number of minutes that will elapse between component activity checks specified in the **ComponentNotificationThreshold** parameter. The default value is **1** minute. To cancel the component activity check, specify **-1**.

The following example shows the values that can be added to the **ComponentNotificationThreshold** parameter to monitor the components:

```
ComponentNotificationThreshold=UserType, SendEmails,
InactivityPeriodNotification, ConsecutiveNotification
```

To monitor several components, specify the above parameters for each component, separated by a semi-colon, as shown below:

```
ComponentNotificationThreshold=UserType, SendEmails,
InactivityPeriodNotification, ConsecutiveNotification;UserType,
SendEmails, InactivityPeriodNotification, ConsecutiveNotification
```

The following details determine the component that is monitored and the procedure that is initiated when this procedure discovers that the component is inactive:

- **UserType** – The application user type to monitor. For example, AppProvider or CPM. For a complete list of application user types, refer to *Types of Users*, page 41.
- **SendEmails** – Whether or not a notification is sent if the component is inactive.
 - Specify **Yes** to sent notifications to members of the **Vault Admins** group, in addition to the message that is displayed in the ITALog.
 - Specify **No** to display ITALog messages.
- **InactivityPeriodNotification** – The number of minutes that the component user is inactive before notifications are sent to members of the **Vault Admins** group.
- **ConsecutiveNotification** – The number of minutes that will pass after the first notification is sent, after which the notification will be resent.

The following table lists the default monitoring values for each component:

User Type	Email Notifications	Inactivity Period Notification	Consecutive Notifications
Privileged Account Security Components:			
PIMProvider	Yes	30	1440
AppProvider	Yes	30	1440
OPMProvider	Yes	30	1440
CPM	Yes	720	1440
PVWA	Yes	90	1440
PSM	Yes	30	1440
SIM Suite Components:			
DCAUser	Yes	60	2880
SFE	Yes	10	2880
FTP	Yes	60	2880

Note: The default value specifies notification thresholds for components in both the Privileged Account Security solution and the SIM Suite. Only the thresholds for components that are relevant to your implementation will be applied.

The following example shows how this parameter can be specified to monitor the OPM and the CPM:

```
ComponentNotificationThreshold=AppProvider,yes,60,1440;CPM,no,100,2880
```

The above example configures the Vault to monitor the following components:

- **OPM** – The Vault will monitor the OPM, whose user type is **OPMProvider**, and will display an ITALog message and send email notifications to members of the **Vault Admins** group when it detects that the OPM has been inactive for at least **60** minutes. After intervals of **1440** minutes (the equivalent of one day), the message will be displayed again and notifications will be sent again.
- **CPM** – The Vault will monitor the CPM, whose user type is **CPM**, and will display an ITALog message when it detects that the CPM has been inactive for at least **100** minutes, but will not send any email notifications. After intervals of **2880** minutes (the equivalent of two days), ITALog messages will be displayed again.

To Enable Component Monitoring

1. Log onto the PrivateArk Administrative Client as a Vault administrator.
2. From the **Tools** menu, select **Administrative Tools**, and then **Users and Groups**; the Users and Groups window appears.
3. Select the user to monitor, then click **Update**; the Update User window appears.

The screenshot shows the 'Update User: PVWAAppUser' dialog box. It has four tabs: 'Time Limitations', 'Personal details', 'Phone/Notes', and 'Business/Internet'. The 'General' sub-tab is selected, showing fields for 'User Name' (PVWAAppUser), 'User type' (PVWA), and 'Location'. There are checkboxes for 'Gateway Account', 'Provide full impersonation', 'Provide partial impersonation', 'Provide impersonation with additional Server authentication', and 'Disable User'. A 'Quota' section has an 'Enable Quota' checkbox and a 'Size' field (0 MB). A 'Monitoring' section at the bottom has a checkbox for 'Send email notification if component is not connected'. The dialog also includes 'Authorized Interfaces...', 'Browse...', and 'Replace Photo...' buttons, and 'OK' and 'Cancel' buttons at the bottom.

4. In the General tab, make sure that this user's user type accesses the component you intend to monitor.
5. Select **Send email notification if component is not connected**, then click **OK**; when the selected user cannot access the component it is configured for, email notifications will be sent according to the **ComponentNotificationThreshold** parameter in DBParm.ini.

Sending Trace Data to the System Log File

The Privileged Account Security solution can be configured to write trace data in the PARAgent.log file. This file can be used to troubleshoot problems with the remote notification mechanism.

To Configure the Vault to Log Trace Data

1. In PARAgent.ini, add the following parameter:
 - **EnableTrace** – Determines whether or not trace data will be written in the PARAgent.log file. To configure the system to write trace data in this log file, specify **Yes**, as shown in the following example:

```
EnableTrace=Yes
```

2. Restart the PrivateArk Remote Control Agent service.

Monitoring the CyberArk Firewall

The **MonitorFWRulesInterval** parameter in DBParm.ini activates a monitoring process that checks the firewall for rules that have been made directly, and not through DBParm.ini. All detected rules are reported to the ITALog.

To Activate the Firewall Monitoring Process

- In DBParm.ini, add the MonitorFWRulesInterval parameter. By default, the firewall is checked every 15 minutes. Specify -1 to disable monitoring.

When the Vault server is started, the initial firewall check is performed by the server. Subsequent checks are performed by this monitoring process.

Monitoring Backup and DR Replications

The Vault can be configured to send email notifications when backup or DR replications are not completed successfully, after a predefined time period. By default, these notifications are sent to the members of the Vault Admins group, although they can be sent to any predefined recipients. In addition, a relevant message will be written in the ITALog.

These notifications are configured by the **BackupNotificationThreshold** parameter and the **DRNotificationThreshold** parameter which specify the following values:

- Whether or not to monitor backup and DR replications.
- Whether or not a notification will be sent whenever a backup or DR replication is missed or fails, after the predefined period of time has passed.
- The time after the backup or DR replication is not performed that a notification is sent. This value can be specified in either hours (default) or minutes (by specifying 'm' after the time period).
- The time after the first notification that a second and consecutive notifications will be sent. This value can be specified in either hours or minutes.
- How frequently the backup and DR status will be checked. This value can be specified in either hours or minutes. By default, this value is set to one quarter of the time after the backup or DR is not performed that a notification is sent.

The following example shows the default values for the **BackupNotificationThreshold** parameter:

```
BackupNotificationThreshold=Yes, Yes, 48, 24, 12
```

The above parameter configures the Vault to monitor missing replication and to send notifications whenever a missing replication is detected according to the following time frames. The first notification will be sent 48 hours after the missing procedure is detected, and subsequent notifications will be sent every 24 hours after that. The backup replication status will be checked every 12 hours.

The following sample shows the default values for the **DRNotificationThreshold** parameter:

```
DRNotificationThreshold=Yes, Yes, 2, 24, 30m
```

The above parameter configures the Vault to monitor missing DR replications and to send notifications whenever a missing replication is detected according to the following time frames. The first notification will be sent 2 hours after the missing procedure is detected, and subsequent notifications will be sent every 24 hours after that. The DR replication status will be checked every 30 minutes.

To Activate the Backup Status Notification

- In DBParm.ini, add the following parameters as described above:
 - BackupNotificationThreshold
 - DRNotificationThreshold

To Deactivate the Backup Status Notification

- In DBParm.ini, add the following parameters:
 - BackupNotificationThreshold=No
 - DRNotificationThreshold=No

Integrating with SIEM Applications

CyberArk can integrate with SIEM to send audit logs through the Syslog protocol and create a complete audit picture of privileged account activities in the enterprise SIEM solution. These audit logs include user and Safe activities in the Vault, which are transferred by the Vault to SIEM applications, such as HP ArcSight and RSA enVision.

CyberArk's flexible configuration enables you to define the target Syslog server, specify dynamic format translators, and to filter the events that will be sent.

Syslog messages can be sent to multiple syslog servers in two different ways:

- A single message can be sent to multiple servers by configuring a single XSLT file.
- Multiple messages can be sent to different Syslog servers and formatted differently for each server by configuring multiple XSLT files, formats and code-message lists. The code-message lists must be matched, meaning they must contain the same number of items in the same order.

To Configure SIEM Integration

- In DBParm.ini, configure the following parameters:
 - SyslogServerIP** – The IP address(es) of the Syslog servers where messages will be sent. Specify multiple values with commas.
 - SyslogServerProtocol** – Specifies the Syslog protocol that will be used to send audit logs. Specify either **TCP** or **UDP**. The default value is **UDP**.
 - SyslogServerPort** – The port used to connect to the Syslog server. The default value is **514**.
 - SyslogMessageCodeFilter** – Defines which message codes will be sent from the Vault to the SIEM application through Syslog protocol. You can specify message numbers and/or ranges of numbers, separated by commas. For example, to specify messages 1,2,3,30 and 5-10, specify the following value: 1,2,3,5-10,30. Specify multiple values with pipelines. By default, all message codes are sent for user and Safe activities. For a list of messages and codes, refer to the Privileged Account Security Reference Guide.
 - SyslogTranslatorFile** – Specifies the XSL file used to parse CyberArk audit records data into Syslog protocol. Specify multiple values with commas.
 - DebugLevel** – Determines the level of debug messages. To include Syslog xml messages in the trace file, specify **SYSLOG(2)**.
 - UseLegacySyslogFormat** - Controls the format of the syslog message, and defines whether it will be sent in a newer syslog format (RFC 5424) or in a legacy format. The default value is **No**, which enables working with the newer syslog format. Specify multiple values with commas.

The following example shows a set of syslog properties that will send a single message to multiple syslog servers.

```
SysLogServerIP=192.168.1.1,192.168.2.2,192.168.3.3
SysLogTranslatorFile=Syslog\Arcsight.sample.xsl
UseLegacySyslogFormat=yes
SyslogMessageCodeFilter=7,8,295
```

The following example shows a set of syslog properties that will send different syslog messages to multiple syslog servers.

```
SysLogServerIP=192.168.1.1,192.168.2.2,192.168.3.3
SysLogTranslatorFile=Syslog\Arcsight.sample.xsl,Syslog\QRadar.xsl,
Syslog\Arcsight.sample.xsl
UseLegacySyslogFormat=yes,no,yes
SyslogMessageCodeFilter=7,8,295|295-296|7
```

- Copy the relevant XSL translator file from the Syslog subfolder of the Server installation folder to the location specified in the SyslogTranslatorFile parameter in DBParm.ini.

During Vault installation or upgrade, the following sample XSL files are copied to the PrivateArk\Server\syslog folder:

- RSAenVision.sample.xsl
- Arcsight.sample.xsl
- QRadar.sample.xsl
- McAfeeESM.sample.xsl

- Open the XSL translator file that you will use, and make any changes that are relevant to your SIEM implementation. For information about customizing an XSL translator file, refer to *Creating a Custom XSL Translator File*, page 864.

The following table lists the most useful events to monitor:

Code	Action
4	User Authentication
22	CPM Verify Password
24	CPM Change Password
31	CPM Reconcile Password
38	CPM Verify Password Failed
57	CPM Change Password Failed
60	CPM Reconcile Password Failed
130	CPM Disable Password
295	Retrieve Password succeeded
300	PSM Connect
302	PSM Disconnect
308	Use Password
319	Retrieve Password (from Provider)
344	Privileged Command Initiated
346	Privileged Command Completed
359	SSH Command
361	PSM Command
378	PSM Secure Connect Session Start
380	PSM Secure Connect Session End
411	PSM Window Title

For a complete list of events and action codes available in the User and Safe Activities (LogList) report that can be exported to a SIEM solution using Syslog protocol, refer to the Privileged Account Security Reference Guide.

Creating a Custom XSL Translator File

In order to control the format of syslog messages generated by the Vault, an XSL translator file can be created and applied. The translator receives the XML stream that is generated by the Vault and creates a syslog output record.

The following examples show the difference between the output XML stream directly from the Vault, and the XSL translator file that changes this information into a syslog output record. A description of each field follows the examples.

The first example, below, shows an output XML generated by the Vault:

```

1. <?xml version="1.0" encoding="UTF-8" standalone="no" ?>
2. <syslog>
3.   <audit_record>
4.     <Rfc5424>yes</Rfc5424>
5.     <Timestamp>Sep 09 11:44:21</Timestamp>
6.     <IsoTimestamp>2013-09-25T11:44:21Z</IsoTimestamp>
7.     <Hostname>MYCOMP</Hostname>
8.     <Vendor>CyberArk</Vendor>
9.     <Product>Vault</Product>
10.    <Version>8.1</Version>
11.    <MessageID>4</MessageID>
12.    <Desc>Authentication failed</Desc>
13.    <Severity>Error</Severity>
14.    <OSUser>ANFIELD\System</OSUser>
15.    <Issuer>rans</Issuer>
16.    <Action>Logon</Action>
17.    <SourceUser></SourceUser>
18.    <TargetUser></TargetUser>
19.    <Safe></Safe>
20.    <File></File>
21.    <Station></Station>
22.    <Location></Location>
23.    <Category></Category>
24.    <RequestId></RequestId>
25.    <Reason></Reason>
26.    <PvwaDetails>
27.      <RequestReason>
28.        <General>
29.          <UserReason>I Need to Update this file</UserReason>
30.        </General>
31.        <ConnectionDetails>
32.          <ConnectionAddress>1.1.1.196</ConnectionAddress>
33.          <RemoteMachine>1.1.1.120</RemoteMachine>
34.          <PSMRemoteMachine>1.1.1.120</PSMRemoteMachine>
35.          <ConnectClient>SSH</ConnectClient>
36.        </ConnectionDetails>
37.        <AdditionalInformation>
38.          <Emergency>Emergency string</Emergency>
39.          <TicketId>1122</TicketId>
40.        </AdditionalInformation>
41.        <RequestDetails>
42.          <From>6/23/2013 8:00:00 AM</From>
43.          <To>6/25/2013 5:00:00 PM</To>
44.          <TimeZone>(GMT+2.00)</TimeZone>
45.          <Type>Single</Type>
46.        </RequestDetails>
47.      </RequestReason>
48.    </PvwaDetails>
49.    <ExtraDetails></ExtraDetails>
50.    <Message></Message>
51.    <GatewayStation></GatewayStation> //I3293
52.    <CAProperties>
53.      <CAProperty Name="UserName" Value="PSMConnect"/>
54.      <CAProperty Name="Address" Value="10.0.1.12"/>
55.      <CAProperty Name="LogonDomain" Value="COMPONENTS"/>
56.    </CAProperties>
57.  </audit_record>
58. </syslog>

```

The second example, below, shows an XSL translator that transforms the XML stream sent by the Vault into an HP ArcSight CEF style entry.

```

1. <?xml version="1.0" encoding="ISO-8859-1"?>
2. <xsl:stylesheet version="1.0"
   xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
3. <xsl:import href='./Syslog/RFC5424Changes.xsl' />
4. <xsl:output method='text' version='1.0' encoding='UTF-8'
   indent='yes' />
5. <xsl:template match="/">
6. <xsl:apply-imports />
7. <xsl:for-each select="syslog/audit_record">
8.     CEF:0|
9.     <xsl:value-of select="Vendor"/>|
10.    <xsl:value-of select="Product"/>|
11.    <xsl:value-of select="Version"/>|
12.    <xsl:value-of select="MessageID"/>|
13.    <xsl:value-of select="Desc"/>|
14.    <xsl:choose>
15.        <xsl:when test="Severity='Critical'">10</xsl:when>
16.        <xsl:when test="Severity='Error'">7</xsl:when>
17.        <xsl:when test="Severity='Info'">5</xsl:when>
18.        <xsl:otherwise>0</xsl:otherwise>
19.    </xsl:choose>
20.    <!--xsl:value-of select="Severity"/>|
21.    suser=<xsl:value-of select="Issuer"/>
22.    act=<xsl:value-of select="Action"/>
23.    duser=<xsl:value-of select="SourceUser"/>
24.    fname=<xsl:value-of select="File"/>
25.    src=<xsl:value-of select="Station"/>
26.    msg=<xsl:value-of select="TargetUser"/>,
27.    <xsl:value-of select="Safe"/>,
28.    <xsl:value-of select="Location"/>,
29.    <xsl:value-of select="Category"/>,
30.    <xsl:value-of select="RequestId"/>,
31.    <xsl:value-of select="Note"/>,
32.    <xsl:value-of select="Reason"/>,
33.    <xsl:value-of select="ExtraDetails"/>,
34.    <xsl:value-of select="Message"/>
35.    <!--xsl:value-of select="OSUser"/>
36.    </xsl:for-each>
37. </xsl:template>
38. </xsl:stylesheet>

```

The following table describes each field displayed in the above examples:

Field	Description
Rfc5424	Whether the syslog format complies with RFC5424.
Timestamp	The timestamp, in MMM DD HH:MM:SS format. For example: Jun 25 10:47:19.
IsoTimestamp	The timestamp, in ISO Timestamp format (RFC 3339). For example: 2013-6-25T10:47:19Z.
Hostname	The hostname, in upper case. For example: MY-COMPUTER.
Vendor	A static value that represents the vendor.
Product	A static value that represents the product.
Version	A static value that represents the version of the Vault.
MessageID	The code ID of the audit records.
Desc	A static value that displays a description of the audit codes.

Field	Description
Severity	The severity of the audit records. This is either 'error' or 'info'.
Issuer	The Vault user who wrote the audit. This is usually the user who performed the operation.
Action	A description of the audit record.
SourceUser	The name of the Vault user who performed the operation.
TargetUser	The name of the Vault user on which the operation was performed.
Safe	The name of the target Safe.
File	The name of the target file.
Station	The IP from where the operation was performed. For PVWA sessions, this will be the real client machine IP.
Location	The target Location (for Location operations).
Category	The category name (for category-related operations).
RequestId	The unique ID of the dual control request (for dual control related audit records).
Reason	The reason entered by the user.
PwaDetails	Specific details of the PVWA audit records.
ExtraDetails	Specific extra details of the audit records.
Message	A description of the audit records (same information as in the Desc field).
GatewayStation	The IP of the web application machine (PVWA).
CAProperties	Account metadata.

Troubleshooting

Server trace files are stored in the server installation folder for reference during troubleshooting. The server stores four trace files in this folder, and each time the size of the current trace file reaches 200MB, it replaces the oldest one.

The **TraceArchiveMaxSize** parameter in DBParm.ini enables you to customize the number of trace files by determining the maximum size of a trace file archive folder, essentially determining how many trace files will be saved. When a trace file reaches 200MB, it is renamed, timestamped, and stored in the local **<server>/Archive/** folder. Files are renamed according to the following format:

Arc-yyyyMMdd-hhmmss.log

When the size of all the trace files in the folder collectively reaches the specified maximum size, the oldest file in the archive folder is deleted. The default value is **5120MB**. In order to archive more than one trace file, specify a maximum size greater than 200MB.

To disable trace file archiving, specify **-1**.

Managing the CyberArk License

The license that you will receive before you install the Vault server determines how many users, passwords, and files you can store in the Vault. In addition, it determines groups of user types and the different interfaces that each type can use.

By default, the Vault issues a warning one week before the license expires, and every day after that until the license expires. The Vault can also be configured to issue notifications when predetermined percentages of licensed users have been created.

Monitoring the User License

The `LicenseUsageAlertLevel` parameter in `DBParm.ini` determines when notifications will be sent to predefined recipients with information about license usage

To Configure License Monitoring

- In `DBParm.ini`, set the `LicenseUsageAlertLevel` parameter.

This parameter defines three thresholds for license usage percentage, which is determined by the number of users of each user type that are defined in the Vault. When the number of licensed users reaches the specified percentage thresholds, notifications are sent to predefined recipients.

When the first percentage threshold is reached, a notification is sent to recipients, and likewise when the second specified threshold is reached. When the third threshold is exceeded, a notification will be sent each time a new user is added.

The following example shows the default setting for the `LicenseUsageAlertLevel` parameter:

```
LicenseUsageAlertLevel=85, 90, 99
```

Using the above example, a notification will be issued when 85% of the maximum number of licensed users has been added to the Vault, and another notification will be sent when 90% of the maximum number of licensed users has been added. When 99% of the maximum number of licensed users has been added to the Vault, a notification will be sent each time another user is added.

Each time a notification is sent, a message will be written in the `ITALog`.

This notification is enabled by default immediately after installation. For more information about configuring notifications, refer to *Enabling the ENE*, page 803.

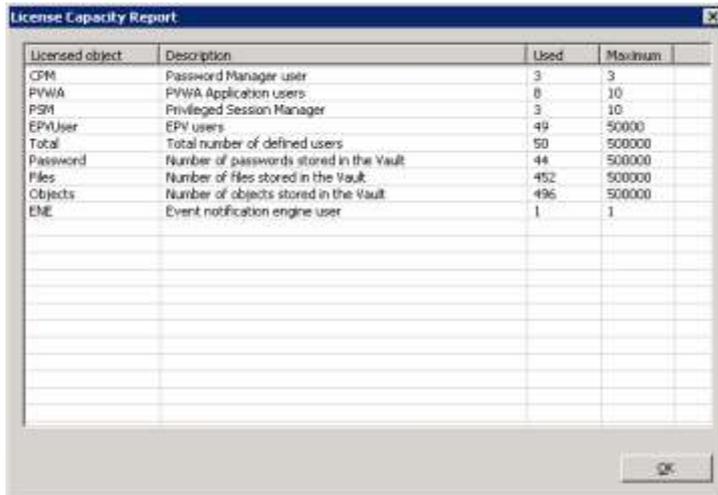
Reporting License Usage

The License Capacity report contains information about the licensed user types and objects in the Vault. It enables users to see the maximum number of licenses for each user type or object, and the number of used licenses for each one.

Only user types and objects that are limited by the license are displayed in this report. Predefined Vault users and groups are not included in the license usage.

To Display the License Capacity Report

- In the PrivateArk Client, from the **Tools** menu, select **Reports**, and then **License Capacity Report**; the report is displayed.



Licensed object	Description	Used	Maximum
CPM	Password Manager user	3	3
PVWA	PVWA Application users	8	10
PSM	Privileged Session Manager	3	10
EPVUser	EPV users	49	50000
Total	Total number of defined users	50	500000
Password	Number of passwords stored in the Vault	44	500000
Files	Number of files stored in the Vault	452	500000
Objects	Number of objects stored in the Vault	496	500000
ENE	Event notification engine user	1	1

Installing a New License

If you receive a new license from your CyberArk representative after you have installed the Vault, you can install it without having to reinstall the Vault. This license can be installed either from the Vault machine or from a remote machine.

To Install a New License from the PrivateArk Client

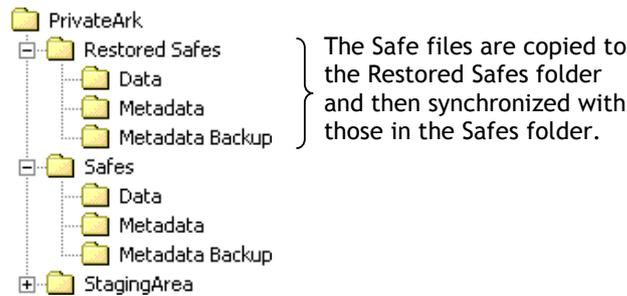
1. In the PrivateArk Client, log onto the Vault using the Administrator user or other user that has access to the System Safe.
2. From the System Safe, retrieve License.xml and save a backup copy.
3. Store the new License.xml in the Root folder of the System Safe; the system automatically installs the new License.xml.

Information Recovery

Restoring Safes or the Vault

In the event of system failure whereby the Vault or Safes are corrupted or in case of loss of data, the Vault enables you to restore the required metadata and data, and retrieve previous versions of the files in the Safe.

The following diagram shows the structure of the folder that contains the Restored Safes.



When the backup set is restored to the Restored Safes folder, the backed up metadata is synchronized with the Vault's metadata and the data files are copied to the Data folder. The Metadata Backup folder in the Restored Safes remains empty.

Note: If the backup was carried out with a third-party backup software, the metadata backup files must be restored from the Metadata Backup folder to the Restored Safes\Metadata folder, and the data files must be restored from the data folder to the Restored Safes\Data folder.

These files are then synchronized with the files in the Safes folder.

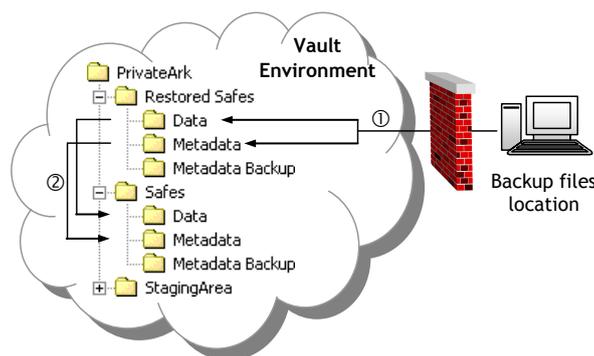
Restoring a Single Safe

In order to restore a single Safe, there must be an existing Safe of the same name in the Vault. The existing safe will not be replaced during a restore operation, and a new Safe will be created using the name specified by the user in PARestore. If a Safe has inadvertently been deleted and there is no Safe with the same name, create an empty Safe with the same name and add the Safe Owners who will require access, so that the Restore process will recognize the Safe as an existing one.

During a single Safe restore process, the following information will not be restored:

- Requests and Confirmations
- Access markers

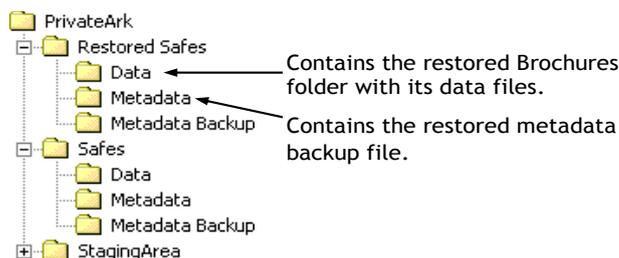
The following diagram shows the restoration process:



Step 1: As shown in the diagram above, restoration begins with transferring the Safe's metadata backup files and data files to the Restored Safes folders on the Server.

The metadata backup files and the data files that are updated to the same date and time must be restored to the Restored Safes folder in the same structure as they were backed up – metadata backup files in the Metadata folder, and data files in the Data folder.

For example, a Marketing department would like to restore a Safe called Brochures. The relevant metadata backup files (Backup-Dump.sql.gz and any Mysql-bin.*.gz files) should be restored to the Restored Safes\Metadata folder, and the folder containing the data files for the Brochures Safe should be restored to the Restored Safes\Data folder.



The way a Safe is restored depends on the way it was backed up, as described below.

- Restoring from the PrivateArk Replicator – If the backup procedure was carried out by the PrivateArk Replicator, use the **PARestore** utility to copy the Safe files to the Restored Safes folder. In order to be able to issue the PARestore command, the User must have the 'Restore All Safes' authorization in the Vault. A predefined group called 'Operators' is created during Vault installation and upgrading, and is added automatically to every Safe that is created. Each user that is subsequently assigned to this group must be given the restore authorizations manually.
- Restoring from an external backup utility – If the backup procedure was carried out by an external backup application, the same application should copy the metadata backup files and the data files to the Restored Safes folder.

Step 2: After the Safe files have been copied to the Restored Safes folder, they must be synchronized with the Safe files in the Safes folder. The PARestore utility does this automatically, but it can be split into two separate procedures.

If the Safe was backed up using the PrivateArk Replicator, the PARestore utility can unify these two steps and enable you to restore the Safe in a single command.

To Restore a Safe

Safes are restored using the PARestore utility, regardless of how they were backed up.

Notes: If a Safe with the name of the backed-up Safe does not exist in the Vault, before beginning the restore process, create a new Safe with the same name as the Safe that was removed. This Safe will remain empty, and the contents of the backed-up Safe will be restored to a target Safe with a different name that is specified during the restore process. To increase the level of security, the restore process synchronizes the Safe's owners of the existing Safe and the original Safe. As a result, when you restore a single Safe, its original Owners may not be restored with the Safe data and must be added manually.

- To restore a Safe that was backed up with the PrivateArk Replicator:

- At a command line prompt, use the following command:

```
PARestore <VaultFile> <User> /RestoreSafe <Safe> /TargetSafe
<NewSafe>
```

- To restore a Safe that was backed up with a third-party application

1. Follow the application's instructions to restore the entire Metadata folder and the Safe's folder into the Restored Safe's folder, as described above.
2. At a command line prompt, use the following command:

```
PARestore <VaultFile> <User> /RestoreSafe <Safe> /TargetSafe
<TargetSafe> /LoadOnly
```

For more information about PARestore, refer to *PARestore*, page 874.

Restore Permissions

To restore a Safe a User must have the 'Restore All Safes' authorization in the Vault. This means that a User is able to restore all the Safes, but it does not grant him automatic access to the Safes after they are restored. Only users who have Safe ownership will be able to access restored Safes.

The 'Restore All Safes' authorization enables a User to issue the PARestore utility. When using this utility, the User will be required to supply his User name and password. The Vault will then verify the User identity and check his authorizations to administer this specific Safe. If the User does not have the required rights, the operation will not be carried out.

The predefined 'Operator' user has the 'Manage Safe' authorization for each Safe, and is also assigned to the 'Operators' predefined group automatically. However, when additional users are added to this group, they must each be given the 'Restore All Safes' authorization in the Vault separately.

The user who will issue 'CAVaultManager RestoreDB' to restore a full Vault is not required to authenticate to the Vault. However, the full Vault can only be restored on the Vault machine.

Restoring a Vault

Full Vault restoration is carried out in the event of general failure.

Restoration involves restoring the Safe folders to a new computer that is suitable as a Vault Server. For more information about Vault requirements, refer to the Privileged Account Security Installation Guide. After installing the Server and configuring it to work with the Safes, the new Vault will have the same Safe names, Users, and authorizations as the old Vault.

The new Server must use the same Master and Operator keys as the old Server.

To Restore the Vault

1. Install a new version of the PrivateArk Server, and make sure that it is fully operational.
 2. Enable the Operator user or another user that has the following authorization in the Vault:
 - Restore All Safes
 3. In `DBParm.ini`, set the following parameter:
`BackupFilesDeletion=No`
 4. Copy all the relevant backup files (metadata and data) to the restored Safes folder on the Vault server machine.
 - If you used the PrivateArk Replicator, use the following command:
`PARestore vault.ini operator /FullVaultRestore`
For more information about `PARestore`, refer to *PARestore*, page 874.
- Or,
- If you used a third-party application to backup the Safe, follow the application's instructions
 5. Open the PrivateArk Server Management Console and stop the CyberArk Vault Server, then close the PrivateArk Server Management Console.
 6. From the Server installation folder, run the `CAVaultManager` utility, as follows:

```
CAVaultManager RecoverBackupFiles
```

Notes:

- This command requires you to use the Master CD.
 - In Windows 2008 R2, run `CAVaultManager RecoverBackupFiles` in an elevated session.
7. From the Server installation folder, run the `CAVaultManager` utility, as follows:

```
CAVaultManager RestoreDB
```

Note: Run `CAVaultManager RestoreDB` in an elevated session.

This will complete the restore process and synchronize the restored metadata and data. For more information, refer to *CAVaultManager*, page 876.

8. In `DBParm.ini`, reset the **BackupFilesDeletion** parameter. This parameter's default value is:

```
BackupFilesDeletion=Yes,24,1,5,7d
```

However, if this parameter specified other values before you restored the Vault, specify the same values again to meet your enterprise needs.

9. Modify the Vault configuration files to duplicate the previous Vault's settings.
10. Start the CyberArk Vault Server.

Restore Utilities

PARestore

The PARestore utility enables you to restore Safes that have previously been either replicated or backed up to the Vault.

The Safe data files are restored to the PrivateArk\Restored Safes folder in the same structure as that in which they were backed up. After the metadata backup files are restored to the PrivateArk\Restored Safes\Metadata folder, a synchronization procedure will take place, after which users will be able to work with the files immediately.

Only Users with the 'Restore All Safes' authorization in the Vault can restore a Safe. For more information, refer to *Restore Permissions*, page 872.

PARestore provides the following options:

```
PARestore <VaultFile> <User> /<Password>
          /LogonFromFile logonfile
          /BackupPoolName BackupPoolName
          /RestoreSafe safename /TargetSafe newname
          /DataOnly | /MetadataOnly
          /LoadOnly
          /FullVaultRestore
          /FromRestored
          /?
```

This usage is explained in the following table and examples:

Option	Description
<VaultFile>	The file containing all the information about the Vault and the Safes within it. By default, this file is called Vault.ini.
<User>	The name of the User issuing the command. This User must have the 'Restore All Safes' authorization in the Vault.
/password	The password of the User specified above. If the User issues this command without specifying the password and without specifying the /LogonFromFile parameter, the User is prompted for it before the command is carried out.
/LogonFromFile	The pathname of a user credentials file containing an encrypted password that the utility will use to log on instead of a password.
/BackupPoolName	Specifies a Backup Pool Name. This is used when there are several backup sets for a Vault, or a number of clients used to backup the server. The Pool Name must match the pool name specified in the backup process, so that you can distinguish between different backup sets.
/RestoreSafe	The name of the Safe to restore. This Safe must exist in the backup files from which the restore is made, as well as in the Vault. If the Safe was deleted after backup, create a Safe with the same name, and add the same Safe Owners that were assigned to the backed up Safe.

Option	Description
/TargetSafe	The name of the restored Safe to create. The restore process does not overwrite an existing Safe – it creates a new one. Therefore, this name must not correspond with an existing Safe.
/MetadataOnly	Only the Safe's metadata will be uploaded and restored, but not the external files.
/DataOnly	Only the Safe's data files will be uploaded and restored, but not the metadata. For this option to work, the metadata must already have been uploaded to the Vault with a separate PARestore command. A Safe cannot be restored without its corresponding metadata.
/LoadOnly	Prevents PARestore from copying files to the Restored Safes folder on the Vault file system. If a third-party backup application is used, after files have been restored to the Restored Safes folders in the Vault, this parameter will enable PARestore to use the files already in the Vault. Note: This parameter and 'FullVaultRestore' are mutually exclusive and cannot be used together.
/FullVaultRestore	Restores a complete Vault by uploading all the metadata and data files to the Vault's Restored Safes folder. After the upload process has finished, run 'CAVaultManager RestoreDB' to complete the restore process. Note: This parameter and 'LoadOnly' are mutually exclusive and cannot be used together.
/FromRestored	The Safes are restored to the Vault from the Restored Safes folder in the Replicator computer.
/?	Displays the list of options available with this utility.

Note: After PARestore restores a single Safe, the entire metadata is loaded. This means that if another Safe is restored from the same backup, there is no need to upload or load the metadata files again, and running PARestore /DataOnly is enough.

For example:

```
parestore C:\PrivateArk\Server\Vault.ini Backup/asdf1 /restoresafe
banners /targetsafe banners_r
```

The above example restores the "Banners" Safe from the backup computer to the Vault. All the relevant information about the Vault will be taken from the Vault.ini file in C:\PrivateArk\Server. This command is issued in the Backup User's name, using his password which is asdf1. All the files in the "Banners" Safe will be restored into a new Safe called "Banners_r". This command will restore the metadata backup files and the data files to the Vault, load the metadata of the relevant Safe, and finally synchronize the Safe's metadata with its external files.

CAVaultManager

The `CAVaultManager` utility enables you to restore a complete Vault after a disaster has occurred.

After uploading the backup files to the Restored Safes folder in the Vault machine either with a third-party backup application or by issuing 'PAREstore /FullVaultRestore', use the `CAVaultManager` utility to synchronize the restored data and make the Vault operational again.

CAVaultManager RestoreDB

`CAVaultManager` uses the `RestoreDB` parameter to provide the following options:

```
CAVaultManager RestoreDB
                [/BackupPoolName BackupPoolName]
                [/NoSynchronize]
                [/Force]
```

Note: In Windows 2008 R2, run `CAVaultManager RestoreDB` in an elevated session.

This usage is explained in the following table and example:

Option	Description
/BackupPoolName	Specifies a Backup Pool Name. This is used when there are several backup sets for a Vault, or when several clients are used to backup the server. This Pool Name must match the pool name specified in the backup process, providing a way to distinguish between different backup sets.
/NoSynchronize	Enables the user carrying out the restore process to separate the synchronization stage from the loading stage. If this parameter is specified, <code>CAVaultManager SynchronizeDB</code> must be executed before the Vault can be used properly.
/Force	Prevents the application from displaying a confirmation message to the user before completing the restore/synchronize process.
/?	Displays the list of options available with this utility.

Note: When `PAREstore` is run to restore a single Safe, the entire metadata is loaded. As a result, there is no need to upload or load the metadata files again to restore another Safe.

Run `PAREstore /DataOnly` to upload and restore the Safe's data files.

For example:

```
CAVaultManager RestoreDB
```

The above example restores the entire Vault from the Restored Safes folder on the Vault machine. It will load the metadata, copy the data files to the Safes\Data folder, and synchronize the data with the metadata. Once the operation is complete, the Vault is operational and usable, and reflects the exact state it was in when the backup was taken.

If you run `CAVaultManager RestoreDB` with `/NoSynchronize`, use the following `CAVaultManager` utility parameters to synchronize the Vault so that you can use the Vault properly.

CAVaultManager SynchronizeDB

CAVaultManager uses the SynchronizeDB parameter to synchronize the Vault database after the backup files have been transferred to the Vault from backup data. It provides the following options:

```
CAVaultManager SynchronizeDB
                [/SafePattern <Pattern>]
                /FilesSyncOnly
                /QuotaSyncOnly
                /Update
                /Force
                /?
```

Note: In Windows 2008 R2, run CAVaultManager SynchronizeDB in an elevated session.

This usage is explained in the following table and example:

Option	Description
/SafePattern	A Safe pattern indicating the Safes that will be synchronized with the restored data and metadata.
/FilesSyncOnly	Enables synchronization between the files in the Safes folder and the restored metadata.
/QuotaSyncOnly	Enables synchronization between the quotas in the Safes folder and the restored metadata.
/Update	Carries out the synchronization process. If this parameter is not specified, the synchronization will only be simulated.
/Force	Prevents the application from displaying a confirmation message to the user before completing the restore/synchronize process.
/?	Displays the list of options available with this utility.

For example:

```
CAVaultManager SynchronizeDB /FilesSyncOnly /Update
```

The above example will synchronize the backup files that were restored in the Vault (using the RestoreDB command) with the restored Metadata. This command will be carried out, rather than simulated, and will prompt the user for confirmation during the process.

CAVaultManager RecoverBackupFiles

CAVaultManager uses the RecoverBackupFiles parameter to access the backup files in the Restored Safes folder with the Vault's Recovery Private Key and re-encrypt them with a new backup key when the original backup key cannot be used. It provides the following options:

```
CAVaultManager RecoverBackupFiles
                [/BackupPoolName]
                /?
```

Notes:

- This command requires you to use the Master CD.
- In Windows 2008 R2, run `CAVaultManager RecoverBackupFiles` in an elevated session.

This usage is explained in the following table and example:

Option	Description
/BackupPoolName	Specifies a Backup Pool Name. This is used when there are several backup sets for a Vault, or when several clients are used to backup the server. This Pool Name must match the pool name specified in the backup process, providing a way to distinguish between different backup sets.
/?	Displays the list of options available with this utility.

For example:

```
CAVaultManager RecoverBackupFiles /BackupPoolName BkpSvr1
```

The above example will recover the backup files from a Backup Pool called **BkpSvr1**, and re-encrypt them with a new accessible backup key.

CyberArk HA Vault Cluster

The following instructions describe the steps to take if one of the Vault cluster nodes becomes inactive and you want to add a new node to the cluster.

Adding a New Node

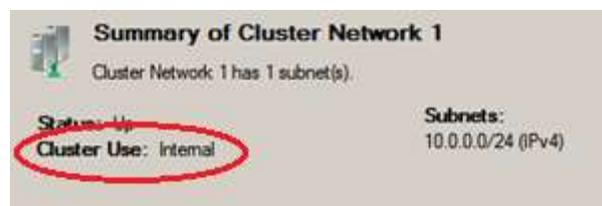
In the following steps, the nodes are described as follows:

- Node A – Working node
- Node B – Re-installed node

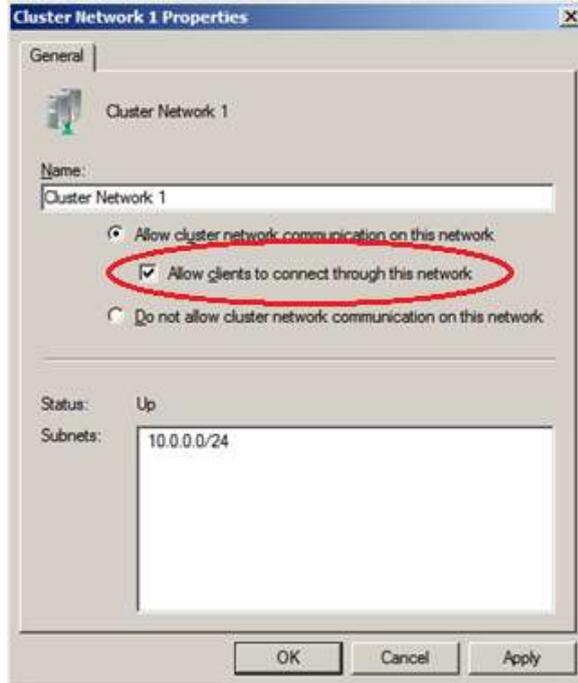
To Add a New Node on Windows 2008

Note: Use the Administrator user for this task.

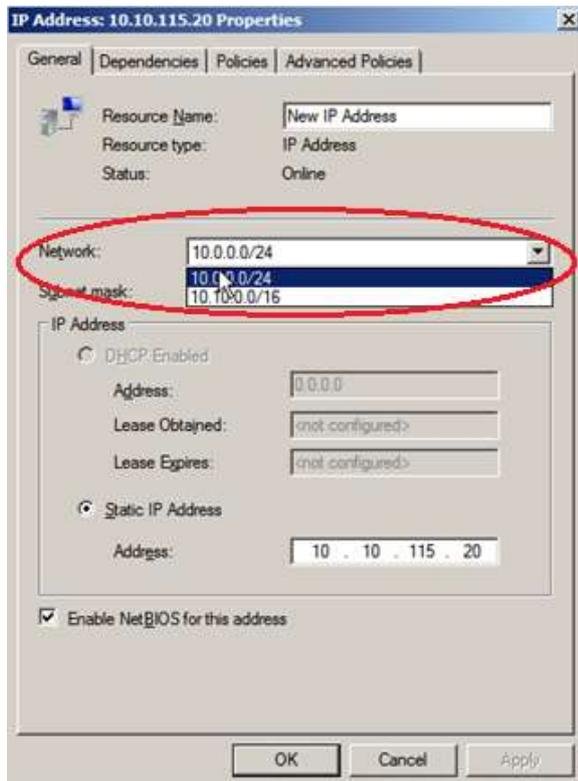
1. In the cluster administrator, evict the old node B from the Vault cluster.
2. Delete the old node B from the Domain Controller and from the DNS.
3. On node A:
 - i. In the registry, set the **HKLM\Software\Microsoft\OLE\EnableDCOM** key to **Y**.
 - ii. Restart node A to enable the above configurations to take effect.
4. On node B:
 - i. On the private network card, set the DNS server address to the private address of node A.
 - ii. Add node B to the Cluster domain, then run **dcpromo** and promote it to Domain Controller.
 - iii. Install the DNS server, then on the private network card set the DNS server address to the private address of node B.
 - iv. Check that there are no errors in the event log.
5. Node B is added to the cluster using the cluster IP, which is automatically blocked by the firewall. The following steps enable you to add node B to the cluster:
 - i. Change the Cluster Network from **Internal** to **External**.
 - a. Display the summary of the private Cluster Network. The Cluster Use displays **Internal**.



- b. Right-click the name of the network then, from the pop-up menu, select **Properties**; the Cluster Network Properties window appears.
- c. Check **Allow clients to connect through this network**.



- d. Click **OK**; the summary of the private Cluster Network now indicates that the cluster is for External use.
- ii. Change the IP address of the cluster to a private network IP:
 - a. In the main Failover Cluster Manager menu, select the cluster; the IP Address window appears.
 - b. In the General tab, in **Network**, select the private network IP.



- c. Select **Static IP Address** then, in the Address, specify a private network IP address.
 - d. Click **OK.**, the cluster IP is now configured as a private network IP.
 - e. Make sure that the IP resource is online with the private IP address.
 - iii. In the firewall rules for node A, make sure that the following rules contain accurate IP and MAC addresses of node B:
 - Cluster-Nodes-Incoming
 - Cluster-Nodes-OutgoingIf you have moved node B to a computer with a different NIC card to the previous node B, one or both addresses probably need to be updated.
6. On node A, add node B to the cluster.
7. Set the **PrivateArk Server** and **Database** resources to **offline** mode.
8. On node A:
 - i. In the registry, set the **HKLM\Software\Microsoft\OLE\EnableDCOM** key to **N**.
 - ii. Return the cluster IP address resource to its former IP address on the public network.
 - iii. In the Private Network Properties, clear **Allow clients to connect through this network**; the private network is reset as an Internal network.
 - iv. Shutdown node A; the cluster resources will be automatically moved to node B.
9. Harden node B according to the Vault Cluster Hardening procedure in the Privileged Account Security Installation Guide.
10. Restart node B.
11. On node B, install the CyberArk Vault according to the Installation procedure in the Privileged Account Security Installation Guide.
12. Set the **PrivateArk Server** and **Database** resources to **online** mode.
13. Start node A.

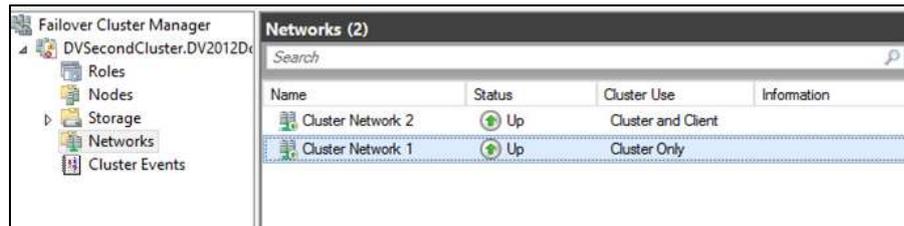
To Add a New Node on Windows 2012 R2

Note: Use the VaultAdmin user for this task.

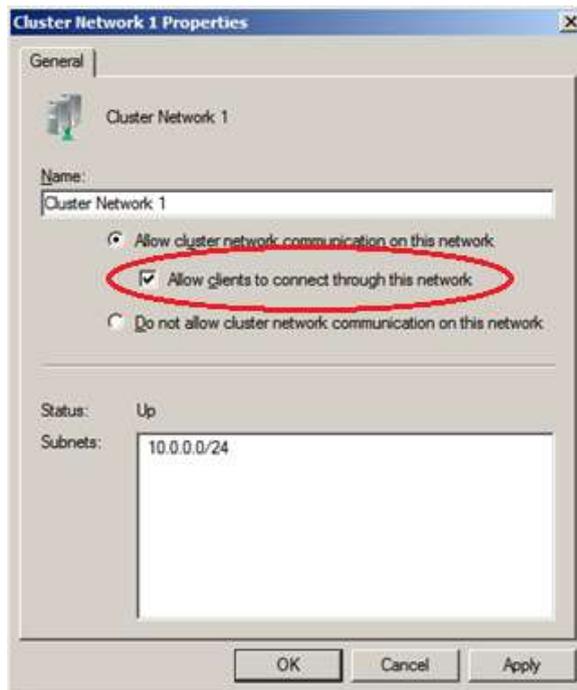
1. In the cluster administrator, evict the old node B from the Vault cluster.
2. Delete the old node B from the Domain Controller and from the DNS.
3. On node A:
 - i. In the registry, set the **HKLM\Software\Microsoft\OLE\EnableDCOM** key to **Y**.
 - ii. Restart node A to enable the above configurations to take effect.
4. On node B:
 - i. On the public network card, set the DNS server address to the domain's DNS server IP.
 - ii. Add node B as a domain member (of the same domain where Node A is a member).

- iii. Using the local Administrator account, add the VaultAdmin domain user as a member of the local Administrators Group.

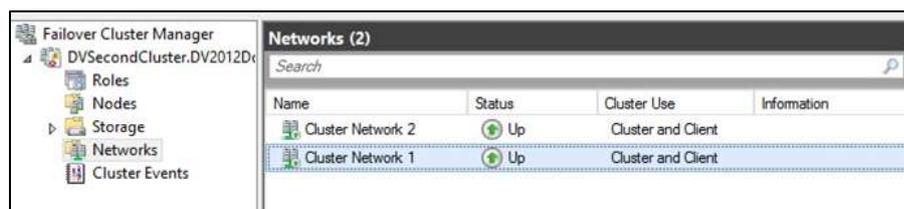
Note: Use the **VaultAdmin** domain user to perform the rest of the steps in this procedure.
- iv. Check that there are no errors in the event log.
5. Node B is added to the cluster using the cluster IP, which is automatically blocked by the firewall. The following steps enable you to add node B to the cluster:
 - i. Change the Cluster Network from **Internal** to **External**.
 - a. In the cluster manager, display the **Networks** menu. The 'Cluster Use' for the private Cluster Network displays **Cluster Only**.



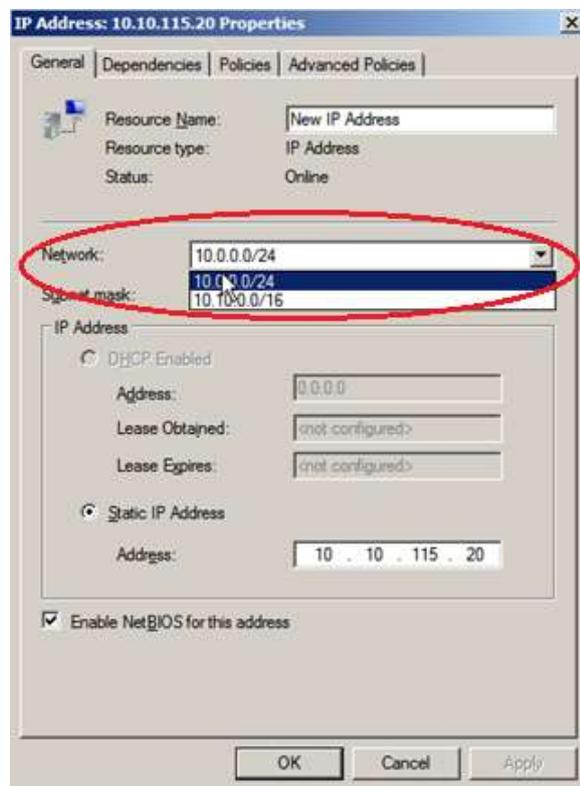
- b. Right-click the name of the network then, from the pop-up menu, select **Properties**; the Cluster Network Properties window appears.
- c. Check **Allow clients to connect through this network**.



- d. Click **OK**; the 'Cluster Use' of the private Cluster Network now indicates that the cluster is for **Cluster and Client** use.



- ii. Change the IP address of the cluster to a private network IP:
 - a. In the main Failover Cluster Manager menu, select the cluster; the IP Address window appears.
 - b. Right-click the IP Address cluster core resource and select **Properties**; the IP Address Properties window appears.
 - c. In the General tab, in **Network**, select the private network IP.



- d. Select **Static IP Address** then, in the Address, specify a private network IP address.
 - e. Click **OK.**, the cluster IP is now configured as a private network IP.
 - f. Make sure that the IP resource is online with the private IP address.
- iii. In the firewall rules for node A, make sure that the following rules contain accurate IP and MAC addresses of node B:
 - Cluster-Nodes-Incoming
 - Cluster-Nodes-Outgoing

If you have moved node B to a computer with a different NIC card to the previous node B, one or both addresses probably need to be updated.
 6. On both nodes A and B, in the Public Network Properties, make sure the following items are enabled:
 - Client for Microsoft Networks
 - File and Printer Sharing for Microsoft Networks
 7. On node A, in the Windows Firewall, **add** the following rules:
 - Allow TCP connections to ports 49152-65535 (Outbound only)
 - Allow TCP connections to and from port 3343 (Inbound and Outbound)
 8. On node A, add node B to the cluster.
 9. Restart node B.

10. On node A, in the Windows Firewall:
 - i. **Remove** the rules that you added in the previous steps:
 - Allow TCP connections to ports 49152-65535 (Outbound only)
 - Allow TCP connections to and from port 3343 (Inbound and Outbound).
 - ii. Make sure the following rules are set to the address of node B Failover Cluster Virtual Adapter:
 - CyberArk Cluster – Failover Clustering – TCP – In (Inbound)
 - CyberArk Cluster – Failover Clustering – TCP – Out (Outbound)

Note: If Node B's "Failover Cluster Virtual Adapter" IPv6 address is dynamic, this rule may have to be configured with "Any" (less recommended).
11. Set the **PrivateArk Server**, **CyberArk Logic Container** and **Database** resources to **offline** mode.
12. On node A:
 - i. Return the cluster IP address resource to its former IP address on the public network.
 - ii. In the Private Network Properties, clear **Allow clients to connect through this network**; the private network is reset as an Internal network.
 - iii. In the Public Network Properties, clear the following properties:
 - Client for Microsoft Networks
 - File and Printer Sharing for Microsoft Networks
 - iv. Restart the cluster service on Node A; the cluster resources will be automatically moved to node B.
13. Harden node B according to the Vault Cluster Hardening procedure in the Privileged Account Security Installation Guide.
14. Restart node B.
15. On node B, install the CyberArk Vault according to the Installation procedure in the Privileged Account Security Installation Guide.
16. Set the **PrivateArk Server** and **Database** resources to **online** mode.
17. On node A, in the registry, set the **HKLM\Software\Microsoft\OLE\EnableDCOM** key to **N**.

CyberArk Disaster Recovery Vault

An important aspect of the CyberArk Vault as a critical data repository for the organization is its ability to continue functioning even during severe failures (the most extreme of which is complete physical destruction of the Vault site). Organizations need the Vault to be able to resume operations quickly (in a matter of minutes) without losing any data as a result of the failure.



Figure 1 – Disaster Recovery Vault – Network Architecture

The Disaster Recovery (DR) Vault is a replication/failover solution designed to create a stand-by copy of a Production Vault on a remote and dedicated machine (the Disaster Recovery Vault Machine) that can be made operational quickly if the original Vault fails.

A completely transparent failover can be configured for critical components, such as PVWA, to enable them to work with a DR Vault as soon as the Production Vault cannot be reached, without any human intervention or reliance on load-balancing dedicated hardware.

The Disaster Recovery Vault meets the following requirements:

- Replicates data from the Production site to the Disaster Recovery site.
- Automatically identifies the Production Vault failure and begins the failover process in the Disaster Recovery Vault.
- Highly Secure Protection of the data on the Disaster Recovery site.

Disaster Recovery User

The Disaster Recovery User (DR User) is a predefined User that is added automatically as an Owner to every Safe, and only has the access rights required to replicate the Safes. The predefined DR User makes it easier to replicate your data to the Disaster Recovery Vault.

Note: After installation, the DR User account is disabled. Before using the DR User, enable it on the Primary Vault and update its password.

Maintaining the DR Vault

For performance enhancement, keep the metadata small by clearing the history regularly. This can be made automatic with the following parameters in DBParm.ini in the Production Vault:

- **AutoClearSafeHistory** – Automatically clears Safe history at predefined intervals.
- **AutoClearUserHistory** – Automatically clears user history at predefined intervals.

In addition, old metadata backup files should be cleared regularly. This can also be made automatic with the following parameter in DBParm.ini in the Production Vault:

- **BackupFilesDeletion** – Schedules deletion of old backup files (exports and binary logs) from metadata, metadata backup, and restored safes directories.

To enforce a full DR process, in PADR.ini delete the following parameters:

- NextBinaryLogNumberToStartAt
- LastDataReplicationTimestamp

Replicating Component User Passwords

Vault component users who access the Vault to perform tasks for different components use credential files to authenticate to the Vault. At regular intervals, the Vault component automatically initiates a password change in the Vault and in the corresponding credential file for that Vault component. In order for the same component user in the DR Vault to access the Vault and continue working seamlessly in a Disaster Recovery situation, the user's new credentials must be replicated to the DR Vault whenever they are changed.

The credential files password synchronization is currently supported by the PWWA, CPM, OPM, and PSM.

This is configured by the following parameter in the CreateCredFile utility:

- **DisableSyncPasswordToDR** – Whether or not passwords in user credential files will be replicated to all DR sites before they are replaced. The default value of this parameter is 'No', which makes sure that user credential files on all DR sites (if they exist) are synchronized with the Production Vault and that users will be able to continue working with the Vault seamlessly after a failover. If this parameter is changed to 'Yes', passwords will be replaced in credential files regardless of whether or not they have been replicated to all DR sites.

Monitoring Vault Availability

Before the DR Vault takes over from the production Vault, it can check whether or not the production Vault is active by trying to access the Vault using ICMP or by initiating Vault activity. The `AccessVaultForInactivity` parameter in `PADR.ini` determines which method the DR user will use to try to access the Vault:

To use ICMP, specify the following value:

```
AccessVaultForInactivity=No
```

To initiate Vault activity, specify the following value:

```
AccessVaultForInactivity=Yes
```

This parameter is added to the `PADR.ini` automatically during installation, but not during upgrade. After upgrading the DR Vault, add this parameter manually to activate this feature.

Logging

A log file, called `PADR.log`, records all the activities carried out by the DR Vault. This file is stored in the `PrivateArk\PADR` folder on the Vault machine. When the log file reaches 100MB, it is automatically moved into the `PrivateArk\PADR\Old` subfolder and a new log file will be created.

The level of information included in the `PADR.log` is determined by the **EnableTrace** parameter in the `PADR.ini` file, as follows:

- To include all DR activities in the log file, specify the following:

```
EnableTrace=no
```

- To include all debug, activity, and error messages in the log file, specify the following:

```
EnableTrace=yes
```

- To include communication related messages, in addition to debug, activity, and error messages in the log file, specify the following:

```
EnableTrace=full
```

In addition, critical log messages are copied to the Microsoft Event log.

Advanced Digital Vault Environment

This chapter introduces you to advanced aspects of the Vault environment and describes how to configure it.

This chapter comprises the following sections:

The CyberArk Vault:

- Server Components
- Server Files
- Server Utilities
- Server Keys

The CyberArk Administrative Client:

- PrivateArk Client Components
- PrivateArk Client Files
- PrivateArk Client Configuration
- The PrivateArk Client in a Remote Desktop Services (Terminal Services) Environment
- The PrivateArk Client Log File
- PrivateArk Information

CyberArk Vault Structure

This section describes the software and file components of the Vault Server and is comprised of the following subsections:

- Server Components describes the Vault software.
- Server Files describes the files that comprise the Vault.
- Server Utilities describes the utilities which assist in managing the Vault and the Server database.

Server Components

The CyberArk Vault software is comprised of the following components:

The PrivateArk Server process (Dbmain)

The PrivateArk Server process is a Windows service. This service can start automatically or manually depending on the Server's key configuration.

You have the option of running the Server process in "console" mode and not as a service. This option is used mainly for troubleshooting.

For more information on how to operate the Server process, refer to *Operating the CyberArk Vault*, page 835.

The PrivateArk Server Interface (SrvGui.EXE)

The Server GUI can be used to operate the Server process, start and stop the Vault, and view the Vault log.

To access the Server GUI, click the PrivateArk Server icon on your desktop.

The Vault Firewall (PAFW.SYS)

Note: This section is not relevant to Vault servers that are installed on Windows 2008 R2, as the Vault utilizes the Windows native firewall.

This controls the incoming and outgoing communications to and from the Server.

The firewall is defined as a Windows device driver and can be viewed using the Windows Devices interface. This firewall is also defined as a network service. There is absolutely no need to configure the firewall. Starting, stopping or changing definitions of the firewall device or the network service are not recommended and can seriously reduce the level of protection given to information stored in the Vault.

The PrivateArk Client

Installation includes a complete installation of the PrivateArk Client, the windows interface that enables you to access the Vault. The PrivateArk Client is used to log onto the Vault from the Vault station itself for initial configuration.

Server Files

The Server is comprised of the following file types:

The PrivateArk Binaries

The PrivateArk binaries are comprised of the Server and Client executables. Both of these files are located in the installation directory. The default directory paths to the two executable files are as follows:

- \Program files\PrivateArk\Server
- \Program files\PrivateArk\Client

Server Configuration Files

The Server uses the registry to point to the location of the Server's configuration files and logs.

The Server registry definitions are located in the following directory:

HKLM\Software\CyberArk\PrivateArk\ Server\

The configuration files control the initial settings and method of operation of the Server. The Vault has several configuration files, as follows:

- DBParm.ini, which contains the general parameters of the database.
- TSParm.ini, which contains the paths to directories where Safes can be located. These definitions can be managed from the Server interface.
- PassParm.ini, which contains the password rules of the server.
- License.xml, which contains the Customer license.
- ExtAuth.ini, which contains the parameters for user definitions from external directories.

For more information, refer to *CyberArk Vault Server Parameter Files* in the Privileged Account Security Reference Guide.

Server Log File

ITALog.log is the Server log file. It can be viewed from the Server interface.

The 'LogRetention' parameter in the DBParm.ini file determines the amount of time that log records will be kept. The expired log records are cleared whenever the Server is started.

Debug File

Debug.log is the Server debug log.

The 'Debug' parameter in the DBParm.ini file, determines whether debug records are written to the debug log.

The Entropy File

The Entropy file, located in the Safe's directory, is created during installation. It is used to increase efficiency and security of the Vault's key files.

Note: Run `CAVaultManager /SecureEntropyFile` using the `Rndbase.dat` file on the Operator's CD to create the Entropy file.

Staging Area Folder

To improve performance, each time a User retrieves a file from the Vault, or returns it, the file is stored in the Staging Area Folder during the transfer. After the file has been transferred successfully, all traces of the file are cleaned from this folder. In addition, all files stored in this folder are encrypted.

Server Utilities

The Server utilities assist in managing the Server and the Server database. They are operated from the command line prompt.

Before you run any of the following Server utilities, stop the Server. Restart the Server after running the utility.

In each server utility, the `/ACKS` option will display an acknowledgement notice and then the utility help.

Notes:

- The parameters of utilities that are run on Windows are preceded by a `'/'` (slash).
- Parameters that contain spaces must be enclosed in quotation marks (`" "`).

CAVaultManager

The `CAVaultManager` utility enables you to manage the Vault database.

`CAVaultManager` has the following usage:

```
CAVaultManager <command> [command parameters]
```

Note: In Windows 2008 R2, run `CAVaultManager` in an elevated session.

The usage is explained in the following table:

Parameter	Description	Mandatory
<code>CreateDB</code>	Creates the Vault database.	
<code>SecureDB</code>	Secures the Vault database.	
<code>/MasterPassword</code>	The password for the Master user. This is a required parameter.	Yes
<code>/RndBaseFileName</code>	The path where the initial entropy file is saved. This is a required parameter.	Yes
<code>/DBEmergency PasswordFileName</code>	The name of the file where the encrypted emergency password for database access is stored. This is a required parameter.	Yes

Parameter	Description	Mandatory
SecureSecretFiles	Secures the Vault's secret files.	
/SecretType	The type of secret to secure. Options are LDAP, Radius, or HSM.	Yes
/Secret	The secret.	Yes
/SecuredFileName	The name of the file where the secured secret is stored.	No
/FileSectionName	Name of LDAP host section to secure within the file. Default is LDAP directory section.	No
SecureEntropyFile	Secures the Vault entropy file.	
/RndBaseFileName	The path where the random number generator state is saved. This is a required parameter.	Yes
OptimizeDB	Optimizes Vault performance. Note: In Windows 2008 R2, this command must be run in an elevated command session.	
UpgradeDB	Upgrades the Vault database.	
DeleteDB	Deletes the Vault database.	
RecoverDBPassword	Recovers the Vault database connection password.	
/DBEmergencyPasswordFileName	The name of the file where the encrypted emergency password for database access is stored. This is a required parameter.	Yes
/DBNewPassword	The new password for database access.	No
LDAPVerify	Verifies LDAP component configuration.	
/ConfOnly	Verifies only LDAP configuration files.	No
/Verbose	Displays details of the LDAP verification checks.	No
RestoreDB	Restores the Vault database.	
/BackupPoolName	The name of the backup set that the command refers to.	No
/NoSynchronize	Does not synchronize the restored external files with the restored metadata, as it may result in safes containing files that aren't actually there.	No
/Force	Synchronizes the existing and the restored databases without prompting the user for confirmation.	No
SynchronizeDB	Synchronizes the files in the Safes folder with the restored metadata.	
/SafePattern	A Safe pattern indicating the Safes that will be synchronized with the restored data.	No

Parameter	Description	Mandatory
/FilesSyncOnly	Enables a synchronization between the files in the Restored Safes folder and the Safes folder.	No
/QuotaSyncOnly	Enables synchronization between the quotas in the Restored Safes folder and the Safes folder.	No
/Update	Updates the data in the Safes folder during the synchronization process.	No
/Force	Prevents the application from displaying a confirmation message to the user before completing the restore/synchronize process.	No
RecoverBackupFiles	Recovers the backup files and re-encrypts them with a new backup key. Note: In Windows 2008 R2, this command must be run in an elevated command session.	
/BackupPoolName	The name of the backup set that the command refers to.	No
DiagnoseDBReport	Compiles a diagnostics report for the CyberArk Vault database	
/OutputFileName	The name of the report output file.	No
GenerateKeyOnHSM	Generates new encryption keys on the HSM. This parameter is mandatory if the HSM key will be generated on the HSM device.	No
/ServerKey	Determines that server keys will be generated on the HSM device. This parameter is mandatory if the HSM key will be generated on the HSM device.	No
LoadServerKeyToHSM	Uploads the Server key to the HSM and updates the relevant parameters in DBParm.ini.	
/Pincode	The PIN code required to upload the Server key to the HSM.	No
ReplaceLDAPDirectory	Changes references in directory maps, users and groups from the current external directory to a different one.	
/CurrentLDAPDirectory <old_directory>	The name of the external directory that these objects currently reference.	Yes
/NewLDAPDirectory <new_directory>	The name of the new external directory that these objects will reference.	Yes
[/Update]	Indicates whether the directory maps, users and groups will be updated or this operation will be performed in simulation mode.	No
AppendFriendlyDomain NameToGroup	Adds active directory domain names to names of groups that are provisioned in the Vault.	

Parameter	Description	Mandatory
/Update	Indicates whether the active directory domain name will be added to names of groups that are provisioned in the Vault or this operation will be performed in simulation mode.	No
TerminateDBTransaction	Enables you to manually terminate transactions that have been running longer than a specified period of time.	
/DBTransactionID	The unique transaction ID of the long transaction. This ID appears in the alert message that is written in the italog file when the transaction is identified by the MonitorLongTransactions parameter in DBParm.ini.	No
RecoverReplicationPassword	Recovers the replication user's password.	No
StartDBReplication	Begins the database replication. This command is issued from the DR site.	No
StopDBReplication	Stops the database replication. This command is issued from the DR site.	

Creating a Database

```
CAVaultManager CreateDB
```

This command will create a new Vault database.

Securing the Vault Database

```
CAVaultManager SecureDB /MasterPassword <Password> RndBaseFileName
<Filename> /DBEmergencyPasswordFileName <Filename>
```

This command secures the Vault database using the master password and the initial entropy file, then creates and stores an encrypted password in an emergency password file which enables access to the Vault database.

For example:

```
CAVaultManager SecureDB /MasterPassword mstrpwd123 /RndBaseFileName
C:\rndbasefile.dat /DBEmergencyPasswordFileName
C:\VaultEmergency.pass
```

The above example will secure the Vault database, using the Master password **mstrpwd123** and the initial entropy file stored in **c:\rndbasefile.dat**, then will create and encrypt an emergency password and store it in **C:\VaultEmergency.pass**.

Securing Secret Files

This command secures the files that contain either the Radius or LDAP secret.

```
CAVaultManager SecureSecretFiles
[/SecretType <Type>] [/Secret <Secret>] [/SecuredFileName <Filename>]
[/FileSectionName <SectionName>]
```

Example 1:

```
CAVaultManager SecureSecretFiles /SecretType Radius /Secret
VaultSecret /SecuredFileName c:\RadiusSecret.txt
```

The above example will create a file called **c:\RadiusSecret.txt** that contains the encrypted **Radius** secret, **VaultSecret**.

Example 2:

```
CAVaultManager SecureSecretFiles /SecretType LDAP /Secret LDAPSecret
/SecuredFileName "c:\Program Files\ PrivateArk\Server\LDAP
\Directories\ ActiveDirectory.ini" /FileSectionName LDAPHost2
```

The above example will open an existing file called **c:\Program Files\ PrivateArk\Server\LDAP\Directories\ ActiveDirectory.ini** that contains the encrypted **LDAP** secret, **LDAPSecret**. This command will secure the section called **LDAPHost2** in the specified file by inserting the encrypted secret into the secured section.

Securing the Vault Entropy File

```
CAVaultManager SecureEntropyFile [/RndBaseFileName <Filename>]
```

This command will create the Vault's entropy file with the initial entropy file and secure it using the server key.

For example:

```
CAVaultManager SecureEntropyFile /RndBaseFileName c:\rndbasefile.dat
```

The above example will create the Vault's entropy file with the initial entropy file stored in **c:\rndbasefile.dat** and then secure it with the server key.

Optimizing Vault Performance

This command configures the Vault for optimal performance, by optimizing the database structure and reclaiming unused database space.

This command creates a folder in **D:\PrivateArk\Safes\Metadata OptimizeDB Backups** especially for backup files that are created when this command is run. The name of the backup file is comprised of the date and time of the backup.

Note: As all the backup files are saved in this folder, which is not cleared automatically, make sure that you clear this folder regularly.

Although this command creates its own backup, before running this command, perform a full backup of the Vault database.

To Optimize Vault Performance

1. Shut down the Vault server.
2. In the **PrivateArk\Server\Database** folder, backup the **my.ini** configuration file.

- Open the my.ini configuration file and make the following changes:

```
innodb_flush_log_at_trx_commit=0 # Default is 1
Comment log-bin=mysql-bin
```

- In HA implementations only, in the Cluster Administration Utility, set the **PrivateArk Database resource to offline**, and manually start this service.
- Run the following command:

```
CAVaultManager.exe OptimizeDB
```

This command does not require any parameters.

- Change the settings in the my.ini configuration file back to how they were before you changed them in step 3. Use the backup file that you created in step 2 for reference.
- Restart the **PrivateArk Database service** in order to apply the initial values.
- In HA implementations only, in the Cluster Administration Utility, set the **PrivateArk Database resource to online**.
- Restart the Vault machine.

Upgrading the Vault Database

```
CAVaultManager UpgradeDB
```

This command will upgrade the Vault database in future versions of the CyberArk Vault.

Deleting the Vault Database

```
CAVaultManager DeleteDB
```

This command will delete all the information from the Vault database.

Note: This information cannot be retrieved after it has been deleted.

Recovering the Database Password

```
CAVaultManager RecoverDBPassword [/DBEmergencyPasswordFileName
<Filename>] [/DBNewPassword <Password>]
```

This command recovers the password that is used to access the Vault database. It uses the password specified in the emergency password file to retrieve the emergency database password which enables access to the Vault database, then generates a new database password and stores it in the file specified in the DatabaseConnectionPasswordFile parameter in DBParm.ini. The new password can either be specified by the user or a random password can be generated.

For example:

```
CAVaultManager RecoverDBPassword /DBEmergencyPasswordFileName
C:\VaultEmergency.pass
```

The above example will retrieve the emergency password stored in **C:\VaultEmergency.pass** then generate a new random database password and store it in the password file specified in the DatabaseConnectionPasswordFile parameter in DBParm.ini.

```
CAVaultManager RecoverDBPassword /DBEmergencyPasswordFileName  
C:\VaultEmergency.pass /DBNewPassword NewDBPwd
```

The above example will retrieve the emergency password stored in **C:\VaultEmergency.pass** then encrypt the new specified password, **NewDBPwd**, and store it in the password file specified in the DatabaseConnectionPasswordFile parameter in DBParm.ini.

Verifying LDAP Configuration

```
CAVaultManager LDAPVerify /ConfOnly /Verbose
```

This command carries out an integrity check on the LDAP configuration files and will check the connection with the LDAP component, and will display a detailed status report.

The `/ConfOnly` parameter will carry out an integrity check on the LDAP configuration files only, but will not check the connection status to the LDAP component.

Synchronizing the Vault Database

```
CAVaultManager SynchronizeDB [/SafePattern <Pattern>] /FilesSyncOnly  
/QuotaSyncOnly /Update /Force
```

This command synchronizes the Vault database after the backup files have been transferred to the Vault from backup data. It can synchronize only files or quotas in specific Safes, in the entire Vault or according to a Safe pattern. This command can either simulate synchronization or carry it out with or without confirmation from the user.

For example:

```
CAVaultManager SynchronizeDB /FilesSyncOnly /Update
```

The above example will synchronize the backup files that were restored in the Vault (using the RestoreDB command) with the restored Metadata. This command will be carried out, rather than simulated, and will prompt the user for confirmation during the process.

Recovering Backup Files

```
CAVaultManager RecoverBackupFiles [/BackupPoolName <BackupPoolName>]
```

This command uses the Vault's Recovery Private Key to access all the backup files in the Restored Safes folder and re-encrypt them with a new backup key when the original backup key cannot be used.

For example:

```
CAVaultManager RecoverBackupFiles /BackupPoolName BkpSvr1
```

The above example will recover the backup files from a Backup Pool called **BkpSvr1**, and re-encrypt them with a new accessible backup key.

Compiling a diagnostics report for the Vault database

```
CAVaultManager DiagnoseDBReport [/OutputFileName <FileName>]
```

This command compiles a diagnostics report for the Vault database.

Note: Use this command only in response to a request from CyberArk support.

For example:

```
CAVaultManager DiagnoseDBReport /OutputFileName
c:\CompanyVaultDiagnostics.txt
```

The above example will compile a diagnostics report of the Vault, and save the report in a text file called **CompanyVaultDiagnostics** stored in **c:**.

Replacing the current LDAP directory

```
CAVaultManager ReplaceLDAPDirectory /CurrentLDAPDirectory
<old_directory> /NewLDAPDirectory <new_directory> [/Update]
```

This command changes references in directory maps, users and groups from a current directory to a different one.

For example:

```
CAVaultManager ReplaceLDAPDirectory /CurrentLDAPDirectory Directory_1
/NewLDAPDirectory Directory_2
```

The above example changes references in directory maps, users and groups that define how external users are managed in the Vault from **Directory_1** to **Directory_2**.

CACert

The CACert utility prepares and manages the certificate that the Vault will use to create a secure channel to a client, so that users can authenticate to the third party securely. After the CACert utility has run, a log file is created which contains details about the process that was carried out.

You can specify any combination of optional parameters, although each parameter can only be used once.

CACert has the following usage:

```
CACert <command> [command parameters] /?
```

The usage is explained in the following table:

Parameter	Description	Mandatory
Request	Prepares a Certificate Signing Request (CSR) file.	
/ReqOutFile	The name of the request output file.	Yes
/ReqOutPrvFile	The name of the private key output file. Default value: The full pathname of the Server PrivateKey parameter as specified in <i>DBParm.ini</i> in the Privileged Account Security Reference Guide.	No
/KeyBitLen	The bit length of the output private key. Default value: 2048.	No
/Country	The name of the country to specify in the certificate. Use a 2-letter code.	No
/State	The full name of the State or Province to specify in the certificate.	No

Parameter	Description	Mandatory
/Locality	The name of the locality or city to specify in the certificate.	No
/Org	The name of the organization/company to specify in the certificate.	No
/OrgUnit	The name of the organizational unit name to specify in the certificate. For example, the department or section.	No
/CommonName	The Common Name to specify in the certificate. For example, the DNS name of the Vault. Note: Either the '/CommonName' parameter or the 'SubjAlt' parameter, or both, must be specified.	Yes
/SubjAlt	The subject alternative names. For example, "DNS:www.cyberark.com, IP:1.1.1.250". Note: Either the '/CommonName' parameter or the 'SubjAlt' parameter, or both, must be specified.	No
Install	Installs the certificate to be used by the Vault.	
/CertFileName	The full pathname of the certificate file to install.	Yes
Uninstall	Uninstalls the current Vault certificate, and generates and installs a new self-signed certificate.	
/Quiet	Uninstalls the Vault certificate without prompting the user for confirmation.	No
Import	Imports and installs a certificate from a ".pfx" file.	
/InFile	The full path of the file that contains the key and certificate to import (.pfx).	Yes
Show	Shows information about the current Vault certificate.	
/OutFormat	Specifies the output format: TEXT, PEM OR DER (default = TEXT).	No
Renew	Renews the current Vault certificate.	
/RenOutFile	The name of the certificate renewal output file.	Yes
SetCA	Handles CA certificates store.	
/CertStore	The certificate store to work with. If this parameter is omitted, the Vault trusted client CA's store is selected.	No
/List	Lists the subjects of the certificates in a store.	No
/Add	The name of the certificate file to add to the store.	No
/Remove	The name of the certificate file to remove from the store.	No
/?	Lists the available options.	

DBConv

This utility is used to upgrade the database when a new version of the Vault is installed in an existing Vault environment. The upgrade is relevant for upgrading version 3.50 of the Vault and below to version 4.0.

During the process, the System Safes are upgraded first, after which all the other Safes can be upgraded. The utility then copies a log of processes that it carried out to a file called DBConv.log which is located in the Server's directory.

It is highly recommended to view the DBConv.log file after running DBConv in order to check for any reported errors.

Notes:

- i. After running DBConv, all existing requests become invalid.
- ii. If the Vault contains more than 256 Safes:
 - In DBParm.ini, increase the number that represents the maximum number of Safes in the ConcurrentLoadedSafes parameter, or,
 - Specify a Safe pattern when you run DBConv, so that you don't upgrade all the Safes at the same time.

DBConv has the following usage:

```
DBConv <Safe Name Wildcard>
      [ /Simulate | /Execute ]
      /?
```

The usage is explained in the following table:

Parameter	Description
Safe Name Wildcard	Name of the Safe(s) to be upgraded.
/Simulate	A simulation of the upgrade process will be carried out and any errors that may happen when the actual upgrade is done will be reported.
/Execute	The upgrade process will be carried out and any necessary changes required by the upgrade process will be made, including altering required data in upgraded Safes and altering identical folder names and filenames.
/?	Lists the available options.

Usually, during upgrading, the following DBConv commands are issued:

```
DBConv * /simulate
```

This command causes DBConv to simulate the upgrade process, and report any errors that may happen when the actual upgrade process will take place. Errors are reported to the DBConv.log file.

```
DBConv mark* /execute
```

This command performs the actual upgrade process on Safes that match the 'mark*' wildcard (for example, 'Marketing', 'Markets', and 'Marks Safe'), altering any required data. Again, errors are reported to the DBConv.log file.

Recover

This utility is used to recover information from a Safe's external files in case of loss or corruption of that Safe. The files are decrypted and saved as readable files.

Recover has the following usage:

```
Recover [in_file_name] [out_directory_name] [key_path]
```

The usage is explained in the following table:

Parameter	Description
in_file_name	The name of the file to be recovered. Use wildcards for multiple files.
Out_directory_name	The name of the folder where the utility will store the recovered files.
Key_path	The path of the Recovery key. Note: This key is on the Master CD only.

For example:

```
Recover D:\cyberark\safes\research\*.* D:\research F:
```

The above example will cause the Recover utility to recover all files in the Safe called Research, and store them, unencrypted, in the D:\research folder. The utility will use the Recovery key on the Master CD, which is on the F: drive.

SafeRecover

This utility is used to recover the contents of a Safe.

SafeRecover has the following usage:

```
Saferecover [safe] [output directory] [keys directory]
```

The usage is explained in the following table:

Parameter	Description
Safe	Name of the Safe to recover. Use wildcards for multiple Safes.
Output directory	Name of the folder in which the utility will store the recovered Safe contents.
Keys directory	The path of the Recovery key. Note: This key is on the Master CD only.

For example:

```
saferecover Research C:\Research F:
```

The above example will cause the SafeRecover utility to recover the Safe called Research, and store the recovered files in the C:\Research folder. The utility will use the Recovery key on the Master CD, which is on the F: drive.

ChangeServerKeys

ChangeServerKeys is used to change the main keys of the Vault server, and re-encrypt Vault data and metadata that were encrypted with the old keys. You can use new keys that support a different algorithm, and change the encryption algorithm used to encrypt the Safes. This utility can also be used to re-encrypt Vault data and metadata with encryption keys on the HSM. The activities carried out by the utility are written to a log file, and stored in the same folder as the utility.

Before running this utility, backup the following folders and files in the Vault:

- The Safes folder
- The Vault installation and LDAP folders
- The RADIUS secret file
- The database password and emergency password
- The PKI Vault private key
- The entropy file

After this utility has been run, the current keys for the Vault database become obsolete, and must be replaced with the new Vault keys.

ChangeServerKeys has the following usage:

```
ChangeServerKeys      [New Master Keys Dir]
                      [DB Emergency Password File Name]
                      [HSM Keyset]
```

The usage is explained in the following table:

Parameter	Description	Acceptable Value
For keys stored on the Vault machine:		
New Master Keys Dir	The location of the new master CD which contains the keys that will be used to re-encrypt the Vault data and metadata.	Path
DB Emergency Password File Name	The full pathname of the file where the encrypted emergency password for database access is stored.	Path
For keys stored on an HSM device:		
New Master Keys Dir	The location of the new master CD which contains the keys that will be used to re-encrypt the Vault data and metadata. Note: After generating a new server key on the HSM, you can replace it by supplying the current master CD path here instead of actually supplying a new one.	Path
DB Emergency Password File Name	The full pathname of the file where the encrypted emergency password for database access is stored.	Path
HSMKeyset	The HSM keyset, including the key generation version. For example, HSM#3.	Keyword

For example:

```
changeserverkeys C:\newkeys C:\VaultEmergency.pass
```

The above example will cause the ChangeServerKeys utility to replace the current encryption key in the Vault with the new operator keys, located in **C:\newkeys**. The utility will then use the new keys to re-encrypt the password used to access the Vault database, which is stored in an emergency password file called **C:\VaultEmergency.pass**.

However, the new keys will not take effect until the following parameters are changed in DBParm.ini.

- ServerKey
- RecoveryPubKey
- RecoveryPrvKey

If keys with a different encryption algorithm were used, the following parameters must be changed as well:

- SymCipherAlg
- AsymCipherAlg

Server Keys

The CyberArk Vault's sophisticated encryption mechanism is designed to ensure maximum security at all times and to provide recovery capabilities, when needed.

There are two external keys associated with the Server. Since these keys enable access to the Server and the data stored within, it is recommended to save the Server keys on a removable media, such as a disk or CD, so that they may be safely secured in a physical Safe.

Server Key

The Server Key is the key used to “open” the Vault, much like the key of a physical Vault. The key is required to start the Vault, after which the Server key can be removed until the Server is restarted. When the Vault is stopped, the information stored in the Vault is completely inaccessible without that key.

The path to the Server key is defined in DBParm.ini.

Recovery Key

The Recovery Key is used to restore data stored in the Vault in the event that the Vault is not operational. This key should only be used in those very rare cases of failure where the Vault is not operational and cannot be repaired in the required time frame or when the key is forgotten to a Safe defined with an external key.

The Recovery Key is essential for the Master User to log on to the Vault.

The Recovery key is an asymmetric key composed of the **Public Recovery Key** and **Private Recovery Key**.

Private Recovery Key

The Private Recovery Key is required for the Master User to log on and to open the Safes in the event of Vault recovery. This Key should be stored separately from the Server in a secured place, such as on a disk or CD, in a physical vault.

To recover the data that is stored in the Safe, the Private Recovery Key should be used with a recovery utility. For more information about the recovery utilities, refer to *Server Utilities*, page 891.

Public Recovery Key

The Public Recovery Key enables the recovery option in the Safe. This enables the Safe to be opened with the Private Recovery Key.

The Public Recovery Key is usually stored with the Server Key and the path to the key is defined in the DBParm.ini file.

Location of the Server Keys

The Server key and the Public Recovery key are required to start the Server. There are two ways of providing those keys upon startup, as follows:

- The keys can be permanently installed on the Server host. This enables you to configure the Server to start automatically.
- The keys can be stored on a removable media, such as a disk or CD. After starting the server, the keys can be removed again to their safe storage.

It is recommended to place the keys on a removable media, so they can be stored in a safe location when they are not needed.

The CyberArk Vault package contains two sets of the following CD keys:

- **Operator CD** – Contains the server key and the public recovery key, which are required to start the server. This CD can be used to operate the Server.
- **Master CD** – Contains the server key, the public recovery key and the private recovery key. This CD should be used for recovery and for Master logon.

The CyberArk Client

The PrivateArk Client is a windows interface that communicates with the CyberArk Vault Server enabling you to perform administrative activities in the Enterprise Password Vault. It is installed on the same machine as the Enterprise Password Vault and can also be installed on users' workstations. This section describes the structure of the PrivateArk Client and its configuration options.

PrivateArk Client Components

This section describes the files and drivers that make up the PrivateArk Client.

The main Client executable (Arkui.exe)

The main Client executable handles the Client User interface and the communication with the CyberArk Vault Servers and displays any alerts or information sent to the client from the Servers and the PrivateArk Workspace.

The main library (PASS600.dll)

The CyberArk Vault's main library is used by all Client components to communicate with the Server and to perform other common tasks.

The PrivateArk Workspace Windows Integration (Safeview.dll)

The PrivateArk Client integrates with the windows file system to enable users to work transparently with the files stored in the Safe. Once a Safe is opened, this component reflects the entire contents of the Safe in the Workspace, enabling the user to open and save files from any application that he or she is working with.

PrivateArk Links (PALink.exe)

The PrivateArk Links executable manages links to files inside the Safe.

Office Extensions

The Microsoft Office extensions provide easy access to Vaults and Safes when working with Microsoft Office applications.

PrivateArk Client Files

PrivateArk Client Binary Files

The executable files of the PrivateArk Client are located in the Client installation directory. The default path to the PrivateArk Client installation directory is as follows:

```
\Program files\PrivateArk\Client
```

The PrivateArk Client DLL files are located in the system directory.

PrivateArk Client Configuration Tokens

The PrivateArk Client configuration tokens are located in configuration files in the client installation directory, or in the registry:

- Arkui.ini – The PrivateArk Client configuration file.
- Registry entries:
 - HKLMSOFTWARE\CyberArk\PrivateArk\ConfigInfo
 - HKLMSOFTWARE\CyberArk\PrivateArk\Client
 - HKCU\SOFTWARE\CyberArk\PrivateArk\Client

The Client configuration tokens are managed from the Client interface.

PrivateArk Client Microsoft Office Extensions

The Microsoft Office extensions are add-ins to Microsoft Office applications.

These extensions enable easy access to and storage of Office files stored in the Vault. These extensions are optional and are not required to operate the PrivateArk Client.

PrivateArk Workspace

The PrivateArk Workspace is the area on the hard disk where files that are retrieved from the Safe are stored. The default location of the PrivateArk Workspace is under the user profile directory.

PrivateArk User Pictures

The PrivateArk Client stores pictures of the Client User and other Users with which he or she share Safes. These pictures are displayed with the relevant User's activities in the monitoring and inspection windows of the Client interface.

The pictures are automatically downloaded from the Server to the pictures directory. The pictures directory is located in the Client installation directory.

Note: The PrivateArk Workspace and the PrivateArk User pictures are stored under the User Profiles.

PrivateArk Workspace

To ensure maximum security, the Vault is installed on a dedicated and isolated computer. This means that there are no other applications installed and operating on the same computer as the CyberArk Vault Server. Thus, when a file is to be updated, it must first be taken out of the Safe and then updated. Once the update is completed, the file can be returned to the Safe.

When retrieving a file from the Safe, it is automatically transferred to a location on the User's workstation, which is called the PrivateArk Workspace.

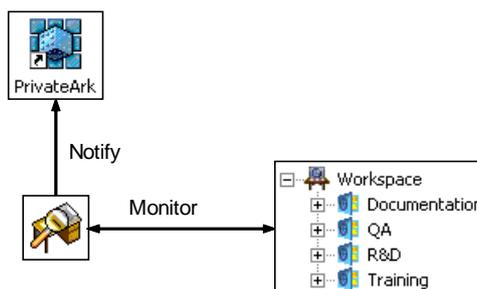
A directory is automatically created on the PrivateArk Workspace for each of the User's Safes, identifiable by the PrivateArk Workspace icon. Files retrieved from a Safe are downloaded to that Safe's directory in the PrivateArk Workspace.

When the Safe is closed, all files in the PrivateArk Workspace are automatically returned to the Safe.

A User can also save files directly in the PrivateArk Workspace instead of storing them in the Safe. When the Safe is closed, these files are automatically transferred to the Safe.

Note: The temporary files belonging to the retrieved files are under a hidden folder in the Workspace.

The PrivateArk Client Disk Monitor tracks the files in the PrivateArk Workspace and notifies the User of any activity. This Disk Monitor sends notifications to the PrivateArk Client, which are displayed in the Client's Application Monitor.



Workspace Files

The Workspace contains a folder for each Safe that the User has accessed. If a User accesses more than one Safe on more than one Vault and these Safes have the same name, then the Workspace directory is comprised of both the Vault and Safe names.

Each folder in the Workspace contains the files that are downloaded from the relevant Safe. In addition to these files, each folder contains three permanent configuration files, as follows:

- Desktop.ini – This file contains the definitions of the Workspace icons.
- Paws.ini – This file stores information about the files retrieved from the Safe to the Workspace, such as, the Safe folder in which the files are located, the Safe's Server, and so on.
- Folders.ini – This file contains a list of the Safe's folders.

Note: The workspace configuration files are hidden files.

PrivateArk Client Configuration

The CyberArk Vault configuration settings can be divided into three different groups:

- **Personal** – These settings are user specific.
- **Global** – These settings include site and organizational settings
- **Fixed** – These are predefined fixed settings, usually for the Vault's internal use.

Personal Settings

Each User maintains control over personal settings, such as the following:

- The User name and timeout
- Interface options, such as views and colors
- Last store and retrieve information

The personal settings of each User are stored under the user profile in the registry or in an ini file. This ensures that if different Users log onto the same workstation, each will have his own settings.

Global Settings

Global settings are those parameters that are identical for all Users or Groups of Users in the organization, such as Vault definitions.

CyberArk Vault Global Client Configuration enables you to define Vault parameters once and then make these parameters available to all Users.

Instead of the tedious task of defining Vault(s) with exactly the same parameters on each computer that has a PrivateArk Client installed, you specify them only once then save the parameters in a global configuration file. When the User invokes the PrivateArk Client, the Vaults that have been defined in the global configuration file appear on the Users' screen. This feature reduces system administration time and streamlines Vault management.

The global settings are usually stored on a network drive that is accessible by all users. However, there are cases where the global settings are stored on the same workstation that the user is working on. This is the case when several Users are working on the same workstation and they all share the global settings or when the CyberArk Vault is installed in a terminal server environment.

The global settings are controlled and managed by the Vault administrator in a global configuration file. The User cannot update any Vault parameters or delete any Vaults defined in that file. Nevertheless, it is possible for Users to define Vaults from their own PrivateArk Client. These Vaults will not be stored in the global configuration file, but in the User's personal settings. The Vaults, therefore, will only appear on the User's screen.

Fixed Settings

The fixed settings are set during installation and are mainly for the CyberArk Vault's internal use. Among the fixed settings are the locations of the Vault files and installation parameters.

The fixed settings are stored in an ini file or under

`HKEY_LOCAL_MACHINE\Software\CyberArk\PrivateArk` in the registry.

Implementing Global Client Configuration

In order for Users to work with global CyberArk Vault settings, the settings should be located where all Users working with them can access, in an ini file or registry.

During installation, the administrator decides whether or not to implement this feature. If you decide not to use Global Client Configuration, the CyberArk Vault installation will install the PrivateArk Client according to default parameters. You can change them in the registry or ini files after installation is complete.

To Install the PrivateArk Client with Global Client Configuration

1. Follow the PrivateArk Client Installation procedure, as detailed in the Privileged Account Security Installation Guide, until Step 12.
2. In the Global Client Configuration window, select **Use Global Configuration**.
3. Select the relevant option button to determine whether the global settings will be stored in an ini file or in the Registry.
4. If you choose to store the global configurations in an ini file, specify its proposed location on the network, then click **Next** and complete installation as in the Privileged Account Security Installation Guide.

Defining a Vault for Global Client Setting

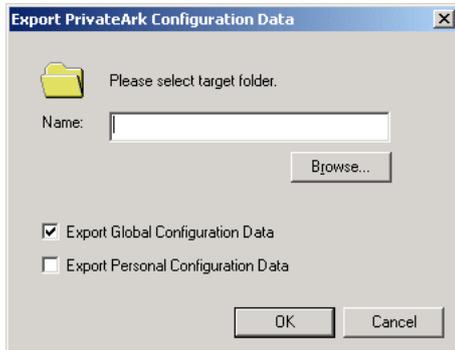
Definitions made in the PrivateArk Client interface are automatically written to the personal settings.

To use them as the global setting, export them from the personal setting and then load them as the global setting.

1. Log onto the PrivateArk Client.
2. Define the Vault(s) that will be available to all users.
3. Log on to each Vault and then log off.
Note: During the first logon to the Vault, the client receives the Vault ID from the Vault server. This step is, therefore, very important to prevent later confusion.
4. Create the global setting ini file or registry, as described below.
5. Configure the PrivateArk Clients to use global settings during installation of the client, or later using the PAConfig utility. For more information, refer to *Changing Default Global Settings*, page 911.

To Prepare an ini File for Global Setting

1. From the Tools menu, select Administrative Tools, then **Export Configuration Data**; the Export Configuration Data window appears.



2. Click **Browse**, then select the location on the network in which to save the ini file.
3. Select **Export Global Configuration Data** to export the properties of the Vault list to the specified location.
4. Select **Export Personal Configuration Data** to export your personal configuration parameters to the specified location.
5. Click **OK**; the selected data is exported. When the process is complete, a confirmation window appears. The default name of the ini file is displayed in this window. If you wish, you can change the filename.

Note: The location on the network where you save the ini file must be accessible by all Users.

To Define a Vault for Global Client Configuration using the Registry

1. Export the following Server parameters:
HKCU\Software\CyberArk\PrivateArk\Client\Servers
2. In Windows Explorer, open the registry file and change the target location from HKEY_CURRENT_USER\... to HKEY_LOCAL_MACHINE\..., and save the file.
3. On the machine where the global registry should be set, double-click the filename; the Global Client Configurations will be copied to the following location:
HKLM\Software\CyberArk\PrivateArk\Client\Servers
4. Delete the key that is located at the original location (HKEY_CURRENT_USER\).

Changing Default Global Settings

The PAConfig utility enables you to update the default client settings configuration. You can introduce this into each User's network logon script so that the update is implemented as soon as the User logs on to the network.

PAConfig has the following usage:

```
PAConfig </NOGLOBAL | /infile <Path> | /registry >
```

This usage is explained in the following table and example:

Parameter	Description
/noglobal	Do not use Global Client Configuration.
/infile <Path>	Use ini based Global Client Configuration.
/registry	Use Registry based Global Client Configuration. This is relevant only in a Remote Desktop Services (Terminal Services) environment.

For example:

```
PAConfig /infile Z:\PrivateArk\GlobalSettings.ini
```

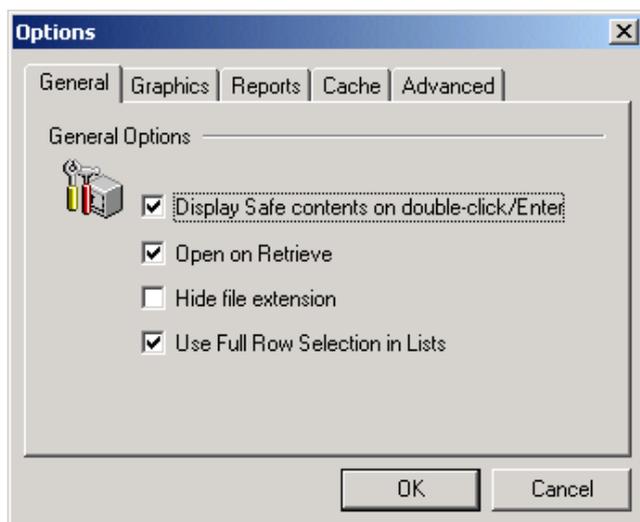
The above example updates the PrivateArk Client to Global Client Configuration, using an ini file. This file is stored on drive Z: on the network, under the PrivateArk folder in an ini file called GlobalSettings.

PrivateArk Client Options

The PrivateArk Client options enable you to configure your PrivateArk Client and determine certain aspects of its interaction with the CyberArk Vault.

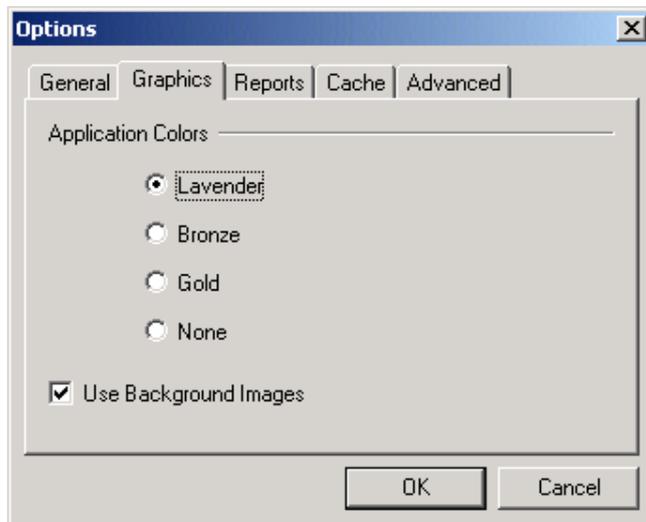
To display the Options window, from the **Tools** menu, select **Options**.

General Tab



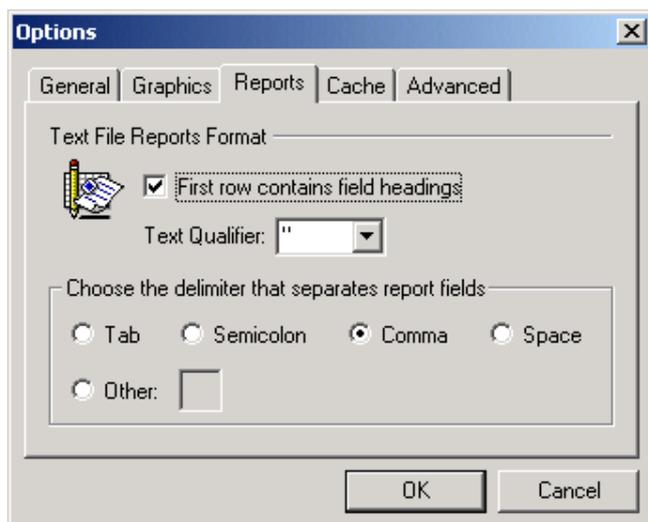
Option	Indicates ...
Display Safe contents on double-click/Enter	Select to open the Safe and display its contents, or clear to open the Safe, but not to display its contents.
Open on Retrieve	Select to retrieve files to the PrivateArk workspace and open them, or clear to retrieve them to the workspace but not open them automatically.
Hide file extension	Select to hide the file extension, or clear to display it.
Use Full Row Selection in Lists	Select to highlight the entire row on file selection, or clear to highlight just the file name.

Graphics Tab



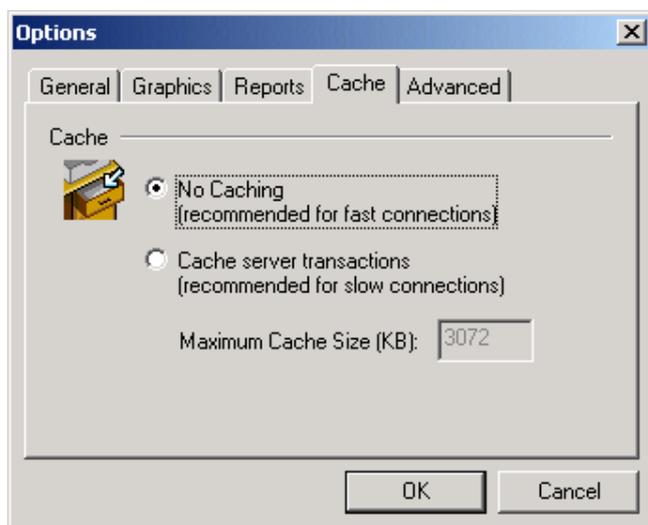
Option	Indicates ...
Application colors	The background color of the PrivateArk Client interface.
Use Background Images	Display the background image in the Vault view.

Reports Tab



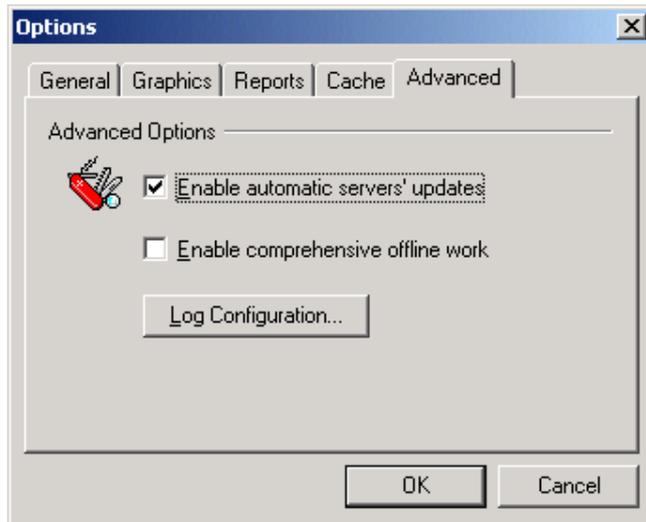
Option	Indicates ...
First row contains field headings	Whether or not the first row of the report will contain the field headings.
Text Qualifier	The symbols that surround the text in the fields of the report.
Delimiter	The mark that separates the various fields in the report.

Cache Tab



Option	Indicates ...
No caching	Update the Client every time an activity occurs in the Vault.
Cache server transactions	The Client will not update its information automatically, but stores the data it receives from the Server until the user presses F5.

Advanced Tab



Option	Indicates ...
Enable automatic servers' updates	Receive Accessed files counters and confirmation information automatically.
Enable comprehensive offline work	Displays the Safes and their contents that have been entered from your terminal using the PrivateArk Client, even when there is no connection to the Vault.
Log Configuration	The various types of log records that the PrivateArk Client can generate.

The PrivateArk Client in a Remote Desktop Services (Terminal Services) Environment

The PrivateArk Client can be installed on a Remote Desktop Services (Terminal Services) server, offering a more accessible, yet more controlled, way of working. The Vault administrator defines the Vault parameters once on the Remote Desktop Services (Terminal Services) server, to which every User gains access when he logs on. Files are retrieved from the Vault to a PrivateArk Workspace on the server that is unique to the User's logon account, thus maintaining the Vault's typically high level of privacy and security.

To Install the PrivateArk Client on a Remote Desktop Server

1. Log onto the Remote Desktop server as the Administrator.
2. Install the PrivateArk Client on the Remote Desktop server.

Note: It is strongly recommended to use Global Configuration.
3. After declaring the Vault(s), all Users must log off the Remote Desktop server, then log on again in order for the settings to take effect.

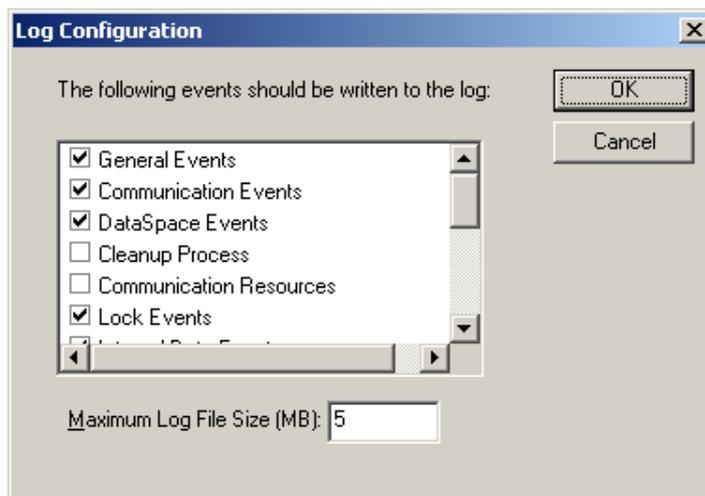
The PrivateArk Client Log File

The PrivateArk Client Log file records events that occur during each session. This enables you to track events and identify common errors.

As the log file tracks events on the PrivateArk Client, you can determine which events are recorded in the log for different workstations. The log file is stored in the workstation's PrivateArk folder as PALog.txt.

To Configure the Log File

1. From the **Tools** menu, select **Options**, then click the **Advanced** tab; the Advanced Options dialog box appears.
2. Click **Log Configuration**; the Log Configuration window appears.



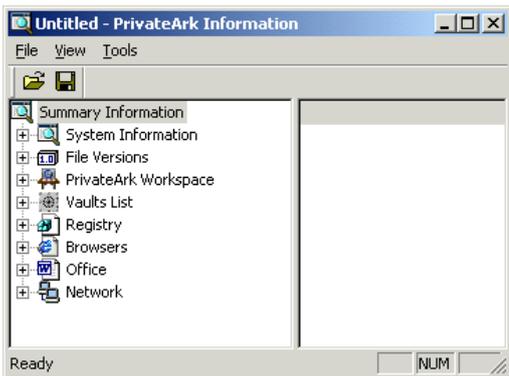
3. Select the events to record in the log file, then click **OK**; all activities that occur during each session will be recorded in the PrivateArk Client Log file.

PrivateArk Information

PrivateArk Information is a set of files that contain all the details needed by the CyberArk support team to enable you to work smoothly and intuitively with the Vault. If you have a problem and cannot solve it yourself, save the PrivateArk Information and send it to the support team, they can isolate hitches and resolve issues in the shortest possible time.

To Access PrivateArk Information

1. In the PrivateArk Administration Client, from the **Help** menu, select **About PrivateArk**; the About PrivateArk window appears.
2. Click **PrivateArk Information**; the PrivateArk Information window appears.



3. From the **File** menu, select **Save**; the Save As dialog box appears.
4. Name the file, leaving the extension; then click **OK**.
5. From the **File** menu, select **Send**; Microsoft Outlook automatically opens.
6. Click **Send** to send the Information file to the CyberArk support team.

System Configuration

This section introduces you to system configuration, and describes how to configure the Enterprise Password Vault.

The following Privileged Account Security solution components can be configured in the system configuration interface in the PVWA:

- Password Vault Web Access
- Privileged Session Manager
- Central Policy Manager
- Event Notification Agent
- Transparent user management (LDAP)
- Privileged Endpoint Protection

The following Privileged Account Security solution components are configured in the relevant configuration files:

- CyberArk Vault
- PrivateArk WebClient
- Remote Control Agent
- Password Vault Web Access application configurations
- Privileged Session Manager application configurations
- On-Demand Manager application configurations
- Disaster Recovery Vault
- Password Upload Utility

Configuring the System through PVWA

Authorized users can configure certain components of the Privileged Account Security solution in the PVWA. This provides seamless administration features, while still maintaining granular access to secure and privileged information.

These configurations can be viewed and modified in the System Configuration page, which can be viewed by users with membership in the following group:

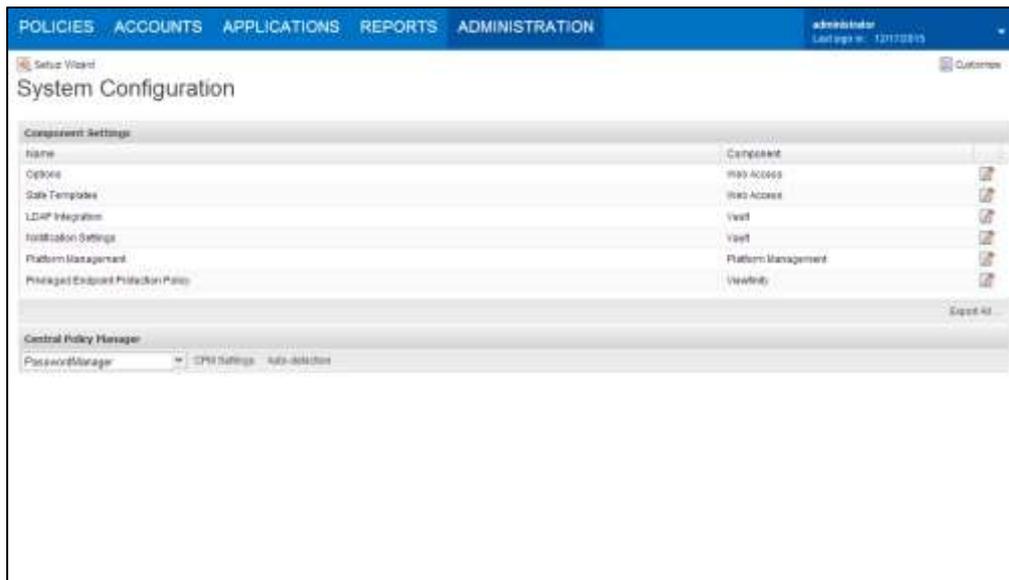
- Vault Admins

The System Configuration page displays the system configuration that can be configured through the PVWA. Users can configure the following:

- **Options** – Users can configure general system settings, including how dual control works, ticketing systems, privileged session management, authentication, and interface options.
- **Safe Templates** – Defines default settings for new account Safes.
- **LDAP integrations** – Configures LDAP integration with the Privileged Account Security solution.
- **Notification Settings** – Configures notification settings and defines automatic notifications.
- **Platform Management** – Enables users to manage technical settings for target and service account platforms.
- **Privileged Endpoint Protection Policy** – Enables you to access the CyberArk Viewfinity Console.

In the Central Policy Manager section, users can select a CPM and configure the following settings for it:

- **CPM Settings** – The general CPM parameters.
- **Auto-Detection** – Parameters that configure auto-detection processes.

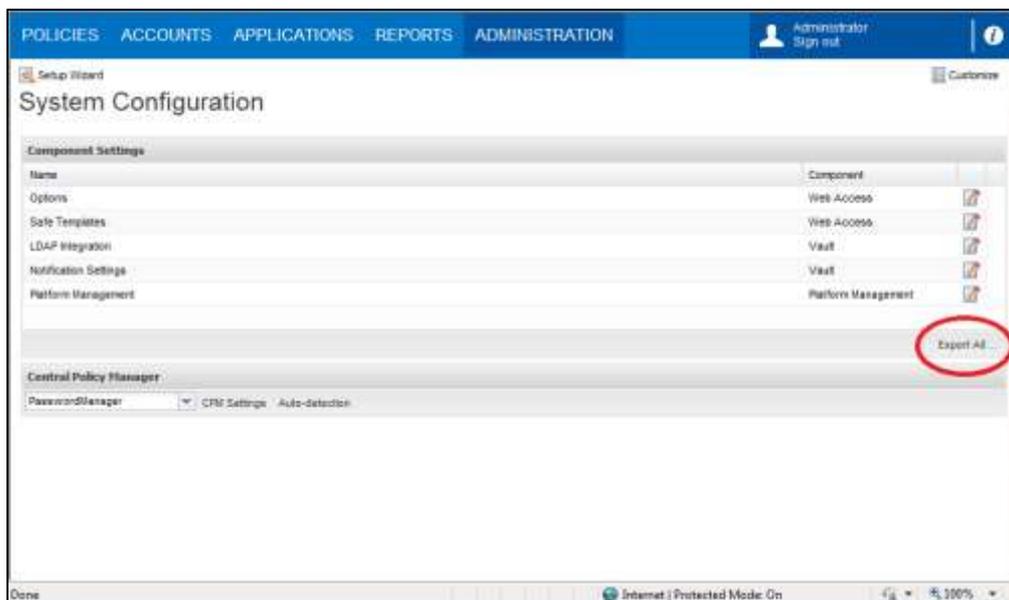


Exporting Configuration

The PWA configuration files contain parameters that enable you to work smoothly and intuitively with the Vault. If you have a problem and cannot solve it yourself, you can export the file contents and send it to the CyberArk support team who will isolate hitches and resolve issues in the shortest possible time.

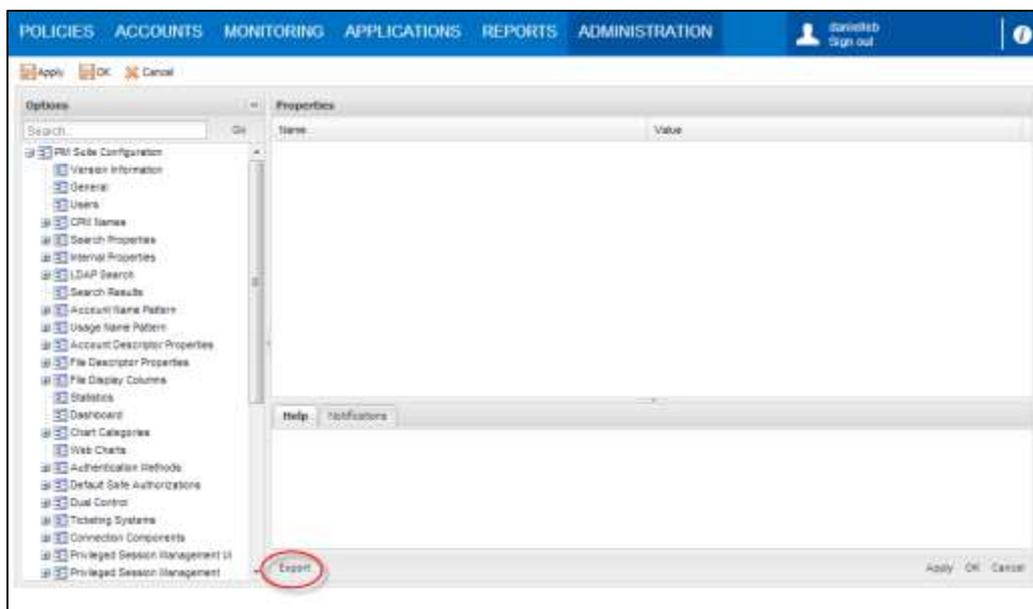
To Export Configuration Files

1. In the **System Configuration** page, click **Export All**

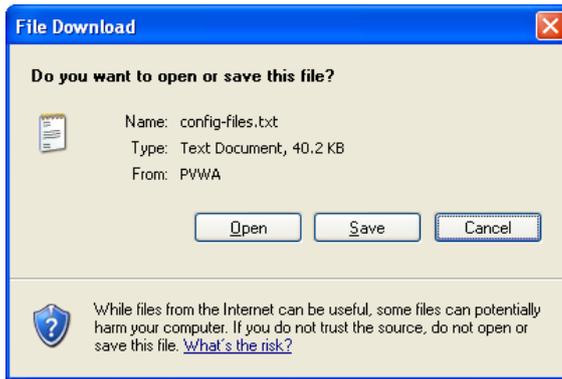


or,

In one of the configuration pages, click **Export**.



The File Download window appears.



2. Click **Open** to open the exported configuration file immediately,
or,
Click **Save** to specify where the file will be saved.

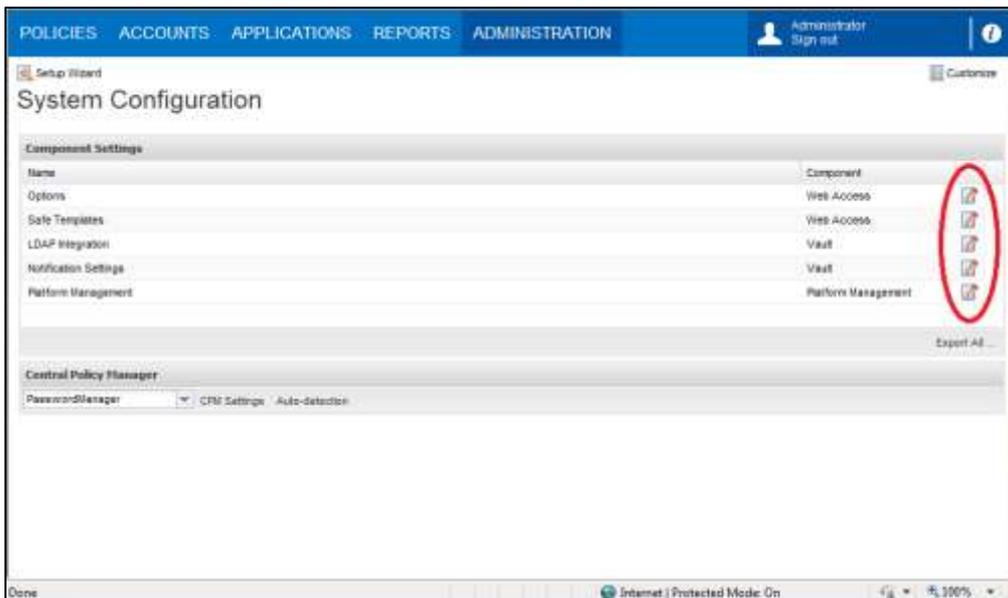
In the System Configuration for each configuration file or platform, you can export the details of that file.

Editing System Settings

The System Configuration page lists the system settings that can be modified. Each line displays an **Edit** icon which enables users to access settings to edit.

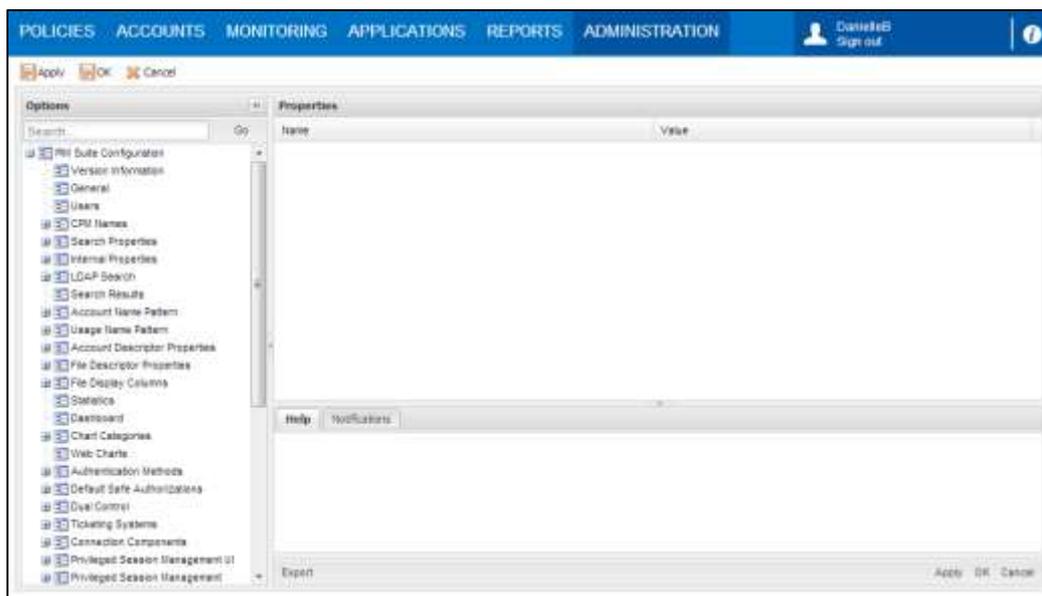
To Edit System Configuration

1. In the System Configuration page, click the name of the configuration option or the **Edit** icon on the line of the settings to edit.



The settings page for the relevant system settings or platform appears.

The following example shows the Options settings page that appears when the user selects **Options** in the System Configuration page.



This page is divided into the following areas:

- **Parameters list** – This area displays the parameters in the selected configuration file or platform.
 - **Properties list** – This area displays the parameter properties and their values. Properties that can be modified are enabled, while properties that cannot be modified are disabled.
 - **Help/Errors** – This area displays the Help tab and the Errors tab. The Help tab displays a description of the selected parameter or property. The Errors tab displays any errors that occurred when the selected file or platform was loaded.
 - **Activities bar** – This area displays links to the following activities:
 - **Export** – This enables you to export the displayed set of parameters directly to a text file.
 - **Apply** – This enables you to save the new parameter values in the configuration file or platform and apply them immediately after you define them.
 - **OK** – This enables you save the new parameter values in the configuration file. Web Access changes will be applied immediately, while CPM and platform files will be updated the next time the CPM configuration is updated.
 - **Cancel** – This displays the main System Configuration page where you can see all the configuration files and platforms.
2. In the Properties list, select the value of the property to modify. If it is configurable, you will either be able to specify a new value or to select a preconfigured value from a drop-down list.
 3. Click **Apply** to apply the configuration changes immediately, or,
 - Click **OK** to save the changes and display the System Configuration page.

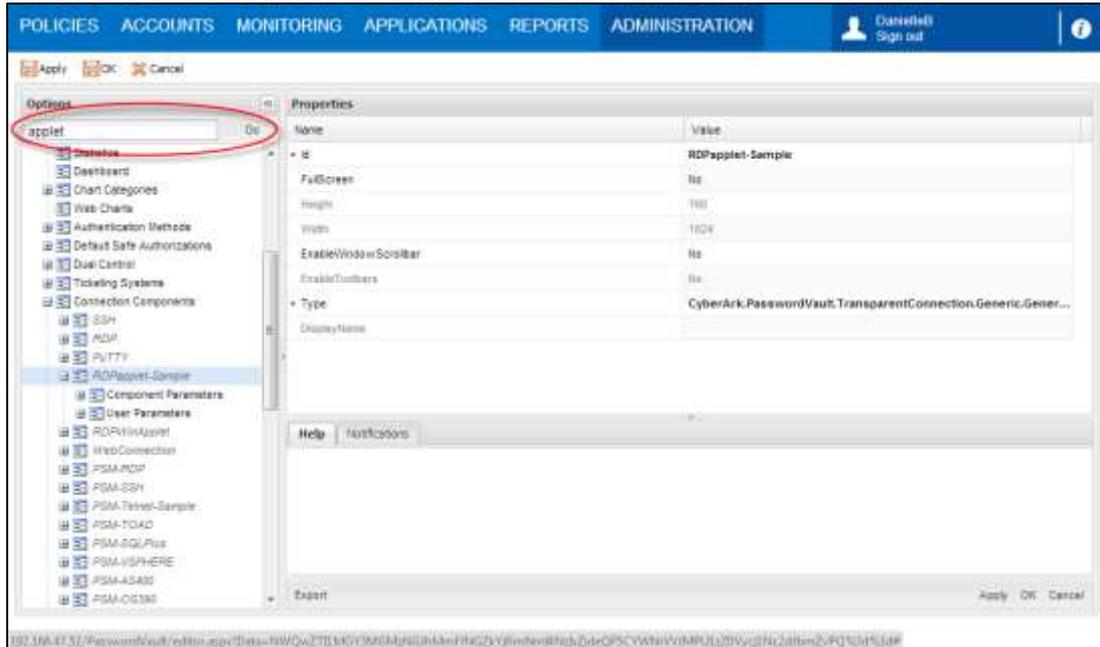
Searching for Configuration Parameters and Values

You can search for configuration parameters and values in the Options settings page, enabling you to find parameters quickly.

To Search for a Parameter or Specified Value

- In the Options settings page, in the Search box, specify the parameter or value to find, then click **Go**; the search finds the parameters or values that match the text you specified

The following example shows the search result for **applet**.



- To move to the next search result, click **Go** or press **F3** until the parameter you are searching for is displayed.
- To search for a parameter or value in a specific section, select the section in the settings page, then click **Go** or press **F3**.

Appendices

This chapter contains the following appendices:

- Appendix A: Account Properties
- Appendix B: Creating User Credential Files
- Appendix C: Configuring Debug Levels
- Appendix D: Managing Platforms for Groups
- Appendix E: Adding Accounts with SSH Keys using the AccountUploader Utility
- Appendix F: Accessing Target Machines through PSMP

Appendix A: Account Properties

The following table displays the account properties that are created by the CPM installation. Users require the 'Update password properties' authorization in the Safe to add or change account property definitions.

Note: The semicolon (;) and hash (#) characters indicate the beginning of a remark. However, if these characters appear between quotation marks ("") they are considered to represent a parameter.

Application Account Properties

CyberArk Accounts

Parameter	Description	Acceptable Values	Default Values
Required Properties			
Platform name	The platform name that is relevant for this password, and is specified in the platform.	Platform name	CyberArk
Address	The address of the remote machine where the password will be used.	IP/DNS address, Windows domain or machine name, TNS name.	
User Name	The name of the user on the remote machine who this password belongs to.	User name	
Optional Properties			
Port	The Vault IP port.	Port	1858
Timeout	The number of seconds to wait for a Vault to respond to a command before a timeout message is displayed.	Number	30
ReconnectPeriod	The number of seconds to wait before the sessions with the Vault is re-established.	Number	60
ProxyType	The type of proxy through which the Vault is accessed.	HTTP, HTTPS, SOCKS4, SOCKS5, NOPROXY	NOPROXY
ProxyAddress	The proxy server's IP address. This is mandatory when using a proxy server.	IP/DNS address	
ProxyPort	The Proxy server IP port.	Port	
ProxyAuthDomain	The domain for the Proxy server if NTLM authentication is required.	String	
ProxyUser	User for Proxy server if NTLM authentication is required.	String	

Parameter	Description	Acceptable Values	Default Values
ProxyPassword	The password for Proxy server if NTLM authentication is required.	Password	
BehindFirewall	Whether or not the Vault is accessed via a Firewall.	Yes/No	No
UseOnlyHTTP1	Whether or not to use only HTTP 1.0 protocol. Valid either with proxy settings or with BehindFirewall.	Yes/No	No

SAP Accounts

Parameter	Description	Acceptable Values	Default Values
Required Properties			
Platform name	The platform name that is relevant for this password, and is specified in the platform.	Platform name	SAP
Address	The address of the remote machine where the password will be used.	IP/DNS address, Windows domain or machine name, TNS name.	
User Name	The name of the user on the remote machine who this password belongs to.	User name	
SAP System Number	The SAP system number.	System number	
SAP Client	The SAP Client	Client name	

Facebook Accounts

Parameter	Description	Acceptable Values	Default Values
Required Properties			
Platform name	The platform name that is relevant for this account, and is specified in the platform.	Platform name	Facebook
Address	The address of the website where the password will be used.	IP/DNS address	www.facebook.com
User Name	The name of the user on the website who the password belongs to.	User name	

Operating System Account Properties

Windows Domain Account Passwords

Parameter	Description	Acceptable Values	Default Values
Required Properties			
Platform name	The platform name that is relevant for this password, and is specified in the platform.	Platform name	WinDomain
Address	The address of the remote machine where the password will be used. This is not mandatory.	IP/DNS address, Windows domain or machine name, TNS name.	
User Name	The name of the user on the remote machine who this password belongs to.	User name	
Optional Properties			
LogonDomain	The domain where the account will be used.	String	
UserDN	User's distinguished name.	String	
Port	The port that will be used to access the remote machine.	Number	
Linked Accounts			
ExtraPass1Name	The name of the linked logon password object.	Valid password object name	
ExtraPass1Safe	The safe name where the linked logon password object is stored	Safe name	
ExtraPass1Folder	The folder name where the linked logon password object is stored.	Folder name	
ExtraPass3Name	The name of the linked reconcile password object.	Valid password object name	
ExtraPass3Safe	The safe name where the linked reconcile password object is stored.	Safe name	
ExtraPass3Folder	The folder name where the linked reconcile password object is stored.	Folder name	

Windows Local Accounts

Parameter	Description	Acceptable Values	Default Values
Required Properties			
Platform name	The platform name that is relevant for this password, and is specified in the platform.	Platform name	WinServerLocal
Address	The address of the remote machine where the password will be used.	IP/DNS address, Windows domain or machine name, TNS name.	
User Name	The name of the user on the remote machine who this password belongs to.	User name	
Optional Properties			
LogonDomain	The domain where the account will be used.	String	
Location	The physical location of the Windows machine.	String	
OwnerName	The full name of the desktop owner.	String	
Linked Accounts			
ExtraPass1Name	The name of the linked logon password object.	Valid password object name	
ExtraPass1Safe	The safe name where the linked logon password object is stored	Safe name	
ExtraPass1Folder	The folder name where the linked logon password object is stored.	Folder name	
ExtraPass3Name	The name of the linked reconcile password object.	Valid password object name	
ExtraPass3Safe	The safe name where the linked reconcile password object is stored.	Safe name	
ExtraPass3Folder	The folder name where the linked reconcile password object is stored.	Folder name	

Windows Local Desktops Accounts

Parameter	Description	Acceptable Values	Default Values
Required Properties			
Platform name	The platform name that is relevant for this password, and is specified in the platform.	Platform name	WinDesktopLocal
Address	The address of the remote machine where the password will be used.	IP/DNS address, Windows domain or machine name, TNS name.	
User Name	The name of the user on the remote machine who this password belongs to.	User name	
Optional Properties			
LogonDomain	The domain where the account will be used.	String	
Location	The physical location of the Windows machine.	String	
OwnerName	The full name of the desktop owner.	String	
Windows Services			
MasterPassName	The name of the password object that contains the logon credentials that are required to log onto the Windows account.	Valid password object name	
ServiceName	The name of the Service that will use the password.	Service name	
Linked Accounts			
ExtraPass1Name	The name of the linked logon password object.	Valid password object name	
ExtraPass1Safe	The safe name where the linked logon password object is stored	Safe name	
ExtraPass1Folder	The folder name where the linked logon password object is stored.	Folder name	
ExtraPass3Name	The name of the linked reconcile password object.	Valid password object name	
ExtraPass3Folder	The folder name where the linked reconcile password object is stored.	Folder name	

Parameter	Description	Acceptable Values	Default Values
ExtraPass3Safe	The safe name where the linked reconcile password object is stored.	Safe name	

AS400 (iSeries) Accounts

Parameter	Description	Acceptable Values	Default Values
-----------	-------------	-------------------	----------------

Required Properties

Platform name	The platform name that is relevant for this password, and is specified in the platform.	Platform name	as400
Address	The address of the remote machine where the password will be used.	IP address	
User Name	The name of the user on the remote machine who this password belongs to.	User name	

Optional Properties

AS400 Account Type	<p>The type of the AS/400 (iSeries) account. Specify one of the following:</p> <ul style="list-style-type: none"> ▪ RegularUserProfile – The account type for regular OS users. ▪ ServiceToolUser – The account type for either Dedicated Service Tools (DST) users or System Service Tools (SST) users. <p>This property can be defined at either platform level or account level.</p>	ServiceToolUser/ RegularUserProfile	RegularUserProfile
--------------------	---	--	--------------------

Linked Accounts

ExtraPass1Name	The name of the linked logon password object.	Valid account name	
ExtraPass1Folder	The folder name where the linked logon password object is stored.	Folder name	
ExtraPass1Safe	The safe name where the linked logon password object is stored	Safe name	
ExtraPass3Name	The name of the linked reconcile password object.	Valid account name	

Parameter	Description	Acceptable Values	Default Values
ExtraPass3Folder	The folder name where the linked reconcile password object is stored.	Folder name	
ExtraPass3Safe	The safe name where the linked reconcile password object is stored.	Safe name	

ESX/i Accounts

Parameter	Description	Acceptable Values
-----------	-------------	-------------------

Required Properties

Platform name	The platform name that is relevant for this account, and is specified in the platform. The default platform name for ESX/i accounts is VMWareESX-API .	
Address	The address of the remote machine where the password will be used.	
User Name	The name of the user on the remote machine who this password belongs to. Specify a local ESX/ESX/i account or 'root'.	

Linked Accounts

ExtraPass1Name	The name of the linked logon password object.	Valid password object name
ExtraPass1Folder	The folder name where the linked logon password object is stored.	Folder name
ExtraPass1Safe	The safe name where the linked logon password object is stored	Safe name
ExtraPass3Name	The name of the linked reconcile password object.	Valid password object name
ExtraPass3Folder	The folder name where the linked reconcile password object is stored.	Folder name
ExtraPass3Safe	The safe name where the linked reconcile password object is stored.	Safe name

Unix SSH Accounts

Parameter	Description	Acceptable Values	Default Values
Required Properties			
Platform name	The platform name that is relevant for this password, and is specified in the platform.	Platform name	Unix SSH
Address	The address of the remote machine where the password will be used.	IP address	
User Name	The name of the user on the remote machine who this password belongs to.	User name	
Optional Properties			
PromptsFilename	A file containing custom prompts, if the environment isn't standard. An expected value is a file name with no path, which corresponds to an actual file in the Central Policy Manager's Bin directory.	File name	
Linked Accounts			
ExtraPass1Name	The name of the linked logon password object.	Valid password object name	
ExtraPass1Folder	The folder name where the linked logon password object is stored.	Folder name	
ExtraPass1Safe	The safe name where the linked logon password object is stored	Safe name	
ExtraPass3Name	The name of the linked reconcile password object.	Valid password object name	
ExtraPass3Folder	The folder name where the linked reconcile password object is stored.	Folder name	
ExtraPass3Safe	The safe name where the linked reconcile password object is stored.	Safe name	

Unix Telnet Accounts

Parameter	Description	Acceptable Values	Default Values
Required Properties			
Platform name	The platform name that is relevant for this password, and is specified in the platform.	Platform name	UnixTelnet
Address	The address of the remote machine where the password will be used.	IP address	
User Name	The name of the user on the remote machine who this password belongs to.	User name	
Optional Properties			
PromptsFilename	A file containing custom prompts, if the environment isn't standard. An expected value is a file name with no path, which corresponds to an actual file in the Central Policy Manager's Bin directory.	File name	

OS390 (Z/OS) SSH Accounts

Parameter	Description	Acceptable Values	Default Values
Required Properties			
Platform name	The platform name that is relevant for this password, and is specified in the platform.	Platform name	OS390SSH
Address	The address of the remote machine where the password will be used.	IP address	
User Name	The name of the user on the remote machine who this password belongs to.	User name	
Optional Properties			
PromptsFilename	A file containing custom prompts, if the environment isn't standard. An expected value is a file name with no path, which corresponds to an actual file in the Central Policy Manager's Bin directory.	name	
Linked Accounts			
ExtraPass1Name	The name of the linked logon password object.	Valid password object name	
ExtraPass1Safe	The safe name where the linked logon password object is stored	Safe name	
ExtraPass1Folder	The folder name where the linked logon password object is stored.	Folder name	

OS390 (Z/OS) Telnet Accounts

Parameter	Description	Acceptable Values	Default Values
Required Properties			
Platform name	The platform name that is relevant for this password, and is specified in the platform.	Platform name	OS390Telnet
Address	The address of the remote machine where the password will be used.	IP address	
User Name	The name of the user on the remote machine who this password belongs to.	User name	
Optional Properties			
Linked Accounts			
ExtraPass1Name	The name of the linked logon password object.	Valid password object name	
ExtraPass1Safe	The safe name where the linked logon password object is stored	Safe name	
ExtraPass1Folder	The folder name where the linked logon password object is stored.	Folder name	

OS390 (Z/OS) FTP Accounts

Parameter	Description	Acceptable Values	Default Values
Required Properties			
Platform name	The platform name that is relevant for this password, and is specified in the platform.	Platform name	OS390FTP
Address	The address of the remote machine where the password will be used.	IP address	
User Name	The name of the user on the remote machine who this password belongs to.	User name	

Database Accounts

DB2 Windows Accounts

Parameter	Description	Acceptable Values	Default Values
Required Properties			
Platform name	The platform name that is relevant for this password, and is specified in the platform.	Platform name	DB2Windows
Address	The address of the remote machine where the password will be used.	IP address	
User Name	The name of the user on the remote machine who this password belongs to.	User name	

Informix Windows Accounts

Parameter	Description	Acceptable Values	Default Values
Required Properties			
Platform name	The platform name that is relevant for this password, and is specified in the platform.	Platform name	InformixWindows
Address	The address of the remote machine where the password will be used.	IP address	
User Name	The name of the user on the remote machine who this password belongs to.	User name	

DB2 Unix SSH Accounts

Parameter	Description	Acceptable Values	Default Values
Required Properties			
Platform name	The platform name that is relevant for this password, and is specified in the platform.	Platform name	DB2 on Unix SSH
Address	The address of the remote machine where the password will be used.	IP address	
User Name	The name of the user on the remote machine who this password belongs to.	User name	

Parameter	Description	Acceptable Values	Default Values
Optional Properties			
PromptsFilename	A file containing custom prompts, if the environment isn't standard. An expected value is a file name with no path, which corresponds to an actual file in the Central Policy Manager's Bin directory.	File name	
Linked Accounts			
ExtraPass1Name	The name of the linked logon password object.	Valid password object name	
ExtraPass1Safe	The safe name where the linked logon password object is stored	Safe name	
ExtraPass1Folder	The folder name where the linked logon password object is stored.	Folder name	

Informix Unix SSH Accounts

Parameter	Description	Acceptable Values	Default Values
Required Properties			
Platform name	The platform name that is relevant for this password, and is specified in the platform.	Platform name	Informix on Unix SSH
Address	The address of the remote machine where the password will be used.	IP address	
User Name	The name of the user on the remote machine who this password belongs to.	User name	
Optional Properties			
PromptsFilename	A file containing custom prompts, if the environment isn't standard. An expected value is a file name with no path, which corresponds to an actual file in the Central Policy Manager's Bin directory.	File name	
Linked Accounts			
ExtraPass1Name	The name of the linked logon password object.	Valid password object name	
ExtraPass1Safe	The safe name where the linked logon password object is stored	Safe name	
ExtraPass1Folder	The folder name where the linked logon password object is stored.	Folder name	

DB2 Unix Telnet Accounts

Parameter	Description	Acceptable Values	Default Values
Required Properties			
Platform name	The platform name that is relevant for this password, and is specified in the platform.	Platform name	DB2UnixTelnet
Address	The address of the remote machine where the password will be used.	IP address	
User Name	The name of the user on the remote machine who this password belongs to.	User name	
Linked Accounts			
ExtraPass1Name	The name of the linked logon password object.	Valid password object name	
ExtraPass1Safe	The safe name where the linked logon password object is stored	Safe name	
ExtraPass1Folder	The folder name where the linked logon password object is stored.	Folder name	

Informix Unix Telnet Accounts

Parameter	Description	Acceptable Values	Default Values
Required Properties			
Platform name	The platform name that is relevant for this password, and is specified in the platform.	Platform name	InformixUnixTelnet
Address	The address of the remote machine where the password will be used.	IP address	
User Name	The name of the user on the remote machine who this password belongs to.	User name	
Linked Accounts			
ExtraPass1Name	The name of the linked logon password object.	Valid password object name	
ExtraPass1Safe	The safe name where the linked logon password object is stored.	Safe name	
ExtraPass1Folder	The folder name where the linked logon password object is stored.	Folder name	

MSSql Accounts

Parameter	Description	Acceptable Values	Default Values
Required Properties			
Platform name	The platform name that is relevant for this password, and is specified in the platform.	Platform name	MSSql
User Name	The name of the user on the remote machine who this password belongs to.	User name	
Optional Properties			
DSN	The name of the DSN connection that will be used. Use either this parameter or 'ConnectionStringFile'.		
Address	The address of the remote machine where the password will be used.	IP address	
Port	The database server port number.	Port	
Database	The name of the database where the account will be used.	String	
Linked Accounts			
ExtraPass1Name	The name of the linked logon password object.	Valid password object name	
ExtraPass1Safe	The safe name where the linked logon password object is stored	Safe name	
ExtraPass1Folder	The folder name where the linked logon password object is stored.	Folder name	
Windows reconcile account	Whether the reconcile account is a Microsoft Windows account or an SQL account.	Yes/No	No

Oracle Accounts

Parameter	Description	Acceptable Values	Default Values
Required Properties			
Platform name	The platform name that is relevant for this password, and is specified in the platform.	Platform name	Oracle
User Name	The name of the user on the remote machine who this password belongs to.	User name	
Optional Properties			
DSN	The name of the DSN connection that will be used. Use either this parameter or 'ConnectionStringFile'.		
Address	The address of the remote machine where the password will be used.	IP address	
Port	The database server port number.	Port	
Database	The name of the database where the account will be used.	String	
Linked Accounts			
ExtraPass1Name	The name of the linked logon password object.	Valid password object name	
ExtraPass1Safe	The safe name where the linked logon password object is stored	Safe name	
ExtraPass1Folder	The folder name where the linked logon password object is stored.	Folder name	

Sybase Accounts

Parameter	Description	Acceptable Values	Default Values
Required Properties			
Platform name	The platform name that is relevant for this password, and is specified in the platform.	Platform name	Sybase
User Name	The name of the user on the remote machine who this password belongs to.	User name	
Optional Properties			
DSN	The name of the DSN connection that will be used. Use either this parameter or 'ConnectionStringFile'.		
Address	The address of the remote machine where the password will be used.	IP address	
Port	The database server port number.	Port	
Database	The name of the database where the account will be used.	String	
Linked Accounts			
ExtraPass1Name	The name of the linked logon password object.	Valid password object name	
ExtraPass1Safe	The safe name where the linked logon password object is stored.	Safe name	
ExtraPass1Folder	The folder name where the linked logon password object is stored.	Folder name	

Security Appliances Accounts

CheckPoint Firewall-1 Accounts

Parameter	Description	Acceptable Values	Default Values
Required Properties			
Platform name	The platform name that is relevant for this password, and is specified in the platform.	Platform name	Firewall1
Address	The address of the remote machine where the password will be used.	IP address	
User Name	The name of the user on the remote machine who this password belongs to.	User name	
ClientDN	The distinguished name of the client entity.	Distinguished name	
ServerDN	The distinguished name of the SmartCenter module.	Distinguished name	
Optional Properties			
SicCertFile	The path and name of the sic certification file. Default: opsec.p12 which should be placed in the Password Manager Bin directory.	Pathname	
Port	The port number to use to connect to the SmartCenter module.	Port	18190

NetScreen SSH Accounts

Parameter	Description	Acceptable Values	Default Values
Required Properties			
Platform name	The platform name that is relevant for this password, and is specified in the platform.	Platform name	NetScreenSSH
Address	The address of the remote machine where the password will be used.	IP address	
User Name	The name of the user on the remote machine who this password belongs to.	User name	

NetScreen Telnet Accounts

Parameter	Description	Acceptable Values	Default Values
Required Properties			
Platform name	The platform name that is relevant for this password, and is specified in the platform.	Platform name	NetScreenTelnet
Address	The address of the remote machine where the password will be used.	IP address	
User Name	The name of the user on the remote machine who this password belongs to.	User name	

Cisco PIX SSH Accounts

Parameter	Description	Acceptable Values	Default Values
Required Properties			
Platform name	The platform name that is relevant for this password, and is specified in the platform.	Platform name	CiscoPixSSH
Address	The address of the remote machine where the password will be used.	IP address	
Type	The type of password to use.	'ciscouser' or 'ciscoenable' or 'ciscoterminal'	
Optional Properties			
User Name	The name of the user on the router that this password belongs to.	'ciscouser' - the name of the user on the PIX machine. 'ciscoenable' -nothing.	
Port	The port that will be used to access the router.	Port	
Linked Accounts			
ExtraPass1Name	The name of the linked logon password object.	Valid password object name	
ExtraPass1Safe	The safe name where the linked logon password object is stored	Safe name	
ExtraPass1Folder	The folder name where the linked logon password object is stored.	Folder name	
ExtraPass2Name	The name of the linked enable password object.	Valid password object name	

Parameter	Description	Acceptable Values	Default Values
ExtraPass2Safe	The safe name where the linked enable password object is stored	Safe name	
ExtraPass2Folder	The folder name where the linked enable password object is stored.	Folder name	

Cisco PIX Telnet Accounts

Parameter	Description	Acceptable Values	Default Values
-----------	-------------	-------------------	----------------

Required Properties

Platform name	The platform name that is relevant for this password, and is specified in the platform.	Platform name	CiscoPixTelnet
Address	The address of the remote machine where the password will be used.	IP address	
Type	The type of password to use.	'ciscouser' or 'ciscoenable' or 'ciscoterminal'	

Optional Properties

User Name	The name of the user on the router that this password belongs to.	'ciscouser' - the name of the user on the PIX machine. 'ciscoenable' – nothing.	
Port	The port that will be used to access the router.	Port	

Linked Accounts

ExtraPass1Name	The name of the linked logon password object.	Valid password object name	
ExtraPass1Safe	The safe name where the linked logon password object is stored	Safe name	
ExtraPass1Folder	The folder name where the linked logon password object is stored.	Folder name	
ExtraPass2Name	The name of the linked enable password object.	Valid password object name	
ExtraPass2Safe	The safe name where the linked enable password object is stored	Safe name	
ExtraPass2Folder	The folder name where the linked enable password object is stored.	Folder name	

Network Device Accounts

Cisco SSH Accounts

Parameter	Description	Acceptable Values	Default Values
Required Properties			
Platform name	The platform name that is relevant for this password, and is specified in the platform.	Platform name	CiscoSSH
Type	The type of password to use.	'ciscouser' or 'ciscoenable' or 'ciscoterminal'	
Optional Properties			
User Name	The name of the user on the router that this password belongs to.	'ciscouser' - the name of the user on the PIX machine. 'ciscoenable' –nothing.	
Address	The address of the remote machine where the password will be used.	IP address	
Port	The port that will be used to access the router.	Port	
vt	The virtual terminal line that will connect to the router.	Number	
Linked Accounts			
ExtraPass1Name	The name of the linked logon password object.	Valid password object name	
ExtraPass1Safe	The safe name where the linked logon password object is stored	Safe name	
ExtraPass1Folder	The folder name where the linked logon password object is stored.	Folder name	
ExtraPass2Name	The name of the linked enable password object.	Valid password object name	
ExtraPass2Safe	The safe name where the linked enable password object is stored	Safe name	
ExtraPass2Folder	The folder name where the linked enable password object is stored.	Folder name	

Parameter	Description	Acceptable Values	Default Values
ExtraPass3Name	The name of the linked reconcile password object.	Valid password object name	
ExtraPass3Safe	The safe name where the linked reconcile password object is stored	Safe name	
ExtraPass3Folder	The folder name where the linked reconcile password object is stored.	Folder name	

Cisco Telnet Accounts

Parameter	Description	Acceptable Values	Default Values
Required Properties			
Platform name	The platform name that is relevant for this password, and is specified in the platform.	Platform name	CiscoTelnet
Type	The type of password to use.	'ciscouser' or 'ciscoenable' or 'ciscoterminal'	
Optional Properties			
User Name	The name of the user on the router that this password belongs to.	'ciscouser' - the name of the user on the PIX machine. 'ciscoenable' -nothing.	
Address	The address of the remote machine where the password will be used.	IP address	
Port	The port that will be used to access the router.	Port	
vt	The virtual terminal line that will connect to the router.	String	
Linked Accounts			
ExtraPass1Name	The name of the linked logon password object.	Valid password object name	
ExtraPass1Safe	The safe name where the linked logon password object is stored	Safe name	
ExtraPass1Folder	The folder name where the linked logon password object is stored.	Folder name	
ExtraPass2Name	The name of the linked enable password object.	Valid password object name	

Parameter	Description	Acceptable Values	Default Values
ExtraPass2Safe	The safe name where the linked enable password object is stored	Safe name	
ExtraPass2Folder	The folder name where the linked enable password object is stored.	Folder name	
ExtraPass3Name	The name of the linked reconcile password object.	Valid password object name	
ExtraPass3Safe	The safe name where the linked reconcile password object is stored	Safe name	
ExtraPass3Folder	The folder name where the linked reconcile password object is stored.	Folder name	

Directory Accounts

SunOne Directory Accounts

Parameter	Description	Acceptable Values	Default Values
Required Properties			
Platform name	The platform name that is relevant for this password, and is specified in the platform.	Platform name	SunOneDirectory
Address	The address of the remote machine where the password will be used.	IP address	
Optional Properties			
Port	The port that will be used to access the remote directory.	Port	
UserDN	The distinguished name of the user.	Distinguished name	
Linked Accounts			
ExtraPass1Name	The name of the linked logon password object.	Valid password object name	
ExtraPass1Safe	The safe name where the linked logon password object is stored	Safe name	
ExtraPass1Folder	The folder name where the linked logon password object is stored.	Folder name	

SunOne Directory SSL Accounts

Parameter	Description	Acceptable Values	Default Values
Required Properties			
Platform name	The platform name that is relevant for this password, and is specified in the platform.	Platform name	SunOneDirectorySSL
Address	The address of the remote machine where the password will be used.	IP address	
Optional Properties			
Port	The port that will be used to access the remote directory.	Port	
UserDN	The distinguished name of the user.	Distinguished name	
Linked Accounts			
ExtraPass1Name	The name of the linked logon password object.	Valid password object name	
ExtraPass1Safe	The safe name where the linked logon password object is stored	Safe name	
ExtraPass1Folder	The folder name where the linked logon password object is stored.	Folder name	

Novell eDirectory Accounts

Parameter	Description	Acceptable Values	Default Values
Required Properties			
Platform name	The platform name that is relevant for this password, and is specified in the platform.	Platform name	Novell-eDirectory
Address	The address of the remote machine where the password will be used.	IP address	
Optional Properties			
Port	The port that will be used to access the remote directory.	Port	
UserDN	The distinguished name of the user.	Distinguished name	

Parameter	Description	Acceptable Values	Default Values
Linked Accounts			
ExtraPass1Name	The name of the linked logon password object.	Valid account name	
ExtraPass1Safe	The safe name where the linked logon password object is stored	Safe name	
ExtraPass1Folder	The folder name where the linked logon password object is stored.	Folder name	

Service Accounts

Windows Services Accounts

Parameter	Description	Acceptable Values	Default Values
Required Properties			
Platform name	The platform name that is relevant for this password, and is specified in the platform.	Platform name	WinService
ServiceName	The name of the Service that will use the password.	Service name	
Address	The address of the Windows Service machine.	IP/DNS address, Windows domain or machine name, TNS name.	
Optional Properties			
RestartService	This Boolean value indicates whether or not to restart the Service after updating the password in the Service successful.	Yes/No	No

Windows Scheduled Tasks Accounts

Parameter	Description	Acceptable Values	Default Values
Required Properties			
Platform name	The platform name that is relevant for this password, and is specified in the platform.	Platform name	SchedTask
TaskName	The name of the Task that will use the password.	Task name	
Address	The address of the Windows task machine.	IP/DNS address, Windows domain or machine name, TNS name.	

IIS Application Pool Accounts

Parameter	Description	Acceptable Values	Default Values
Required Properties			
Platform name	The platform name that is relevant for this password, and is specified in the platform.	Platform name	IISAppPool
AppPoolName	The name of the IIS Application Pool that will use the password.	IIS Application Pool name	
Address	The address of the IIS Application Pool machine.	IP/DNS address, Windows domain or machine name, TNS name.	
Optional Properties			
RestartService	This Boolean value indicates whether or not to restart the Service after updating the password in the IIS Application Pool successfully.	Yes/No	No

Registry Accounts

Parameter	Description	Acceptable Values	Default Values
Required Properties			
Platform name	The platform name that is relevant for this password, and is specified in the platform.	Platform name	Registry
RegistryPathName	The name of the registry path to update.	Registry path name	
RegistryValueName	The value of the registry path to update.	Registry value	
Address	The address of the registry.	IP/DNS address, Windows domain or machine name, TNS name.	
Optional Properties			
Prefix	A prefix for the registry value.	String	
Postfix	A postfix for the registry value.	String	

COM Plus Accounts

Parameter	Description	Acceptable Values	Default Values
Required Properties			
Platform name	The platform name that is relevant for this password, and is specified in the platform.	Platform name	ComPlus
ApplicationName	The name of the COM Plus application that will use the password.	COM Plus application name	
Address	The address of the COM Plus application machine.	IP/DNS address.	
Optional Properties			
Restart	This Boolean value indicates whether or not to restart the COM Plus application after updating the password in the application successfully.	Yes/No	No
Linked Accounts			
LogonAccount	The name and property index of the account used to log onto the COM Plus application.	Valid account name and index	

IIS Anonymous Accounts

Parameter	Description	Acceptable Values	Default Values
Required Properties			
Platform name	The platform name that is relevant for this password, and is specified in the platform.	Platform name	IISAnonymous
WebSiteName	The name of the web site or virtual directory where the IIS Anonymous account will be used.	URL	
Address	The address of the web site or virtual directory where the IIS Anonymous account will be used.	IP address	
Optional Properties			
WebSiteID	The unique ID of the website where the IIS Anonymous account will be used.	String	
VirtualDirPath	The name of the virtual directory where the IIS Anonymous account will be used.	String	

Parameter	Description	Acceptable Values	Default Values
-----------	-------------	-------------------	----------------

Linked Accounts

LogonAccount	The name and property index of the account used to log onto the COM Plus application.	Valid account name and index	
--------------	---	------------------------------	--

Accounts Stored in Text Configuration Files

Parameter	Description	Acceptable Values	Default Values
-----------	-------------	-------------------	----------------

Required Properties

Address	The address of the remote machine where the Text configuration file is located.	IP address	
File Path	The full file path including name and extension of the configuration file that contains the password. <ul style="list-style-type: none"> ▪ For example, in Unix, "/home/passwords/filename". ▪ For example in Windows, "C:\NewShare\passwords\FileName.txt". 	Full path	
Password Regex	The password regular expression that matches the line of the password in the configuration file. Specify a prefix and a postfix to allow the CPM to identify the entire line, and specify the password in parenthesis "(" and ")". This single-groups the password and defines the location of the password to change. Example 1: Specify Password=(.*) ; to replace the new password with all the characters between Password= and ; (semi-colon). Example 2: Specify Password=(.....) ; to replace the new password with eight characters after Password= . If the ; (semi-colon) is not found after eight characters, this regex will not match the line in the password file and will not replace the password successfully. The CPM will replace all occurrences of the specified regex in the password file. If the password regex is empty, the CPM will return an error message.	String	
ConnectionType	The connection type that will be used to access the target machine where the configuration file resides.	Windows File Sharing/ SSH	

Optional Properties

Port	The port that is used for non-standard SSH ports. This parameter is only relevant for SSH protocol.	Port	22
------	---	------	----

Parameter	Description	Acceptable Values	Default Values
Backup Password File	Whether or not a backup configuration file will be created. The backup file will be named "Backup__%fileName%" where %fileName% will be the suffix after the last trailing '/' or '\' and will reside in the same directory as the original password file.	Yes/No	

Accounts Stored in INI Configuration Files

Parameter	Description	Acceptable Values	Default Values
Required Properties			
Address	The address of the remote machine where the INI configuration file is saved.	IP address	
File Path	The full file path including name and extension of the configuration file that contains the password. <ul style="list-style-type: none"> ▪ For example, in Unix, "/home/passwords/filename". ▪ For example in Windows, "C:\NewShare\passwords\FileName.ini". 	Full path	
INI Section	The ini section that contains the password string. If more than one ini sections contain a password to manage, create multiple service accounts for this account or use the TextConfigFile usage.	String	
INI Parameter Name	The name of the parameter in the configuration file that contains the password. If more than one parameter in the same ini section contains a password to manage, create multiple service accounts for this account or use the TextConfigFile usage.	String	
ConnectionType	The connection type that will be used to access the target machine where the configuration file resides.	Windows File Sharing/SSH	
Optional Properties			
Port	The port that is used for non-standard SSH ports. This parameter is only relevant for SSH protocol.	Port	22
Backup Password File	Whether or not a backup configuration file will be created. The backup file will be named "Backup__%fileName%" where %fileName% will be the suffix after the last trailing '/' or '\' and will reside in the same directory as the original password file.	Yes/No	

Accounts Stored in XML Configuration Files

Parameter	Description	Acceptable Values	Default Values
Required Properties			
Address	The address of the remote machine where the XML configuration file is located.	IP address	
File Path	The full file path including name and extension of the configuration file that contains the password. <ul style="list-style-type: none"> For example, in Unix, "/home/passwords/fileName". For example in Windows, "C:\NewShare\passwords\FileName.xml". 	Full path	
XML Element	The XPath that represents the xml element which contains the text/attribute to change. For more information about XPath, refer to http://www.w3schools.com/XPath/xpath_syntax.asp .	String	
Connection Type	The connection type that will be used to access the target machine where the configuration file resides. Valid values are: <ul style="list-style-type: none"> Windows File Sharing SSH 		
Optional Properties			
Port	The port that is used for non-standard SSH ports. This parameter is only relevant for SSH protocol.	Port	22
Backup Password File	Whether or not a backup configuration file will be created. The backup file will be named "Backup__%fileName%" where %fileName% will be the suffix after the last trailing '/' or '\' and will reside in the same directory as the original password file.	Yes/No	
Password Regex	The password regex that matches the line of the password in the configuration file. Specify a prefix and a postfix to allow the CPM to identify the entire line, and specify the password in parenthesis "(" and ")". This single-groups the password and defines the location of the password to change. Example 1: Specify Password=(.*) ; to replace the new password with all the characters between Password= and ; (semi-colon). Example 2: Specify Password=(.....) ; to replace the new password with eight characters after Password= . If the ; (semi-colon) is not found after eight characters, this regex will not match the line in the password file and will not replace the password successfully. The CPM will replace all occurrences of the specified regex in the password file. If the password regex is empty, the CPM will return an error message.		

Parameter	Description	Acceptable Values	Default Values
XML Attribute	The xml attribute of the xml element (defined in the XML Element parameter) to change. If this parameter is empty, the new password value will be written in the text of the xml element.		

Accounts Stored in Web Configuration Files

Parameter	Description	Acceptable Values	Default Values
Required Properties			
Address	The address of the remote machine where the WebConfig configuration file is located.	IP address	
File Path	The full file path including name and extension of the file that contains the password. <ul style="list-style-type: none"> ▪ For example, in Unix, "/home/passwords/filename". ▪ For example in Windows, "C:\NewShare\passwords\FileName.xml". 	Full path	
XML Element	The XPath that represents the xml element which contains the text/attribute to change.	String	
ConnectionType	The connection type that will be used to access the target machine where the configuration file resides.	Windows File Sharing/ SSH	
Optional Properties			
Port	The port that is used for non-standard SSH ports. This parameter is only relevant for SSH protocol.	Port	22
Backup Password File	Whether or not a backup configuration file will be created. The backup file will be named "Backup__%fileName%" where %fileName% will be the suffix after the last trailing '/' or '\' and will reside in the same directory as the original password file.	Yes/No	

Parameter	Description	Acceptable Values	Default Values
Password Regex	<p>The password regex that matches the line of the password in the configuration file. Specify a prefix and a postfix to allow the CPM to identify the entire line, and specify the password in parenthesis "(" and ")". This single-groups the password and defines the location of the password to change.</p> <p>Example 1: Specify Password=(.*); to replace the new password with all the characters between Password= and ; (semi-colon).</p> <p>Example 2: Specify Password=(.....); to replace the new password with eight characters after Password=. If the ; (semi-colon) is not found after eight characters, this regex will not match the line in the password file and will not replace the password successfully.</p> <p>The CPM will replace all occurrences of the specified regex in the password file.</p> <p>If the password regex is empty, the CPM will return an error message.</p>		
XML Attribute	<p>The xml attribute of the xml element (defined in the XML Element parameter) to change. If this parameter is empty, the new password value will be written in the text of the xml element.</p>		

Accounts Stored in Databases

Parameter	Description	Acceptable Values	Default Values
Required Property			
UsageDisplayName	The display name for the service account defined by the user	String	
Optional Properties			
DSN (ODBC)	The name of the DSN connection that will be used.		
Address	The address of the remote machine.		
Port	The port used to access the remote machine.		
Database	The name of the database that contains the passwords to manage.		
TableName	The name of the table on the remote database that contains the passwords to manage.		
ColumnName	The name of the column that contains passwords in the table in the remote database.		

Parameter	Description	Acceptable Values	Default Values
UniqueIDColumnName	The unique primary key column name that will be used to identify specific records containing passwords for the account.		
UniqueIDColumnValue	The unique primary key column value that will be used to identify specific records containing passwords for the account.		

Appendix B: Creating User Credential Files

Some Vault components can access the Vault server with a user credential file that contains the user's name and encrypted authentication details, preventing the need for interactive authentication and enabling file sharing and transfer processes to be performed automatically.

Before creating the user credential file, make sure that you are familiar with the user's authentication details in the Vault.

CreateCredFile Utility

The Vault interfaces access the Vault with a user credential file that contains the user's Vault username and encrypted logon information. This user credential file can be created for password, Token, PKI, or Radius authentication with a utility that is run from a command line prompt. It can also create a credentials file for authentication through a Proxy server.

User credential files can specify restrictions which increase their security level and ensure that they cannot be used by anyone who is not permitted to do so, nor from an unauthorized location. The updated CreateCredFile utility can enforce any of the following restrictions:

- **Specific application** – The credentials file can only be used by a specific CyberArk application or module. This can be specified for Password, Token, or PKI authentication but not for Proxy authentication. For more details about specific applications, refer to *Specifying Applications*.
- **Specific path** – The credentials file can only be used by an executable located in a certain path.
- **IP address or hostname** – The credentials file can only be used on the machine where it is created.
- **Operating System user** – The credentials file can only be used by an application started by a specified Operating System user.

These restrictions are specified during the credentials file creation process.

Credential files that were created in versions prior to version 4.5 with the CreateAuthFile and CreateCredFile utilities can still be used. However, they do not contain the increased security restrictions that are included in the CreateCredFile utility that is released with this version.

Credentials files that are created with restrictions will not be supported by CyberArk components from previous versions.

Before creating or updating the user credential file, make sure that you are familiar with the user's authentication details in the Vault as you will be required to provide logon credentials to generate the encrypted credentials file.

Credential File Security

Credential files are protected using the following mechanisms:

1. The encrypted token (320-bit) is changed on a daily basis. This means that a credential file that was used today will not be usable tomorrow.
2. The encrypted token is encrypted using AES 256-bit key that comprises the following parts:
 - i. Random salt that is stored in the credential file (160-bit). This randomness assures that each credential file is encrypted with a unique key.
 - ii. Environmental key material:
 - Client id – Ten characters that identify a specific component
 - OS user – The ID of the OS user who runs the component
 - IP address of the local machine
 - Application – The specific application or module that will use the credentials file.
 - iii. The key is generated by a secure hash (SHA1) of the above key materials.
3. You can protect your credential files even more using the appropriate operating system permissions.

Specifying Applications

The following CyberArk applications can be specified in a user credentials file:

Application	ID
Central Policy Manager	CPM
Password Vault Web Access	PVWA
Password Vault Web Access application user	PVWAAApp
OPM and Credential Provider	AppPrv
Privileged Session Manager	PSM
Privileged Session Manager application user	PSMAApp
CyberArk Replicator/Restore/Prebackup	CABACKUP
Disaster Recovery Vault	DR
Event Notification Engine	ENE
PrivateArk Client	WINCLIENT, GUI
CyberArk CLI	PACLI
CyberArk ActiveX API	XAPI
CyberArk .Net API	NAPI
Export Vault Data	EVD
CyberArk Encryption Utility	CACrypt

Creating User Credentials Files

The CreateCredFile utility is located in the CyberArk\Utilities installation folder. It can be used to create a user credential file for password, RADIUS, Token, or PKI authentication with a utility that is run from a command line prompt.

It can also create a user credential file for authentication through a Proxy server.

The CreateCredFile utility uses the following syntax:

```
CreateCredFile <FileName> <command> [command parameters]
```

Parameter	Unix Command	Specifies
Filename	Filename	The name of the user credential file to create or update, specifically user.cred .
Password	Password	Indicates that the credential file will be created with password authentication details.
/Username	-username	Sets the username in the credential file. This parameter is required. If you do not specify it in the command, you will be prompted for it.
/Password	-password	The password that will be encrypted in the credential file. This parameter is required. If you do not specify it in the command, you will be prompted for it.
/UseOSProtected Storage		Use Operating System protected storage for credential file secret (Windows only). Valid values are Machine , User and No . By default, this parameter is set to No .
User		Use protected storage that is accessible only to the user who is logged on and invoked the CreateCredFile utility.
Machine		Use protected storage that is accessible only for the machine where the CreateCredFile utility was invoked.
None		Do not use operating system protected storage.
/DisableSyncPassword ToDR	-DisableSync PasswordToDR	Whether or not passwords in user credential files will be replicated to all DR sites before they are replaced. By default, this parameter is set to 'No', which makes sure that user credential files on all DR sites (if they exist) are synchronized with the Production Vault and that users will be able to continue working with the Vault seamlessly after a failover. If this parameter is changed to 'Yes', passwords will be replaced in credential files regardless of whether or not they have been replicated to all DR sites.
/ExternalAuth	-externalauth	The type of external authentication that will be used to authenticate users to the Vault.
Radius	-radius	Creates a user name-password credential file for use with RADIUS server.

Parameter	Unix Command	Specifies
LDAP	-ldap	Creates a user name-password credential file for use with an LDAP directory.
No	-no	This credential file will not be used with either a Radius server or an LDAP directory.
/AppType <Application ID>	-apptype <application id>	A unique application ID that specifies the application that will be able use this file.
/ExePath <Path>	-exepath <path>	The full path of the executable that will be able to use this file. Notes: <ul style="list-style-type: none"> On UNIX machines, if the executable will be executed from the PATH you can specify only the name of the executable. Otherwise, specify the complete path. When you specify PVWA, specify the full path of the web server executable, e.g. c:\windows\system32\inetsrv\w3wp.exe.
/IpAddress	-ipaddress	The IP address of the current machine. When this parameter is specified, the credentials file will specify the IP address of the current machine and will only authenticate the user to the Vault from the current machine. Note: Specify either the 'IpAddress' parameter or the 'ClientHostName' parameter. You cannot specify both.
/ClientHostname	-/clienthostname	The hostname of the current machine. When this parameter is specified, the credentials file will specify the hostname of the current machine and will only authenticate the user to the Vault from a machine with the specified hostname. Note: Specify either the 'IpAddress' parameter or the 'ClientHostName' parameter. You cannot specify both.
/OSUsername <Operating System User name>	-osusername <operating system user name>	The name of the Operating System user who will be able to use this file. Notes: <ul style="list-style-type: none"> On UNIX machines, specify only the username. On Windows machines, specify the username in "domain_name\username" format. When the application is executed as a Windows service that uses local system permissions, specify "nt authority\system". The quotation marks are required because of the space in "nt authority".
/DisplayRestrictions	- displayrestrictions	When this parameter is specified, the generated credentials file will specify all the restrictions in a readable manner. This will enable users to understand the exact restrictions on the file.

Parameter	Unix Command	Specifies
Token		Creates a user credential file with a key stored on a token.
/Username		Sets the username in the credential file. This parameter is required. If you do not specify it in the command, you will be prompted for it.
/Password		The password that will be encrypted in the credential file. This parameter is required. If you do not specify it in the command, you will be prompted for it.
/DLLpath		Specifies the DLL file path used by the token device. This parameter is required. If you do not specify it in the command, you will be prompted for it.
/PIN		Specifies the PIN code required by the token device. This parameter is required. If you do not specify it in the command, you will be prompted for it.
/ExternalAuth		The type of external authentication that will be used to authenticate users to the Vault.
Radius		Creates a credential file for use with RADIUS server.
LDAP		Creates a credential file for use with an LDAP directory.
No		This credential file will not be used with either a Radius server or an LDAP directory.
/InitToken		Initializes the token device for use with CyberArk password authentication. This parameter must be specified the first time you use a token device to store a CyberArk password encryption key.
/AppType <Application ID>		A unique application ID that specifies the application that will be able use this file.
/ExePath <Path>		The full path of the executable that will be able to use this file. Notes: <ul style="list-style-type: none"> ▪ On UNIX machines, if the executable will be executed from the PATH you can specify only the name of the executable. Otherwise, specify the complete path. ▪ When you specify PVWA, specify the full path of the web server executable.

Parameter	Unix Command	Specifies
/IpAddress		The IP address of the current machine. When this parameter is specified, the credentials file will specify the IP address of the current machine and will only authenticate the user to the Vault from the current machine. Note: Specify either the 'IpAddress' parameter or the 'ClientHostName' parameter. You cannot specify both.
/ClientHostname		The hostname of the current machine. When this parameter is specified, the credentials file will specify the hostname of the current machine and will only authenticate the user to the Vault from a machine with the specified hostname. Note: Specify either the 'IpAddress' parameter or the 'ClientHostName' parameter. You cannot specify both.
/OSUsername <Operating System User name>		The name of the Operating System user who will be able to use this file. Notes: <ul style="list-style-type: none"> ▪ On UNIX machines, specify only the username. ▪ On Windows machines, specify the username in "domain_name\username" format. ▪ When the application is executed as a Windows service that uses local system permissions, specify "nt authority\system". The quotation marks are required because of the space in "nt authority".
/DisplayRestrictions		When this parameter is specified, the generated credentials file will specify all the restrictions in a readable manner. This will enable users to understand the exact restrictions on the file.
PKI		Creates a credential file based on a PKI certificate.
/CertIssuer		Personal certificate issuer.
/CertSerial		Personal certificate serial number.
/PIN		Specifies the PIN code required to access the certificate. This parameter is required if the certificate is stored on a Token.
/AppType <Application ID>		A unique application ID that specifies the application that will be able use this file.

Parameter	Unix Command	Specifies
/ExePath <Path>		<p>The full path of the executable that will be able to use this file.</p> <p>Notes:</p> <ul style="list-style-type: none"> On UNIX machines, if the executable will be executed from the PATH you can specify only the name of the executable. Otherwise, specify the complete path. When you specify PVWA, specify the full path of the web server executable.
/IpAddress		<p>The IP address of the current machine. When this parameter is specified, the credentials file will specify the IP address of the current machine and will only authenticate the user to the Vault from the current machine.</p> <p>Note: Specify either the 'IpAddress' parameter or the 'ClientHostName' parameter. You cannot specify both.</p>
/ClientHostname		<p>The hostname of the current machine. When this parameter is specified, the credentials file will specify the hostname of the current machine and will only authenticate the user to the Vault from a machine with the specified hostname.</p> <p>Note: Specify either the 'IpAddress' parameter or the 'ClientHostName' parameter. You cannot specify both.</p>
/OSUsername <Operating System User name>		<p>The name of the Operating System user who will be able to use this file.</p> <p>Notes:</p> <ul style="list-style-type: none"> On UNIX machines, specify only the username. On Windows machines, specify the username in "domain_name\username" format. When the application is executed as a Windows service that uses local system permissions, specify "nt authority\system". The quotation marks are required because of the space in "nt authority".
/DisplayRestrictions		<p>When this parameter is specified, the generated credentials file will specify all the restrictions in a readable manner. This will enable users to understand the exact restrictions on the file.</p>
PROXY		<p>Creates a credential file based on PROXY authentication.</p>
/ProxyUser		<p>The name of the Proxy user. This parameter is required. If you do not specify it in the command, you will be prompted for it.</p>

Parameter	Unix Command	Specifies
/ProxyPassword		The password that will be decrypted in the credential file. This parameter is required. If you do not specify it in the command, you will be prompted for it.
/ProxyAuth Domain		The domain name of the Proxy user.
/ExePath <Path>		The full path of the executable that will be able to use this file. Notes: <ul style="list-style-type: none"> On UNIX machines, if the executable will be executed from the PATH you can specify only the name of the executable. Otherwise, specify the complete path. When you specify PVWA, specify the full path of the web server executable.
/IpAddress		The IP address of the current machine. When this parameter is specified, the credentials file will specify the IP address of the current machine and will only authenticate the user to the Vault from the current machine. Note: Specify either the 'IpAddress' parameter or the 'ClientHostName' parameter. You cannot specify both.
/ClientHostname	-/clienthostname	The hostname of the current machine. When this parameter is specified, the credentials file will specify the hostname of the current machine and will only authenticate the user to the Vault from a machine with the specified hostname. Note: Specify either the 'IpAddress' parameter or the 'ClientHostName' parameter. You cannot specify both.
/OSUsername <Operating System User name>		The name of the Operating System user who will be able to use this file. Notes: <ul style="list-style-type: none"> On UNIX machines, specify only the username. On Windows machines, specify the username in "domain_name\username" format. When the application is executed as a Windows service that uses local system permissions, specify "nt authority\system". The quotation marks are required because of the space in "nt authority".
/DisplayRestrictions		When this parameter is specified, the generated credentials file will specify all the restrictions in a readable manner. This will enable users to understand the exact restrictions on the file.
/?		Lists the available options.

The following instructions explain how to create a user credential file. The examples used in these instructions run the utility from the Utilities subfolder, and create a credential file called 'user.cred'.

Note: The text typed by the user appears in bold.

Creating the User Credential File for Password Authentication

1. At the command line prompt, run the **CreateCredFile.exe** utility. You must specify the username and password to the Vault. You can also specify whether or not Radius authentication will be used.

For extended security on Windows systems, store the secret of the credential file in Windows protected storage by using the **/UseOSProtectedStorage** parameter. Use the following guidelines when protecting the secret in the Windows protected storage:

- **When the user who creates the credential file is the only user who will use it:** Store the credential file secret in the user's Windows protected storage by specifying the **/UseOSProtectedStorage User** parameter. This ensures that only the user who created the credential file will be able to access its secret.
- **For CyberArk services or when the user that created the credential file is not the user that will use it:** Store the credential file secret in the machine's Windows protected storage by specifying the **/UseOSProtectedStorage Machine** parameter. This ensures that the credential file secret will only be accessible from the machine where it was created.

```
>createcredfile.exe user.cred Password /username Paul /password Pass
/ExternalAuth radius /UseOSProtectedStorage Machine
```

The above example shows that this credential file will be called 'user.cred', and will contain an encrypted password for the Vault user called 'Paul'. The credential file's secret will be stored in the machine's Windows protected storage. The file can be used to log onto the file with Radius authentication.

If you do not specify the command parameters, username, password, and radius, you are prompted for them now. An example of this appears in the following example:

```
Vault Username [mandatory] ==> Paul
Vault Password (will be encrypted in credential file) ==> *****
Radius server will be used for authentication (yes/no) [y] ==> yes
```

The user's credential file will now be created and saved in the current folder.

```
Command ended successfully
```

Creating the User Credential File using a Token

The Vault supports logon with a password that has been encrypted by a key on a USB token or a Smartcard. This password is stored in the user's credential file, and is decrypted by the external token for logon.

Any PKCS#11 token can be used for this type of authentication, as long as it meets all of the following criteria:

- The token must be a hardware token.
- The token is accessible through the PKCS#11 interface.
- Access to the token is only possible after supplying a PIN.
- The token supports RSA with 1024 or 2048 bit key length.
- The token must be able to perform encryption and key generation in hardware.

These instructions are for creating a user credential file with a new external token.

1. Attach the token to the computer.
 - If you are using a USB token, place the token in the USB port.
 - If you are using a Smartcard, place the card in the Smartcard reader.
2. At the command line prompt, run the **CreateCredFile.exe** utility. You must specify the username and password to the Vault, the full path of the PKCS#11 dll file that will encrypt the password, and the PIN that is required by the token device. You can also specify

```
>CreateCredFile.exe user.cred token /username Paul /password Pass  
/dllpath i:\windows\system32\etpkcs11.dll /pin PinPass
```

The above example shows that this credential file will be called 'user.cred', and will be created with a key that is stored on a token. 'Paul' is the user who will be specified in the credential file, together with his password, asdf. The dll path used by the token device is specified, as well as the PIN that is required to access the token device.

If you have not specified the username, password, dll path and password, you are prompted for it now.

```
Vault Username [mandatory] ==> Paul  
Vault Password (will be encrypted in credential file) ==> ****  
Path of Token dll [mandatory] ==> i:\windows\system32\etpkcs11.dll  
Pin code required by the Token device ==> *****  
Radius server will be used for authentication (yes/no) [optional] ==> no  
Initialize the Token (yes/no) [optional] ==> no
```

3. To initialize the token, type **yes**,

or,

If the token has already been initialized with the CreateCredFile utility, type **no**.

The user credential file is now created and saved in the current folder.

```
Command ended successfully
```

Creating the User Credential File for PKI Authentication

The user can create a user credential file for logon with a PKI certificate. Before creating the credential file, the authentication certificate must be imported into the Microsoft Windows certificate store. For more details, refer to *Importing a Certificate for Authentication*, page 967.

Note: A PIN to access a PKI certificate can only be used in a Windows 2000 environment or higher.

- At the command line prompt, run the **CreateCredFile.exe** utility.

```
CreateCredFile.exe user.cred PKI /certissuer CN=MyCompany_CA
/certserial "1963f68d0000000017c" /Pin PinPass /AppType PACLI
/ExePath "C:\Program Files\PrivateArk\Client\PACLI.exe" /IPAddress
/OSUsername my_dom\Paul /DisplayRestrictions
```

The above example shows that this credential file will be called 'user.cred', and will be created based on a PKI certificate. The certificate issuer for this credential file is MyCompany_CA and the certificate detail serial number is '1963f68d0000000017c'. The PIN required to access this certificate is '12341234'.

If you do not specify the certificate issuer and serial number, the Select Certificate window appears to enable you to select the PKI certificate that will give the user access to the Vault.

Note: If a PIN is required to access the certificate, you must enter the PIN in the command line.



- Select the PKI certificate to use, then click **OK**; the user's credential file will now be created and saved in the current folder.

The following message appears to confirm that the authentication file has been created successfully.

```
Command ended successfully
```

Importing a Certificate for Authentication

Authentication certificates can be used to authenticate to the Vault if the certificate has been imported into the Microsoft Windows certificate store.

The certificate store is divided into several locations to limit accessibility (for security reasons). The most common location for certificates is the “Current User” location. When importing certificates into Microsoft Windows, this is the default location into which the certificates are imported. The certificates in the “Current User” location are only accessible to the user that is currently logged on. One user will not be able to access certificates in another user’s “Current User” location.

Creating the User Credential File for Proxy Authentication

The Proxy user and password can be stored encrypted in a credentials file instead of being specified in the Vault parameter file.

1. At the command line prompt, run the **CreateCredFile.exe** utility.

```
>createcredfile.exe user.cred Proxy /ProxyUser PUser  
/ProxyPassword Pass /ExePath "C:\Program  
Files\PrivateArk\Client\PACLI.exe" /IPAddress /OSUsername  
my_dom\Paul /DisplayRestrictions
```

The above example will create a file called ‘user.cred’ and will enable the proxy user to log onto the Vault with proxy authentication. The credentials file will contain an encrypted proxy password for the proxy user called PUser.

If you do not specify the name and password of the proxy user, you will be prompted for them. An example of this appears in the following example:

```
Proxy Username [mandatory] ==> PUser  
Proxy Password (will be encrypted in credential file) ==> ****  
Domain name of ProxyUser [optional] ==> MyCompany.com
```

The user’s credential file will now be created and saved in the current folder.

```
Command ended successfully
```

CreateAuthFile Utility

The PWWA agent account requires a user credential file in order to be able to log onto the Vault and enable users to access files stored inside.

The user credential file, called `user.ini`, contains the agent account's Vault username and logon information. The location of this credential file is specified in the 'GWUserFile' parameter in the Password Vault Web Access configuration file, `PVConfiguration.xml`.

The user credential file is created with a utility that is included on the Password Vault Web Access installation CD. It is run from a command line prompt, and can specify standard password or token authentication. The authentication used in the credential file must be the same as the authentication method that is specified for the user in the PrivateArk Client.

For more information about creating users and updating their authentications, refer to *Creating and Managing Locations, Users, and Groups*, page 39.

The `CreateAuthFile` utility uses the following syntax:

```
CreateAuthFile <FileName> [/TOKEN]
                [/USERNAME=<username>] [/PASSWORD=<password>]
                [/DLLPATH=<dll path>]
                [/PIN=<pin>]
                /?
```

Parameter	Specifies
Filename	The name of the user credential file to create or update, specifically user.ini . If this utility is run without any more parameters, the file will be created for the default authentication method, which is password authentication.
/Token	Creates a user name/password credential file. The password is encrypted with a key stored on a token.
/Username	Sets the username in the credential file.
/Password	Sets the password in the credential file.
/DLLpath	Specifies the DLL file path used by the token device.
/PIN	Specifies the PIN code required by the token device.
/?	Lists the available options.

The following instructions explain how to create a user credential file. The example used in these instructions runs the utility from the 'Program Files\CyberArkUtilities' folder, and creates a credential file called 'user.ini'.

Note: The text typed by the user appears in bold.

Creating the User Credential File for Password Authentication

1. At the command line prompt, run the **CreateAuthFile.exe** utility. You can specify the username and password to the Vault.

```
C:\Program Files\CyberArk\Utilities> createauthfile.exe user.ini  
/username=Paul
```

If you have not specified the username and password, you are prompted for it now. An example of this appears in the following example, which prompts the user for his password.

```
Enter PrivateArk password: *****
```

The user's credential file will now be created and saved in the current folder.

```
Create Auth File has been successful.
```

Creating the User Credential File using a Token

The Vault supports logon with a password that has been encrypted by a key on a USB token or a Smartcard. This password is stored in the user's credential file, and is decrypted by the external token for logon.

Note: In a Windows 2003 environment, this authentication method is only supported when the application is run using a local system account.

Any PKCS#11 token can be used for this type of authentication, as long as it meets all of the following criteria:

- The token must be a hardware token.
- The token is accessible through the PKCS#11 interface.
- Access to the token is only possible after supplying a PIN.
- The token supports RSA with 1024 or 2048 bit key length.
- The token must be able to perform encryption and key generation in hardware.

These instructions are for creating a user credential file with a new external token.

1. At the command line prompt, run the **CreateAuthFile.exe** utility. You can specify the username and password to the Vault, the full path of the PKCS#11 dll file that will encrypt the password, and the PIN that is required by the token device.

```
C:\Program Files\CyberArk\Utilities> createauthfile.exe user.ini  
/token /username=Paul DLLpath="C:\WINNT\system32\tpkcs11.dll"
```

The following prompt appears.

```
Make sure token is properly inserted.
```

2. At this point, attach the token.
 - If you are using a USB token, place the token in the USB port.
 - If you are using a Smartcard, place the card in the Smartcard reader.

If you have not specified all the parameters that are required to create the authentication file, you will be prompted for them now. An example of this appears in the following example, which prompts the user for his password and then the PIN.

```
Enter PrivateArk password: *****
Enter Token PIN: *****
```

The token information is displayed. Make sure that the information displayed is the correct information for the token you wish to save the encryption key on.

For this example, we used an Aladdin eToken pro token.

```
Token Info:
Label = #3 v4.1.5.3
Model = eToken CardOS/M4
Serial Number = 00030743
Token does not contain an encryption key for the CyberArk
applications.
```

3. If the token does not contain an encryption key for the CyberArk Vault, you can create one now; type **yes** to create the key, then press **Enter** to generate the encryption key.

```
Type "YES" to create the key now: yes
The key is being generated. This may take a few minutes...
```

As soon as the encryption key is generated, the user credential file can be created and saved in the current folder.

```
Create Auth File has been successful.
Token Info:
Label = #3 v4.1.5.3
Model = eToken CardOS/M4
Serial Number = 00030743
```

Confirmation that the user credential file has been created successfully includes the token information.

Appendix C: Configuring Debug Levels

The following tables list the configuration files per component of the Privileged Account Security solution, specify how to set the debug mode, and give the location of the log files for each component.

Vault

Configuration file(s)	<ul style="list-style-type: none"> ▪ DBParm.ini ▪ \Database\my.ini (database configuration file)
To set debug mode	<ul style="list-style-type: none"> ▪ DebugLevel=PE(1,6),PERF(1),LDAP(14,15) Debug levels indicate: <ul style="list-style-type: none"> ▪ PE(1) - A service start and end. This is an elementary trace level which is usually activated. ▪ PE(2) - Special cases. ▪ PE(3) - Includes messages related to activities performed during the FilesList service. This trace level is specifically for FilesList performance issues. ▪ PE(4) - Includes messages related to activities performed during the AddSafe service. This trace level is specifically for timing issues related to impersonated users during an AddSafe transaction. ▪ PE(5) - Includes messages related to activities performed during atomic bundle transactions. ▪ PE(6) - Special cases. ▪ PE(7) - Special cases related to error ITATS093E. ▪ PE(8) - Includes messages related to the UserBlock instantiation during a service execution. ▪ PE(9) - Special cases for HandleInactiveApplications. ▪ PE(10) - Includes messages related to errors that occurred during the DB locks mechanism. ▪ PE(13) - Includes messages related to ENE events which are invoked from Vault services. ▪ LDAP(14,15) – Detailed LDAP debug ▪ UI(8) - Includes messages related to the UserBlock instantiation during a UI action. ▪ SYSLOG(1) - Includes messages related to decisions made about Syslog jobs. ▪ SYSLOG(2) - The xml output will be added to the trace log for each Syslog written. ▪ CRYPT(2) – PKI authentication debug messages (relevant only from vault 7.1.1) ▪ *DM(13) - The query will be written to the trace for each MySQL query. ▪ *CRYPT(1) - Special case for Cryptolib. ▪ *SM(13) – Detailed Database access debug ▪ *DM(13) – Detailed Database debug ▪ *UI(1,2),COMM(1,2),CONNPOOL(1,2) – Detailed communication debug * Consult CyberArk Technical Support before applying these settings. ▪ TraceArchive MaxSize - Determines the maximum size of a trace file archive folder, essentially determining how many trace files will be saved. <ul style="list-style-type: none"> ▪ Default value is 5120. Must be greater than 200MB or -1 to disable

Log files location	<ul style="list-style-type: none"> ▪ Italog.log ▪ Trace.dX (X – number from 0 to 4) ▪ Archive – Trace archive folder ▪ \Database\VaultDB.log - Database log
--------------------	---

PARAgent

Configuration file(s)	▪ PARAgent.ini
To set debug mode (version 6.0 patch #3 or higher):	▪ EnableTrace=yes
Log files location	▪ PARAgent.log

DR

Configuration file(s)	▪ PADR.ini
To set debug mode	<ul style="list-style-type: none"> ▪ EnableTrace=yes – Includes debug, activity, and error messages in the log file. ▪ EnableTrace=full – Includes communication related messages in the log file, in addition to debug, activity, and error messages.
Log files location	<ul style="list-style-type: none"> ▪ PADR.log (version 5.5 or higher also contains the replicate information) ▪ PAReplicate.log (version 5.0 or lower)

PAReplicate

Configuration file(s)	-
To set debug mode	▪ Execute the PAReplicate.exe command, add the following flag: /EnableTrace
Log files location	▪ PAReplicate.log

PrivateArk Client

Configuration file(s)	-
To set debug mode	In the PrivateArk Client: Tools→Options→Advanced→Log Configuration...
Log files location	<ul style="list-style-type: none"> ▪ WinXP / Win2003: \Documents and Settings\<user>\Application Data\CyberArk\PrivateArk\PALog.txt</user> ▪ Win7 / Win2008: \Users\<user>\AppData\Roaming\CyberArk\PrivateArk</user>
Executable	▪ PAInfo.exe

CPM

Configuration file(s)	<ul style="list-style-type: none"> ▪ PasswordManagerShared Safe → Root\policies\<platform>.ini ▪ \Program Files\CyberArk\Password Manager\bin\<process>.ini ▪ \Program Files\CyberArk\Password Manager\bin\<prompts>.ini
To set debug mode	<ul style="list-style-type: none"> ▪ For CPM - CPM.ini (or via PVWA System Configuration): <ul style="list-style-type: none"> ▪ CPMDebugLevels=2 (default) <ul style="list-style-type: none"> ▪ 0 - No messages will be written to the trace log. ▪ 1 - CPM exceptions will be written to the trace log. This is the default debug level. ▪ 2 - CPM trace messages will be written to the trace log. ▪ 3 - CPM CASOS activities will be written to the trace log. ▪ 4 - CPM CASOS debug activities will be written to the trace log. ▪ 5 - CPM CASOS errors will be written to the trace log. ▪ 6 - All CPM CASOS activities and errors will be written to the trace log. ▪ OldLogRetention – The number of days that trace and console log files will be saved, after which they will be deleted. By default, log files are saved for seven days. To prevent old files from being deleted, specify 0 (zero). ▪ For policy handling - In the policy (must be under “ExtraInfo” section and case sensitive): <ul style="list-style-type: none"> ▪ Debug=Yes ▪ For PMTerminal (\bin\<process>.ini) <pre>[Debug Information] DebugLogFullParsingInfo=yes DebugLogFullExecutionInfo=yes DebugLogDetailBuiltInActions=yes ExpectLog=yes ConsoleOutput=yes</pre> ▪ For Windows AD Autodetect (PVWA → System Configuration page → Auto-Detection button → Auto-detection processes → <Auto-detection process>): <ul style="list-style-type: none"> ▪ In Machine Detection, select the relevant detection section): <ul style="list-style-type: none"> ▪ ADLDAPDebug=Yes ▪ In Machine Scan → Usage Types → <UsageName> → Usage Parameters: <ul style="list-style-type: none"> ▪ ADUsageParameterName=Debug ▪ ADUsageParameterValue=Yes
Log files location	<ul style="list-style-type: none"> ▪ CPM <ul style="list-style-type: none"> \Program Files\CyberArk\PasswordManager\Logs\pm.log \Program Files\CyberArk\PasswordManager\Logs\pm_error.log \Program Files\CyberArk\PasswordManager\Logs\PMConsole.log \Program Files\CyberArk\PasswordManager\Logs\PMTrace.log ▪ All plug-ins <ul style="list-style-type: none"> \Program Files\CyberArk\PasswordManager\Logs\ThirdParty*.log ▪ For Windows AD Autodetect <ul style="list-style-type: none"> \Program Files\CyberArk\PasswordManager\Logs\ThirdParty LDAP-Debug.log

PVWA

Configuration file(s)	<ul style="list-style-type: none"> ▪ \wwwroot\PasswordVault\web.config ▪ PVWAConfig Safe → Root\PVConfiguration.xml ▪ PVWAConfig Safe → Root\Policies.xml
To set debug mode	<p>Debug version 4.6 or lower: PVWAConfig Safe → Root\PVConfiguration.xml)</p> <p>Version 5.0 or higher:</p> <ol style="list-style-type: none"> 1. In the PVWA, display the System Configuration page. 2. In the Web Access section, click Options and then Logging: <ul style="list-style-type: none"> ▪ DebugLevel=High (None/High/Low) ▪ InformationLevel=High (None/High/Low) ▪ Profiling – Provides additional information about performance
Log files location	<ul style="list-style-type: none"> ▪ %windir%\temp\ or ▪ The LogFolder parameter in web.config in the IIS PasswordVault folder: <ul style="list-style-type: none"> ▪ CyberArk.WebApplication.log ▪ CyberArk.WebConsole.log ▪ CyberArk.WebSession.<SessionId>.log

ScheduledTasks

Configuration file(s)	<ul style="list-style-type: none"> ▪ C:\CyberArk\Password Vault Web Access\Services\CyberArkScheduledTasks.exe.config
To set debug mode	<ul style="list-style-type: none"> ▪ Key="EnableTrace" value="true" ▪ In the IIS Web Site Properties, in the Web Site tab, select Enable Logging.
Log files location	<ul style="list-style-type: none"> ▪ C:\CyberArk\Password Vault Web Access\Logs or ▪ The LogFolder parameter in CyberArkScheduledTasks.exe.config: <ul style="list-style-type: none"> ▪ CyberArk.WebConsole.log ▪ CyberArk.WebTasksService.log ▪ IIS Logs: <ul style="list-style-type: none"> ▪ C:\Windows\System32\LogFiles <p>or</p> <ol style="list-style-type: none"> 1. Run Internet Information Services (IIS): Start → Run → type "inetmgr" then click Enter. 2. Find your Web site in the tree on the left. 3. Right-click it, then select Properties. 4. In the Web site tab, display Active Log Format, then click Properties. 5. In the General Properties tab, the log file directory and the log file name are displayed. The full log path is comprised of the log file directory and the first part of the log file name. <ul style="list-style-type: none"> ▪ Files: <ul style="list-style-type: none"> ▪ W3SVC1\exyymmdd.log

PSM

Configuration file(s)	<ul style="list-style-type: none"> ▪ \Program Files\CyberArk\PSM\Basic_psm.ini ▪ PVWA → System tab → Options → Privileged Session Management
To set debug mode	<p>PVWA → System tab → Options → Privileged Session Management → General Settings</p> <ul style="list-style-type: none"> ▪ Server Settings → TraceLevels=1,2,3,4,5,6,7 ▪ Recorder Settings → TraceLevels=1,2 ▪ Connection Client Settings → TraceLevels=1,2 <p>TraceLevels indicate:</p> <p>0 – None.</p> <p>1 – Exceptions only. Each error in the system will be sent to the trace file, whether it is recoverable or not.</p> <p>2 – Controller trace. Includes the initialization of the PSM Server, recovery procedure and configuration.</p> <p>3 – Listener trace. Each session identified by the listener is reported, whether it is handled or not.</p> <p>4 – Session trace. Includes all the work done for a session (authentication and impersonation, password retrieval, activation of components, etc.).</p> <p>5 – Uploader trace.</p> <p>6 – CASOS errors trace (Vault errors trace).</p> <p>7 – CASOS debug and activity trace.</p>
Log files location	<ul style="list-style-type: none"> ▪ <installation folder>\Logs and subfolders, or ▪ The LogsFolder parameter in Basic_psm.ini.

PSMP

Configuration file(s)	<ul style="list-style-type: none"> ▪ /etc/opt/CARKpsmp/conf/basic_psmserver.conf ▪ /etc/ssh/sshd_config ▪ PVWA → System tab → Options → Privileged Session Management`
-----------------------	---

To set debug mode	<p>PVWA → System tab → Options → Privileged Session Management → General Settings</p> <ul style="list-style-type: none"> ▪ Server Settings → TraceLevels=1,2,3,4,5 ▪ Connection Client Settings → TraceLevels=1,2
-------------------	---

To enable SSHD.log file:

1. vi /etc/ssh/sshd_config
2. At the end of the file add:


```
PSMP_OpenSSHTraceLevels 1,2
PSMP_OpenSSHLogFolder
/var/opt/CARKpsmp/logs/components
```
3. /etc/init.d/sshd restart

To enable secure.log file (/var/log/secure):

1. vi /etc/ssh/sshd_config
2. Add: LogLevel DEBUG3
3. Restart the sshd service: /etc/init.d/sshd restart

TraceLevels indicate:

0 – None.

1 – Exceptions only. Each error in the system will be sent to the trace file, whether it is recoverable or not.

2 – Controller trace. Includes the initialization of the PSMServer, recovery procedure and configuration.

3 – Session trace. Includes all the work done for a session (authentication and impersonation, password retrieval, activation of components, etc.).

4 – CASOS errors trace (Vault errors trace).

5 – CASOS debug and activity trace.

TraceLevels indicate:

0 – None.

1 – Exceptions only. Each error in the system will be sent to the trace file, whether it is recoverable or not.

2 – Trace messages. Includes the initialization of the PSMP server, recovery procedure and configuration.

Log files location	<ul style="list-style-type: none"> ▪ /var/opt/CARKpsmp/logs/ or ▪ The LogsFolder parameter in basic_psmserver.conf.
	<ul style="list-style-type: none"> ▪ /var/opt/CARKpsmp/logs/components or ▪ The PSMP_OpenSSHLogFolder parameter in /etc/ssh/sshd_config.

ENE

Configuration file(s)	<ul style="list-style-type: none"> ▪ \Program Files\PrivateArk\Server\Event Notification Engine\ENEConf.ini ▪ Notification Engine Safe→root\EventNotificationEngine.ini
To set debug mode	In EventNotificationEngine.ini: <ul style="list-style-type: none"> ▪ [Debug] ControllerDebugLevel=1,2,3,4 CollectorDebugLevel=1,2 ParserDebugLevel=1,2 SMTPSenderDebugLevel=1,2 ConfigurationManagerDebugLevel=1,2
Log files location	<ul style="list-style-type: none"> ▪ \Program Files\PrivateArk\Server\Event Notification Engine\Logs\ENEConsole.log ▪ \Program Files\PrivateArk\Server\Event Notification Engine\Logs\ENETrace.log

ExportVaultData

Configuration file(s)	-
To set debug mode	<ul style="list-style-type: none"> ▪ Create "Logs" folder under the installation folder
Log files location	<p><installation folder>\Logs</p> <ul style="list-style-type: none"> ▪ Casos.Activity.log ▪ Casos.Debug.log ▪ Casos.Error.log

Vault Activity Email Notification

Configuration file(s)	<ul style="list-style-type: none"> ▪ Conf.ini ▪ Vault location (specified in conf.ini)\VAConf.ini
To set debug mode	-
Log files location	<ul style="list-style-type: none"> ▪ Log file specified in the conf.ini file ▪ Error log file specified in the conf.ini file

NT Authentication Agent

Configuration file(s)	-
To set debug mode	<ul style="list-style-type: none"> ▪ In the registry, add a string value named debug and set its value to 1 in the following location: HKEY_LOCAL_MACHINE\SOFTWARE\CyberArk\PrivateArk\NT Authentication Agent\<version>
Log files location	<ul style="list-style-type: none"> ▪ Event Viewer application log

Credential Provider

Configuration file(s)	<ul style="list-style-type: none"> ▪ Windows: %ProgramFiles%\CyberArk\ApplicationPasswordProvider\basic_appprovider.conf ▪ Unix: /etc/opt/CARKaim/conf/basic_appprovider.conf ▪ main_appprovider.conf (name and location are specified in basic_appprovider.conf)
To set debug mode	<ul style="list-style-type: none"> ▪ CacheDebugLevels=1,2 0 – No messages will be written to the trace log. This is the default debug level. 1 – Cache errors will be written to the trace log. 2 – Cache trace messages will be written to the trace log. ▪ ProtocolDebugLevels=1,2 0 – No messages will be written to the trace log. This is the default debug level. 1 – Protocol errors will be written to the trace log. 2 – Protocol trace messages will be written to the trace log. ▪ AppProviderDebugLevels=1,2,3,4,5 0 – No messages will be written to the trace log. This is the default debug level. 1 – Provider errors will be written to the trace log. 2 – Provider trace messages will be written to the trace log. 3 – Provider CASOS errors will be written to the trace log. 4 – Provider CASOS activities and trace messages will be written to the trace log. 5 – Provider background refresh trace messages will be written to the trace log.

Log files location	<p>Windows:</p> <ul style="list-style-type: none"> ▪ %Program Files%\CyberArk\ApplicationPasswordProvider\Logs or ▪ The LogsFolder parameter in basic_appprovider.conf: <ul style="list-style-type: none"> ▪ APPAudit.log ▪ APPConsole.log ▪ APPTrace.log <p>Unix:</p> <ul style="list-style-type: none"> ▪ /var/opt/CARKaim/logs or ▪ The LogsFolder parameter in basic_appprovider.conf: <ul style="list-style-type: none"> ▪ APPAudit.log ▪ APPConsole.log ▪ APPTrace.log
--------------------	--

On-Demand Privileges Manager

Configuration file(s)	<ul style="list-style-type: none"> ▪ /etc/opt/CARKaim/conf/basic_opm.conf ▪ main_opm.conf (name and location are specified in basic_opm.conf)
To set debug mode	<ul style="list-style-type: none"> ▪ CacheDebugLevels=1,2 <ul style="list-style-type: none"> 0 – No messages will be written to the trace log. This is the default debug level. 1 – Cache errors will be written to the trace log. 2 – Cache trace messages will be written to the trace log. ▪ ProtocolDebugLevels=1,2 <ul style="list-style-type: none"> 0 – No messages will be written to the trace log. This is the default debug level. 1 – Protocol errors will be written to the trace log. 2 – Protocol trace messages will be written to the trace log. ▪ PIMSuDebugLevels=1,2,3,4,5 <ul style="list-style-type: none"> 0 – No messages will be written to the trace log. This is the default debug level. 1 – Provider errors will be written to the trace log. This has the same effect as setting the DebugLevels parameter to 1. 2 – OPM command transmission trace messages will be written to the trace log. 3 – OPM background refresh trace messages will be written to the trace log. 4 – OPM execute command service trace messages will be written to the trace log. 5 – The OPM objects created during execute command service trace messages will be written to the trace log.
Log files location	<ul style="list-style-type: none"> ▪ /var/opt/CARKaim/logs or ▪ The LogsFolder parameter in basic_opm.conf: <ul style="list-style-type: none"> ▪ APPAudit.log ▪ OPMConsole.log ▪ OPMTrace.log

Appendix D: Managing Platforms for Groups

The following tables list all the properties that can be configured for platforms that can be applied to Account Groups.

Group Manager Platform

The following table lists the properties that can be configured for platforms that can be applied to the Group Manager Account.

Property	Indicates ...
General:	
PolicyID	The unique name/ID of the group manager platform. This property is required.
PolicyName	The descriptive name of the platform. This property is required.
PolicyType	The type of platform. Specify Group to define this platform as a group manager platform. This property is required.
ImmediateInterval	The number of minutes that will elapse between when the user initiates an account management process and when the process is performed.
Interval	The number of minutes that the CPM waits between loops when processing accounts of this platform.
SearchForUsages	Whether or not CPM will search for copies of the account after it successfully changed and synchronized them. Specify Yes . This property is required.
AllowedSafes	A Safes pattern that indicates the Safes that this platform can be applied to.
Privileged Account Management:	
MinValidityPeriod	The number of minutes to wait from the last retrieval of the password until it is replaced. This gives the user a minimum period to be able to use the password before it is replaced.
ResetOverridesMinValidity	If the account is marked with the 'ResetImmediately' property, it will be changed, regardless of the period defined in the MinValidityPeriod parameter.
ResetOverridesTimeFrame	If the account is marked with the 'ResetImmediately' property, it will be changed, regardless of the time frame defined in the FromHour and ToHour parameters.
Timeout	The number of seconds to wait for the change password plug-in to finish its execution.
UnlockIfFail	Whether or not the account will be unlocked and made available to other users if it was not changed successfully. This is relevant to exclusive accounts mode only.

Property	Indicates ...
Password Change:	
AllowManualChange	Whether or not a 'Change Now' process can be initiated manually. This parameter can be specified in the group manager as well as in group members.
PerformPeriodicChange	Whether or not accounts related to this platform will be changed periodically according to the value of the ExpirationPeriod parameter.
ExpirationPeriod	The number of days after which the passwords related to this platform must be replaced. Specify -1 to ignore this property.
HeadStartInterval	The number of days before the password expires (according to the ExpirationPeriod parameter) that the CPM will initiate a password change process.
FromHour	The time from when the CPM can change passwords, either manually or automatically.
ToHour	The time until when the CPM can change passwords, either manually or automatically.
DaysNotifyPriorExpiration	The number of days before a password is changed that a notification will be sent to recipients, a re-notification interval that determines the number of days between notifications for the same password expiration (optional) and a re-notification period (optional) that determines the period of time during which these notifications will be sent. Separate these values by commas.
ExecutionDays	The days of the week when the CPM will change passwords.
Password Verification:	
VFAllowManualVerification	Whether or not a password verification process can be initiated manually in the PVWA interface. This configuration is only relevant to group member platforms.
VFPerformPeriodicVerification	Whether or not a password verification process will be performed automatically according to the number of days specified in the VFVerificationPeriod parameter.
VFVerificationPeriod	The number of days between automatic password verification processes.
VFFromHour	The time frame in hours during which the CPM can verify passwords, either manually or automatically.
VFToHour	The time frame in hours during which the CPM can verify passwords, either manually or automatically.
VFExecutionDays	The days of the week when the CPM will verify passwords.

Property	Indicates ...
Password Reconciliation:	
RCAAllowManualReconciliation	Whether or not passwords will be reconciled when a user initiates the procedure manually through the PVWA interface. This parameter can be specified in the group manager as well as in group members.
RCFFromHour	The time from when the CPM can reconcile passwords, either manually or automatically.
RCToHour	The time until when the CPM can reconcile passwords, either manually or automatically.
ReconcileAccountSafe	The name of the Safe where the reconcile account is stored or a dynamic rule to specify this value.
ReconcileAccountFolder	The name of the folder where the reconcile account is stored or a dynamic rule to specify this value.
ReconcileAccountName	The name of the reconcile account or a dynamic rule to specify this value.
RCExecutionDays	The days of the week when the CPM will reconcile passwords.
Notifications:	
NFInterval	The interval in minutes between the notification tasks.
NFFromHour	The hour when notification will begin.
NFToHour	The hour when notification will end.
NFNotifyPriorExpiration	Whether or not notifications will be sent to recipients.
NFPriorExpirationRecipients	The list of email addresses that notifications will be sent to.
NFNotifyOnUnreleasedPasswords	Whether or not specified recipients will receive notifications when an account is not released after the time defined in MinValidityPeriod. This parameter is not relevant if the platform is a group platform.
NFOnUnreleasedPassword Recipients	The email addresses of users who will receive notifications when an account is not released after the time defined in MinValidityPeriod.
NFNotifyOnPasswordDisable	Whether or not specified recipients will receive notifications when an account is disabled. This parameter is not relevant if the platform is a group platform.
NFOnPasswordDisableRecipients	The email addresses of users who will receive notifications when an account is disabled.
NFNotifyOnVerificationErrors	Whether or not specified recipients will receive notifications when an account verification process results in an error. This parameter is not relevant if the platform is a group platform.
NFOnVerificationErrorsRecipients	The email addresses of users who will receive notifications when an account verification process results in an error.

Property	Indicates ...
NFNotifyOnPasswordUsed	Whether or not specified recipients will receive notifications when an account is used. This parameter is not relevant if the platform is a group platform.
NFOnPasswordUsedRecipients	The email addresses of users who will receive notifications when an account is used.
Generate Password:	
PasswordLength	The length of the newly generated password.
MinUpperCase	The minimum number of uppercase characters in the newly generated password. To exclude upper case characters from the password, specify '-1'.
MinLowerCase	The minimum number of lower case characters in the newly generated password. To exclude lower case characters from the password, specify '-1'.
MinDigit	The minimum number of digits in the newly generated password. To exclude digits from the password, specify '-1'.
MinSpecial	The minimum number of special characters in the newly generated password. To exclude special characters from the password, specify '-1'.
PasswordForbiddenChars	The characters that cannot be used when generating a new password, e.g. "/~\`".
PasswordEffectiveLength	The number of characters in the newly generated password in which the above rules are effective. If this parameter is not specified, the PasswordLength parameter is used as the effective length.
PreventSameChar PerPrevPassPosition	Whether or not characters (alphabetic or numeric) can be used in the same positions as in the previous password. This property is relevant for AS400 (iSeries) accounts only.
PreventRepeating Characters	Whether or not characters can be used more than once in a password. This property is relevant for AS400 (iSeries) accounts only.

Appendix E: Adding Accounts with SSH Keys using the AccountUploader Utility

The **AccountUploader** utility enables you to create accounts with SSH keys.

- Copy the following files to a directory on your local unix machine from where you will run the utility:
 - AccountUploader
 - icudt42l.dat

The AccountUploader utility is supported on Linux and has the following usage:

```
AccountUploader -VaultFile VaultFile
                -CredFile CredFile
                -SafeName SafeName
                -KeyFile KeyFile
                -DeviceType DeviceType
                -PolicyId PolicyId
                -Address Address
                -UserName UserName
                [-SubnetMask SubnetMask]
                [-ObjectName ObjectName]
```

Parameter	Description
Vault file	The full or relative path of the vault.ini file of the Vault where the account will be added.
CredFile	The full or relative path of the credentials file that will be used to connect to the vault
SafeName	The name of the Safe where the account will be added.
KeyFile	The full or relative path of the SSH private key file that will be attached to the account. The SSH Key can be either in OpenSSH format or putty format (ppk).
DeviceType	The type of device on which the account will be used.
PolicyId	The ID of the platform that the account will associated with. Make sure that the specified policy supports connections with SSH keys. By default, the Unix SSH Keys platform supports these connections.
Address	The IP address or DNS of the target machine where the account will be used.
UserName	The user who will be used to connect to the target machine.
SubnetMask	The subnet mask for this account, if this is a subnet account. This parameter is optional.
ObjectName	The name by which the account will be saved in the Vault. This parameter is optional.

Appendix F: Accessing Target Machines through PSMP

The PSMP enables end users to connect to target UNIX systems from their own workstation, using either SSH or Telnet protocol. Users connect by specifying a basic command line syntax as explained below.

The PSMP can be used by any ssh client using two different syntaxes. A complete description of Option 1 appears in *Connecting Transparently with PSMP*, page 283, and the following syntax describes Option 2.

The PSMP utility runs from a command line, using the following syntax:

```
ssh -t PSMConnect@<proxyaddress> <vaultuser> <targetuser>
<targetmachine> [-protocol <telnet|ssh>] [-port <port>] [-vp <Vault-
password>]
[-tpw <targetpassword>]
```

The following table explains these parameters:

Parameter	Description	Required
proxyaddress	The IP address or DNS of the PSMP machine. For example, 1.1.1.1 or 'myhost'.	Yes
vaultuser	The name of the Vault user running this command.	Yes
targetuser	The name of the account that will be used on the target system. For example, root.	Yes
targetmachine	The address of the target system in any of the following formats: <ul style="list-style-type: none"> IPv4 – For example, 1.1.1.1 IPv6 – For example, 1000-1000-1000-1000-1000-1000-0055 <p>Note: Use hyphens instead of colons as separators.</p> <ul style="list-style-type: none"> DNS – For example, 'myhost' <p>As the PSMP resolves DNS names to IP addresses when necessary, you can specify either the machine's DNS name or an IP address, regardless of whether the account of the target machine was defined with an IP address, subnet or DNS name</p>	Yes
telnet ssh	The protocol to use for the connection. Specify either ssh (default) or telnet . If this parameter is not specified, the default protocol will be used.	No
port	The connection port used to access the system. If this is not specified, it will be taken from the account properties. If neither of these ports are specified, plink defaults. Default values are: SSH - 22 Telnet – 23	No
Vault-Password	The password of the Vault user used to retrieve the password to connect to the target machine. If this password is not specified, the user is prompted for it.	No

Parameter	Description	Required
targetpassword	The password of the target account. This parameter is only relevant when privileged SSO is not enabled and the password is not managed in the Vault. If this password is not specified, the user is prompted for it.	No

Examples

Example 1: Running sessions with Privileged SSO

The following example initiates an SSH privileged SSO session. The command contains all the information that is required to log onto the target system through the PSMP.

```
ssh -t PSMConnect@psmp.proxymachine.com john root
target.ciscorouter.com -vp johnvaultpass
```

In this example, the name of the proxy machine is **psmp.proxymachine.com**. This command will use a Vault user called **john** to access the Vault and retrieve an account for the **root** user on the target system. This user will be used to log onto a target system called **target.ciscorouter.com**. This command also includes the password of the Vault user, which is **johnvaultpass**. This command does not include the protocol or port, so the default values of **SSH** protocol and port **22** will be used.

The following example initiates a Telnet privileged SSO session.

```
ssh -t PSMConnect@psmp.proxymachine.com john root
target.ciscorouter.com -protocol telnet
```

Similar to the previous example, this command includes the name of the proxy machine and the name of the Vault user, **john**, who will access the Vault to retrieve the account for the **root** user on the target system, which is also specified. However, this command specifies **Telnet** protocol, which will automatically use port **23**. The command does not include the Vault password for **john**, so the user will be prompted for it so that the PSMP can complete the connection to the remote machine.

Example 2: Running sessions without Privileged SSO

The following example initiates a non-privileged session.

```
ssh -t PSMConnect@psmp.proxymachine.com john root
target.ciscorouter.com -vp johnvaultpass -tpw targetciscorootpass
```

This command includes the name of the proxy machine, **psmp.proxymachine.com**, and the name of the Vault user, **john**, who will access the Vault to retrieve the account for the **root** user on the target system, **target.ciscorouter.com**, which is also specified. The Vault user's password, **johnvaultpass**, is also specified. As this command does not include the protocol or port, the default values of **SSH** protocol and port **22** will be used.

This example shows a non-privileged SSO session, meaning that the account stored in the Vault for the target system is not configured for Privileged SSO and does not contain the password. Therefore, the password of the target system is specified, **targetciscorootpass**. If either this password or the Vault user's password is not specified in the command, the user is prompted for it so that the PSMP can complete the connection to the remote machine.

Accessing Target Machines Using an SSH Tunnel

The PSMP utility that was included in previous versions enables authorized users to open a tunnel to a remote SSH server. To access target machines using an SSH tunnel, use this utility with the following additional parameters:

- **-L <srcport>:desthost:destport** – A standard SSH parameter that enables port forwarding setup. For more information, refer to the manual page of the SSH client.

Specify **localhost** to replace desthost. Specify this parameter **before** the PSMConnect command.

- **-tunnel <target_port>** - The port of the target machine to which data transferred through the tunnel will be forwarded to. This is the same value as the **destport** specified above.

Specify this parameter **after** the PSMConnect command.

The following usage shows where to add these parameters in the PSMP command line utility:

```
ssh [-L < srcport>:localhost:target_port] -t
PSMConnect@<proxyaddress> <vaultuser> <targetuser> <targetmachine> [-
protocol <telnet|ssh>] [-port <port>] [-vp <vault-password>] [-tpw
<targetpassword>] [-tunnel <target_port>]
```