



# BeyondTrust

## **Privileged Identity 7.3 Admin Guide**

# Table of Contents

---

<b>Performance Notes</b> .....	14
<b>Background and Goals</b> .....	15
<b>Get Started with Privileged Identity</b> .....	16
<b>Management Console Introduction</b> .....	17
<b>Five Views</b> .....	19
<b>Switch to the Account Store View</b> .....	20
<b>Windows Systems View</b> .....	22
<b>Windows Accounts View</b> .....	24
<b>Linux/Unix Systems View</b> .....	26
<b>SSH Keys View</b> .....	27
<b>Display Options</b> .....	30
Windows Accounts Display .....	30
Store Types .....	31
Column Display Options .....	32
Key Filters .....	32
<b>Program Options</b> .....	34
Program General Tab .....	34
<b>Event Log Messages</b> .....	37
<b>Password Check-in Options</b> .....	38
<b>Client Password Storage</b> .....	39
<b>Heartbeat Monitor</b> .....	40
<b>Service Start/Stop Timeouts</b> .....	41
<b>SSH Options</b> .....	42
<b>Performance</b> .....	43
<b>Control Access to the Management Console</b> .....	44
Delegate Console Access .....	45
<b>Management Set Introduction</b> .....	49
<b>Scheduled Jobs and Deferred Processing Introduction</b> .....	51
Install Deferred & Zone Processors .....	51
View the Job Queues .....	52
View and Edit Existing Jobs .....	53

---

Deferred Processor Status and Configuration .....	55
<b>Password Change Jobs Introduction .....</b>	<b>56</b>
<b>Password Retrieval Introduction .....</b>	<b>57</b>
<b>Auditing and Alerting Introduction .....</b>	<b>58</b>
<b>Registration, Licensing, and Help .....</b>	<b>59</b>
<b>Enroll New Systems and Devices .....</b>	<b>62</b>
<b>Create Management Sets and Enroll Systems .....</b>	<b>63</b>
<b>Create Management Sets .....</b>	<b>64</b>
<b>Manage Restricted and Orphaned Systems .....</b>	<b>66</b>
<b>Configure Management Set Properties .....</b>	<b>67</b>
<b>Management Set Job Options .....</b>	<b>69</b>
<b>Dynamic Discovery of Systems and Devices .....</b>	<b>72</b>
<b>Add a Static List of Targets .....</b>	<b>73</b>
<b>Add Targets from Windows Domain Enumeration .....</b>	<b>74</b>
<b>Add Targets from Windows Active Directory Query .....</b>	<b>75</b>
<b>Add Targets from Query to LDAP .....</b>	<b>79</b>
<b>Add Targets to Query from a Data Source .....</b>	<b>80</b>
<b>Add Targets from IP Range .....</b>	<b>83</b>
<b>Add Targets from Another Management Set .....</b>	<b>84</b>
<b>Map Scanned Targets .....</b>	<b>85</b>
<b>Dynamic Mapping Settings .....</b>	<b>87</b>
<b>Configure Scan Settings .....</b>	<b>90</b>
<b>Configure Known Credentials .....</b>	<b>95</b>
<b>Enroll Linux, Unix, Mainframe and Related Systems .....</b>	<b>97</b>
Enroll AIX .....	97
Enroll AS400 .....	98
Enroll Linux, Unix and Solaris .....	98
Enroll OpenVMS .....	99
Enroll OS/390 .....	99
Enroll OSX .....	100
Enroll TN3270 or TN5250 Terminal-Based Systems .....	101
Enroll Tandem Realtime Systems .....	102
<b>Enroll VMware ESX .....</b>	<b>103</b>

---

<b>Enroll Windows Systems</b> .....	<b>105</b>
<b>Enroll Databases</b> .....	<b>109</b>
Enroll IBM DB2 .....	109
<b>Enroll MySQL and MariaDB</b> .....	<b>111</b>
<b>Enroll Oracle</b> .....	<b>112</b>
<b>Enroll PostgreSQL</b> .....	<b>114</b>
<b>Enroll Microsoft SQL Server</b> .....	<b>115</b>
<b>Enroll Sybase ASE</b> .....	<b>117</b>
<b>Enroll Teradata</b> .....	<b>118</b>
<b>Enroll LDAP Directories</b> .....	<b>119</b>
<b>Enroll Middleware, Application Servers, and Enterprise Software</b> .....	<b>121</b>
Enroll McAfee ePO .....	121
<b>Enrolling SAP</b> .....	<b>123</b>
Configure SAP Gateway for Enrollment .....	125
<b>Enroll Oracle</b> .....	<b>131</b>
Enroll Oracle WebLogic .....	131
Configure Oracle WebLogic for Enrollment .....	132
Enroll Oracle PeopleSoft .....	135
<b>Enroll IBM WebSphere</b> .....	<b>136</b>
Configure IBM WebSphere for Enrollment .....	137
<b>Enroll Network Devices</b> .....	<b>141</b>
Enroll CheckPoint Devices .....	141
<b>Enroll Cisco Devices</b> .....	<b>142</b>
<b>Enrolling Dell Remote Access Control (DRAC) Devices</b> .....	<b>144</b>
<b>Enroll F5 Devices</b> .....	<b>145</b>
<b>Enroll Fortigate Devices</b> .....	<b>146</b>
<b>Enroll Foundry Devices</b> .....	<b>147</b>
<b>Enroll HP Procurve Devices</b> .....	<b>148</b>
<b>Enroll IPMI Devices</b> .....	<b>149</b>
<b>Enroll Juniper Devices</b> .....	<b>151</b>
<b>Enroll NetApp Devices</b> .....	<b>152</b>
<b>Enroll Palo Alto Devices</b> .....	<b>153</b>
<b>Enroll Xerox Phaser Printers</b> .....	<b>154</b>



---

<b>Enroll Cloud Service Providers</b>	<b>156</b>
Enroll Amazon Web Services (AWS)	156
<b>Enroll Microsoft Azure</b>	<b>159</b>
<b>Enroll RackSpace</b>	<b>161</b>
<b>Enroll Salesforce</b>	<b>163</b>
<b>Enroll Softlayer</b>	<b>165</b>
<b>Add a Cloud Service Provider Account Store Manually</b>	<b>167</b>
<b>Enroll Custom Account Stores</b>	<b>169</b>
<b>Enroll Key Vault and Secrets Manager Providers</b>	<b>172</b>
<b>Discover Privileged Accounts</b>	<b>175</b>
<b>Alternate Administrator Accounts</b>	<b>177</b>
<b>Configure Account Discovery and Password Propagation</b>	<b>179</b>
<b>Discover Linux, Unix, OSX and Solaris Privileged Accounts &amp; Keys</b>	<b>182</b>
<b>SSH Key Discovery</b>	<b>186</b>
<b>SSH Access Rules Discovery</b>	<b>189</b>
<b>SSHD Configuration Settings Discovery</b>	<b>190</b>
<b>Sudoers Configuration Settings Discovery</b>	<b>191</b>
<b>Discover Windows Privileged Accounts</b>	<b>196</b>
<b>Discover Database Privileged Accounts</b>	<b>198</b>
Discover IBM DB2 Privileged Accounts	198
Discover Microsoft SQL Server Privileged Accounts	198
Discover MySQL and MariaDB Privileged Accounts	199
Discover Oracle Database Privileged Accounts	200
Discover PostgreSQL Privileged Accounts	201
Discover Sybase ASE Privileged Accounts	201
Discover Teradata Privileged Accounts	202
<b>Discover LDAP Directory Privileged Accounts</b>	<b>204</b>
<b>Discovering Middleware, Application Server, and Enterprise Software Privileged Accounts</b>	<b>205</b>
Discover McAfee EPO Privileged Accounts	205
Discover Oracle PeopleSoft Privileged Accounts	205
Discover SAP Privileged Accounts	206
Discover Oracle WebLogic Privileged Accounts	206

---

Discover IBM WebSphere Privileged Accounts .....	207
<b>Discover Network Device Privileged Accounts .....</b>	<b>208</b>
Discover IPMI Privileged Accounts .....	208
<b>Discover Cloud Services Privileged Accounts .....</b>	<b>209</b>
Discover Amazon Web Services Privileged Accounts .....	209
Discover IBM SoftLayer Privileged Accounts .....	209
Discover Microsoft Azure Privileged Accounts .....	210
Discover RackSpace Privileged Accounts .....	210
Discover SalesForce Privileged Accounts .....	211
<b>Discover VMware ESX Privileged Accounts .....</b>	<b>213</b>
<b>Discover Custom Account Store Privileged Accounts .....</b>	<b>214</b>
<b>Configure Scheduled Job Options .....</b>	<b>215</b>
Job Priority .....	215
<b>Change Default Job Priority .....</b>	<b>217</b>
<b>Throttle Job Creation .....</b>	<b>218</b>
<b>Create Custom Communication Types .....</b>	<b>219</b>
Create a Custom Communication Type for Shell Access .....	219
Create a Custom Communication Type for SSH Tunnel Access .....	220
Associate Custom Communications with Targets .....	222
<b>Enable Account Pooling .....</b>	<b>224</b>
<b>Configure Password Settings .....</b>	<b>227</b>
<b>Use Pre and Post Run Steps to Run Scripts and Applications .....</b>	<b>231</b>
<b>Configure Pre-Run Alerts .....</b>	<b>233</b>
<b>Configure Propagation Scope .....</b>	<b>234</b>
<b>Configure Propagation Settings .....</b>	<b>235</b>
<b>Define Run Settings .....</b>	<b>241</b>
<b>Set the Job Schedule .....</b>	<b>242</b>
<b>Manage Passwords and SSH Keys .....</b>	<b>244</b>
<b>Prepare to Manage SSH and Telnet Targets .....</b>	<b>245</b>
<b>About Response Files .....</b>	<b>246</b>
<b>About Response File Sections .....</b>	<b>251</b>
<b>Connect with SSH Keys .....</b>	<b>255</b>
<b>Change Passwords and SSH Keys .....</b>	<b>259</b>

---

<b>Manage Passwords on Linux, Unix, and Related OSs</b> .....	260
<b>AS400 and OS390 Considerations</b> .....	264
<b>ESX Considerations</b> .....	267
<b>Manage SSH Keys on Linux, Unix, and Related OSs</b> .....	270
<b>Manage Passwords on OSX</b> .....	273
<b>Manage Passwords on Windows</b> .....	275
<b>Manage Database Passwords</b> .....	277
Manage IBM DB2 Passwords .....	277
Manage MySQL and MariaDB Passwords .....	277
Manage Oracle Database Passwords .....	278
Manage Microsoft SQL Server Passwords .....	279
Manage PostgreSQL Passwords .....	279
Manage Sybase ASE Passwords .....	280
Manage Teradata Database Passwords .....	281
<b>Manage Passwords on LDAP Directories</b> .....	283
<b>Manage Passwords on McAfee ePO, PeopleSoft, and SAP NetWeaver</b> .....	284
<b>Manage Passwords on Network Devices</b> .....	285
Manage Cisco Node Passwords .....	285
Managing Passwords for SSH/Telnet Devices Not Under the Cisco Node .....	286
Manage Passwords for IPMI Devices .....	287
Manage Password on Xerox Phaser Printers .....	288
<b>Manage Passwords on WebLogic and WebSphere</b> .....	290
<b>Manage Cloud Service Provider Passwords</b> .....	291
<b>Manage Passwords on VMware ESX</b> .....	292
<b>Manage Secrets in Key Vaults and Secrets Managers</b> .....	293
<b>Verify Stored Passwords</b> .....	294
<b>Create Passwords Lists and Pre-Import Managed Passwords</b> .....	295
<b>Pre-Import Managed Passwords</b> .....	296
<b>Shared Credential Lists</b> .....	298
<b>Add Credentials to a Shared Credential List</b> .....	300
<b>Personal Password Stores</b> .....	302
<b>App-to-App Password Management</b> .....	304
<b>Web Application Access</b> .....	309

---

<b>Configure Authentication Servers</b> .....	<b>310</b>
<b>LDAP Authentication Servers</b> .....	<b>311</b>
<b>OAuth Authentication Servers</b> .....	<b>313</b>
<b>OAuth - Azure AD</b> .....	<b>314</b>
<b>OAuth - Salesforce</b> .....	<b>315</b>
<b>OAuth - Facebook</b> .....	<b>316</b>
<b>OAuth - Google</b> .....	<b>318</b>
<b>SAML Authentication Servers</b> .....	<b>320</b>
<b>SAML - ADFS</b> .....	<b>322</b>
<b>SAML - Okta</b> .....	<b>328</b>
<b>SAML - OneLogin</b> .....	<b>330</b>
<b>SAML - Ping</b> .....	<b>332</b>
<b>RADIUS Authentication Servers</b> .....	<b>334</b>
<b>Configure MFA</b> .....	<b>335</b>
<b>OATH 2-Factor</b> .....	<b>336</b>
OATH 2-Factor Overview .....	336
<b>OATH With Existing Tokens</b> .....	<b>338</b>
<b>OATH Without Existing Tokens</b> .....	<b>340</b>
<b>OATH Token Configuration</b> .....	<b>344</b>
<b>Configure OATH for Web Client Access</b> .....	<b>348</b>
<b>Additional OATH Resources</b> .....	<b>349</b>
<b>DUO via RADIUS</b> .....	<b>350</b>
<b>InfoCrypt</b> .....	<b>353</b>
<b>RADIUS 2-Factor</b> .....	<b>355</b>
<b>RADIUS 2-Factor for Explicit Accounts</b> .....	<b>357</b>
<b>RSA SecurID</b> .....	<b>360</b>
<b>Verify RSA SecurID Communication</b> .....	<b>364</b>
<b>SafeNet</b> .....	<b>366</b>
<b>YubiKey</b> .....	<b>368</b>
Using YubiKey to Authenticate to Privileged Identity .....	368
<b>Use YubiKey Configured as a Smart Card</b> .....	<b>371</b>
<b>Manage Identities and Delegations for Password and System Access</b> .....	<b>374</b>
Add Identities for Password and System Access .....	374

---

<b>Active Directory Users</b> .....	<b>375</b>
<b>Active Directory Groups</b> .....	<b>377</b>
<b>Roles for LDAP, OAuth, and SAML Users</b> .....	<b>379</b>
<b>LDAP Users</b> .....	<b>381</b>
<b>RADIUS Users</b> .....	<b>382</b>
<b>Certificates</b> .....	<b>383</b>
<b>Explicit Accounts</b> .....	<b>385</b>
<b>Administrate Managed Password Permissions</b> .....	<b>387</b>
<b>Account Masks</b> .....	<b>389</b>
<b>Self-Recovery Permissions</b> .....	<b>391</b>
<b>Global Delegations</b> .....	<b>393</b>
<b>Web Application Global Permissions in the Management Console</b> .....	<b>394</b>
<b>Import and Export Global Permissions</b> .....	<b>398</b>
<b>Web Application Global Permissions in the Web Application</b> .....	<b>401</b>
<b>Web Application Global Permissions Time Restrictions</b> .....	<b>403</b>
<b>Per-Management Set Delegations</b> .....	<b>405</b>
Web Application Per-Management Set Permissions in the Management Console .....	405
Web Application Per-Management Set Permissions in the Web Application .....	406
<b>Per-System Delegations</b> .....	<b>408</b>
Web Application Per System Permissions in the Management Console .....	408
Web Application Per-System Permissions in the Web Application .....	410
<b>Per-Account Delegations</b> .....	<b>411</b>
Web Application Per Account Permissions in the Management Console .....	411
Web Application Per-Account Permissions in the Web Application .....	412
<b>Per-Job Delegations</b> .....	<b>414</b>
Web Application Per-Job Permissions in the Management Console .....	414
Web Application Per-Job Permissions in the Web Application .....	414
<b>Assign SSH Key Permissions in the Management Console</b> .....	<b>416</b>
<b>Assign SSH Key Permissions in the Web Application</b> .....	<b>421</b>
<b>Request and Authorize SSH Keys from the Web Application</b> .....	<b>422</b>
<b>Manage Shared Credential List Permissions</b> .....	<b>423</b>
Shared Credential List Permissions in the Management Console .....	423
Shared Credential List Permissions in the Web Application .....	425

---

<b>Retrieve Passwords</b> .....	426
<b>Retrieve Managed Passwords</b> .....	427
<b>Retrieve Managed Passwords from the Web Application</b> .....	429
<b>Request and Grant Access to Managed Passwords from the Web Application</b> .....	431
<b>View Managed Passwords in the Management Console</b> .....	433
<b>Retrieve Shared Managed and Stored Passwords</b> .....	435
<b>Retrieve Shared Passwords from the Web Application</b> .....	437
<b>Request and Grant Access to Shared Passwords from the Web Application</b> .....	439
<b>View Shared Passwords in the Management Console</b> .....	441
<b>Password History</b> .....	442
<b>Retrieve Personal Passwords</b> .....	444
<b>Work with Compartmentalized Passwords (Four Eyes)</b> .....	446
<b>Share Personal Passwords</b> .....	449
Share Personal Passwords .....	449
<b>Use Additional Features in the Web Application</b> .....	451
Configure Management Sets in the Web Application .....	451
<b>Manage Jobs in the Web Application</b> .....	455
<b>Elevate Accounts</b> .....	462
Configure Account Elevation in the Web Application .....	462
Configure Programmatic Account Elevation .....	462
<b>Self-Service Elevation - Simple</b> .....	463
<b>Self-Service Elevation - Advanced</b> .....	465
Self-Elevation Permissions .....	465
Perform Self-Elevation .....	466
<b>Arbitrary Account Elevation</b> .....	467
<b>Elevate Accounts for Linux Systems</b> .....	468
Linux Elevation in Privileged Identity Versions 5.5.0 - 6.x .....	469
<b>Use the Secure File Store</b> .....	474
<b>Manage File Store Settings</b> .....	477
<b>Web Application Settings</b> .....	478
<b>Session Settings</b> .....	479
Current Session Information .....	479
<b>Main Panel Configuration</b> .....	481

---

<b>User Settings</b>	482
<b>RDP Settings</b>	483
<b>SSH Settings</b>	485
<b>Server Certificates</b>	487
<b>Web Services</b>	488
<b>Message Center</b>	489
<b>Configure Delegation</b>	490
<b>Customize Web Content</b>	491
<b>Message Templates</b>	493
<b>Remote Applications</b>	496
<b>Live Activity</b>	497
<b>Unlock Locked Out Accounts</b>	498
<b>Configure Event Sinks in the Web Application</b>	499
<b>Create an Event Sink</b>	500
<b>Configure Log File Event Output Type</b>	503
<b>Configure Set Registry Value Event Output Type</b>	504
<b>Configure Named Pipe Event Output Type</b>	505
<b>Configure COM Call Event Output Type</b>	506
Configure Send Email Event Output Type	506
Configure Event Log Event Output Type	507
Configure Syslog Compatible Event Output Types	508
Configure MSMQ Event Output Type	509
Configure Run Specific Application Event Output Type	510
Configure Pre-Built Ticketing and Logging Event Output Types	511
<b>Configure Propagation Types in the Web Application</b>	512
<b>Configure Site Settings</b>	515
<b>Audits and Alerts</b>	525
Charts	525
<b>Compliance Reports</b>	528
<b>Delegations Reporting</b>	532
<b>Report Generator</b>	533
<b>Audit Web Activity</b>	534
<b>Alerts and Integration Using Event Sinks</b>	535

---

<b>Event Sink Events List</b> .....	<b>536</b>
<b>Configure Event Sinks in the Management Console</b> .....	<b>542</b>
Configure Event Sink Logging Options .....	542
Create an Event Sink .....	542
Configure Event Sink Output .....	544
Restrict Event Sink to a Specific Zone .....	545
<b>Configure Log File Event Output Type</b> .....	<b>546</b>
<b>Configure Update Registry Value Event Output Type</b> .....	<b>547</b>
<b>Configure Named Pipe Event Output Type</b> .....	<b>548</b>
<b>Configure COM Call Event Output Type</b> .....	<b>549</b>
<b>Configure Send Email Event Output Type</b> .....	<b>550</b>
<b>Configure Windows Event Log Event Output Type</b> .....	<b>551</b>
<b>Configure Syslog Compatible Event Output Types</b> .....	<b>552</b>
<b>Configure MSMQ Event Output Type</b> .....	<b>553</b>
<b>Configure Run a Specified Application Output Type</b> .....	<b>554</b>
<b>Pre-Built Ticketing and Logging Integrations</b> .....	<b>555</b>
ArcSight .....	555
<b>BMC Remedy</b> .....	<b>556</b>
<b>HP Service Manager</b> .....	<b>557</b>
<b>Jira</b> .....	<b>558</b>
<b>OTRS</b> .....	<b>559</b>
<b>CA Service Desk Manager</b> .....	<b>560</b>
<b>QRadar</b> .....	<b>561</b>
<b>ServiceNow</b> .....	<b>562</b>
<b>Microsoft System Center Service Manager</b> .....	<b>563</b>
<b>RSA NetWitness (enVision)</b> .....	<b>564</b>
<b>Event Sink Descriptors and Modifiers</b> .....	<b>566</b>
Event Data Message Format .....	566
<b>Event Sink Transform Files</b> .....	<b>568</b>
<b>Event Sink XML File Format</b> .....	<b>570</b>
<b>Privileged Sessions</b> .....	<b>573</b>
<b>Program Maintenance &amp; Security</b> .....	<b>574</b>
Application Components .....	574



---

<b>Database Maintenance</b> .....	<b>575</b>
<b>SQL Server Auto-Index Tuning</b> .....	<b>576</b>
<b>SQL Server Index Defragmentation</b> .....	<b>578</b>
<b>SQL Server Generate Stats FullScan</b> .....	<b>579</b>
<b>App Data Store Maintenance</b> .....	<b>580</b>
<b>Security Considerations</b> .....	<b>582</b>
<b>Addenda</b> .....	<b>587</b>
Help Desk Integrations on Remote Systems .....	587
<b>Host Server Patching, Anti-Virus &amp; IDS/IPS</b> .....	<b>588</b>
<b>Namespace Values</b> .....	<b>589</b>
<b>Privileged Identity Limited Warranty</b> .....	<b>591</b>
<b>Privileged Identity License Agreement</b> .....	<b>592</b>

## Performance Notes

Privileged Identity is multi-threaded and supports automatic retry for failed systems in an operation. At the default settings of 100 threads (100 simultaneous connections) on a well-connected network (100Mbps) where all systems are accessible, password change performance is typically 400-500 machines per minute for a simple password change (not including propagation steps). This is not a guarantee of service because offline systems, high-latency, low-bandwidth, and unhealthy systems can affect performance. That said, a single Privileged Identity node, hosted on Windows Server 2012 R2 or later, can easily scale to 1,000 to 2,000 nodes per minute just by changing one setting in the product to boost the maximum thread count.

It is highly recommended to run this product on the most current version of Windows Server for best performance. With the introduction of SMB 3.0, Windows management and threading performance was significantly enhanced, with most customers able to spawn 250-300 threads or more simultaneously rather than only 100 threads. Windows 2008 R2 will likely encounter thread loss and thread abandonment past 200 threads when managing other Windows systems due to limitations in the SMB 2.0 stack.

You can tune threading options up or down by changing the maximum number of threads that will be dispatched from **Settings > Program Options > Program General > Maximum number of threads that will be dispatched**. Variances in customer environments and provided hardware may permit more simultaneous threads or may require a reduced number of threads.

All scheduled operations and job retries are handled in the background by a deferred processor service. The effect on network traffic during an operation using the default settings is about 2% of available bandwidth. (In Windows environments this is equivalent to WINS type traffic.) Typically, target machine impact will not be noticed (CPU, Memory, Hard Disk, Network) but will vary based on the type of operation performed (for example, changing an account password versus changing a password change and restarting a service).

## Background and Goals

### The Need for Strong Local Credentials

If your organization needs even the most basic access security, you should use unique local login credentials customized for each workstation and server in your environment. Unfortunately, most organizations use common credentials, with the same username and password for the built-in admin account on each system, because it's easier for IT to manage those systems. However, this does not take into account the consequences to the organization should these common credentials be compromised.

With the mandates of PCI-DSS, Sarbanes-Oxley, HIPAA, Gramm-Leach-Bliley, California Security Breach Information Acts, NASD 3010, SEC 17a-4, 21 CFR Part 11, DoD 5015.2, and others, you must implement reasonably secure local login credentials not only to protect the confidentiality of your data but also to protect against tampering.

### Creating Strong Local Credentials

Privileged Identity can change any common account on all workstations and servers in just a few minutes without needing scripts or any other type of program. These new common credentials can be stored in a local or remote SQL Server database and can be recovered on demand using the web application.

You can configure a schedule on Privileged Identity to regularly change common account passwords on all target systems (e.g., the built-in admin account on a workstation), making sure that each account regularly receives a cryptographically strong password. This feature protects the overall security of an organization so that even if a single machine's local admin password is compromised, this does not lead to the total compromise of the entire organization's security. Privileged Identity further builds on these concepts by automatically discovering all references to the specified account, such as services, tasks, COM and DCOM objects, and more; when an account, either domain or local, has a password change, Privileged Identity propagates the new password to all discovered references.

### Delegated Password Recovery

Privileged Identity also contains a web client to allow remote recovery of passwords, access to privileged sessions, and more. The web app is made up of ASP and ASP.NET web pages. Any user with appropriate group memberships may use the app and may recover passwords for managed accounts. All access to the web app and all actions taken therein are logged; authorized users can view this history via the same web interface.

Because Privileged Identity protects and provides extremely sensitive information, it is essential that you pay particular attention to the application's security settings and that you use an appropriate encryption method, such as SSL, based on the scope of access provided.

## Get Started with Privileged Identity

This section covers basic concepts for getting started with Privileged Identity, post installation.

The following are suggested steps to get started with Privileged Identity:

1. Become familiar with the management console, which is PI's administrative console.
2. Create a list of systems and devices to manage.
3. Configure the deferred processor and possibly zone processors to handle scheduled jobs.
4. Create a password change job.
5. Log into to the web application and retrieve a password.
6. Audit access to the managed credentials or other data.



*For more information on the above, please see the following:*

- *"Management Set Introduction" on page 49*
- *"Scheduled Jobs and Deferred Processing Introduction" on page 51*
- *"Password Change Jobs Introduction" on page 56*
- *"Password Retrieval Introduction" on page 57*
- *"Auditing and Alerting Introduction" on page 58*

## Management Console Introduction

The management console is the administrative console where administrators add systems to manage, configure password change jobs, and configure settings for the scheduling processors and web application. By default, any user who is both an administrator of the system where the console is installed, and who has rights to the SQL database, is able to access the management console. To further restrict who has access to the management console, please see Controlling Access to the Admin Console.

Following the first launch of Privileged Identity, the program will open to its default view: Windows Systems view, Default management set. This will be the view every user sees the first time they open the management console. Subsequent views from the same user profile will automatically open to the last view with all the previous filters intact. For more information about views, see Five Views.

The left side of the management console is called the **Actions** pane. The **Actions** pane houses shortcuts to commonly used features including:

- **Add systems:** Manually add Windows systems to the current management set.
- **Change Passwords:** Start the password change process for selected systems and devices. Systems and devices must be of same type.
- **Jobs:** View all jobs (current, past, and pending).
- **Management Sets:** Create management sets and modify active management sets.
- **Set Properties:** Modify the dynamic properties of the current management set.
- **Manage Web App:** Manage all web application installations. Use this dialog to add, remove, or modify a web application instance.
- **Compliance:** Gather reporting data to generate compliance reports for activity performed in the product. This is also accessible in the web application when **Grant All Access** or **View Web Activity Logs** is granted.

The rest of the features are accessed through the various menu options across the top of the dialog.

The main panel shows the list of systems (or accounts or keys, based on the view option selected from the **View** menu) and provides information for each system or account. Because it is quite possible to have a large number of systems in a set, there are search filter settings applied to the list at all times. This allows inclusion of many more systems in the list than would be feasible to draw in a reasonable amount of time. These search filters are located directly above the system list or may be accessed in their entirety from **View | Display Options**. A filter may be defined by system name (or account name) using wild cards (using dos-style \* wild card replacements) and also by the maximum number of items to display. Keep in mind that these filter settings only affect what is drawn into the display list and there may be many more computers or accounts than are currently shown based on the applied filter settings. Filter settings can also be relaxed to include the entire list of systems if desired though this may impact display performance. Once the filter options are changed, click the **refresh** button in the upper right (with the image of a magnifier), to refresh the display.

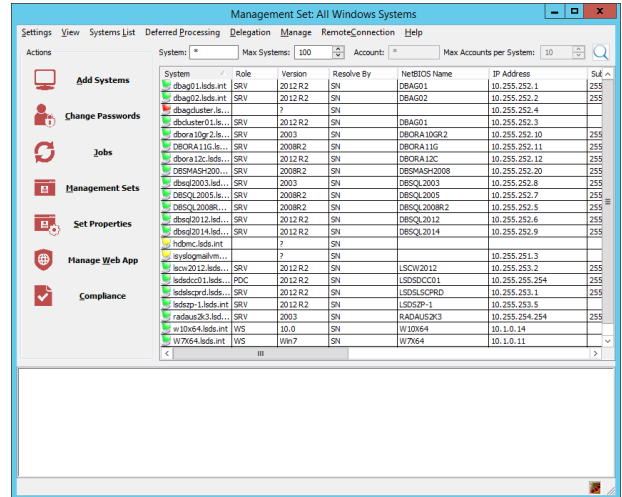
The text field at the bottom of the dialog is a live view of the program log. This is the text log file that records all activity performed by the management console. As operations are performed interactively, logging updates scroll through this window. The log may be examined to see what happened during or after any operation by scrolling the current operation or double-click the log window to open the log file using the default text editor or viewed from **Settings > Logging Options**. All interactive operations are recorded to this log.

The account store view and systems views contain various visual indicators. For Windows systems, last operation attempt status are color coded. A red system means the last operation fails. A green system indicates the last operation succeeded. A yellow system indicates no operation has yet been performed. For systems under Linux/Unix view or node, a system with a red line through it indicates the last operation failed.

The view depicted in the image below is the Windows Systems view. Notice some systems are green, red, or yellow.

Nearly all the features of Privileged Identity can be found from the various menus.

- **Settings:** This menu contains program-level configuration options, as well as access to logging options, and web application installation/configuration.
- **View:** Use this menu to switch between the five views and configure the display option filters.
- **SystemsList:** This menu contains managed system and management set features. Use it to add and remove systems from the current management set, and add, configure, or remove additional management sets.
- **DeferredProcessing:** This menu contains options to access the deferred jobs monitor and retry settings used for automatically retried jobs.
- **Delegation:** This menu contains delegation options for password recovery and console access.
- **Manage:** This menu contains options for changing, recovering, and testing managed passwords.
- **RemoteConnection:** This menu contains options for remotely connecting to managed systems via VNC, remote desktop, SSH (separate download), and Telnet.
- **Help:** This menu contains access to about information, licensing displays, registration information and support material.



## Five Views

The management console has five views accessible from the **View** menu:

- **Account Store View:** Shows all system types, all accounts (on those that support discovery), and when service account manage is enabled, all propagation targets. All platforms will be managed from this dialog. When using Privileged Identity, it is not necessary to pre-populate this view with account discovery or refresh this view in order to perform password management, or in the case of service account management, use password propagation.
- **Windows Systems View:** Only shows Windows systems from the current management set. This view shows information about the system, such as NetBIOS name, IP, subnet mask, and last connection attempt status. Although discovery can be initiated/scheduled from this view, discovery information is not accessible from this view. If a non-Windows system is added to the Windows Systems View, it is added to the appropriate node as visible in the Account Store View when its password is changed (using any other non-Windows setting).
- **Windows Account View:** Only shows Windows accounts that have been discovered on managed Windows systems in the current management set. This view can show information about password age, account status, last login, and account flags (account properties).
- **Linux Systems View:** Only shows Linux/Unix systems from the current management set. This view shows information about the system such as the system name according to the system, IP, subnet mast, and last connection attempt status.
- **SSH Keys View:** The SSH Keys view provides a display of all discovered SSH keys, including the originating host and account information, as well as other machines where those keys are present, regardless of the management set you are currently viewing.

## Switch to the Account Store View

To switch to the account store, click **View > Account Store View**.

The primary difference between the account store view and any other view is that the account store view shows:

- A list of systems, devices, and accounts from any supported platform in the current management set.
- How the accounts are used (once account usage has been discovered).

In order to add Linux/Unix, MS SQL, MySQL, Oracle, or other systems into this display, right-click the appropriate header and choose **Add {Platform}**, or click **Systems List > Add Systems to Management Set**, and then select your preferred method.

**i** For details on adding systems, please see ["Enroll New Systems and Devices" on page 62](#). For prerequisites needed to communicate with different systems, please see [Managed Computer and Devices Requirements](#).

Windows Systems	1 Windows Systems	Windows Systems
Linux/Unix Systems	0 Linux/Unix Systems	Linux/Unix Systems
Cisco Devices	0 Cisco Devices	Cisco Devices
AS-400 Systems	0 AS-400 Systems	AS-400 Systems
OS/390 Mainframes	0 OS/390 Mainframes	OS/390 Mainframes
IPMI Devices	0 IPMI Devices	IPMI Devices
DRAC Devices	0 DRAC Devices	DRAC Devices
SQL Server Instances	1 SQL Server Instances	SQL Server Instances
Oracle Databases	1 Oracle Databases	Oracle Databases
Sybase ASE Databases	1 Sybase ASE Databases	Sybase ASE Databases
MySQL Database Instances	1 MySQL Database Instances	MySQL Database Instances
PostgreSQL Database Instances	1 PostgreSQL Database Instances	PostgreSQL Database Instances
Teradata Database Instances	1 Teradata Database Instances	Teradata Database Instances
Xerox Phaser Printers	0 Xerox Phaser Printers	Xerox Phaser Printers
Oracle Internet Directories	0 Oracle Internet Directories	Oracle Internet Directories
Novell eDirectory Databases	0 Novell eDirectory Databases	Novell eDirectory Databases
IBM Tivoli Directories	0 IBM Tivoli Directories	IBM Tivoli Directories
ViewDS Directories	0 ViewDS Directories	ViewDS Directories
PaloAlto	0 PaloAlto	PaloAlto
Juniper	0 Juniper	Juniper
DB2 Databases	0 DB2 Databases	DB2 Databases
ePolicy Orchestrator	0 ePolicy Orchestrator	ePolicy Orchestrator
SAP	0 SAP	SAP
Oracle WebLogic	0 Oracle WebLogic	Oracle WebLogic
IBM WebSphere	0 IBM WebSphere	IBM WebSphere
Azure Active Directory	0 Azure Active Directory	Azure Active Directory
Amazon Web Services	0 Amazon Web Services	Amazon Web Services
RackSpace Public Cloud	0 RackSpace Public Cloud	RackSpace Public Cloud
Force.com	0 Force.com	Force.com
SoftLayer	0 SoftLayer	SoftLayer
VMWare (ESX)	0 VMWare (ESX)	VMWare (ESX)
Un-categorized Targets	0 Un-categorized Targets	Un-categorized Targets
Explicitly Categorized Targets	0 Explicitly Categorized Targets	Explicitly Categorized Targets

To view accounts and their usage after the systems list is populated:

1. Select all desired systems in the list.
2. Next, either:
  - Right-click and select **Refresh system and discover local account usage**, or
  - Select **Systems List > Refresh Information > Refresh System and Account Information**.

This reconnects to the systems and scans the various subsystems to retrieve the list of accounts and the account usage information.

To retrieve all trusting domains where the domain accounts may be valid, either:

1. Right click and select **Refresh Trusted Domains and Systems for this System**, or
2. Select **Systems List > Refresh Information > Refresh System Trust Information**.

If this step is not performed at least once, account usage will be unavailable. After the first time this step is performed, it must be repeated only when new domains are added or removed.

**Note:** To discover SCOM run-as account usage, place the SCOM SDK binaries in the program installation directory. Typically, these files are found in `%programfiles%\System Center Operations Manager 2007\SDK Binaries`.


Once the account information and usage information have been refreshed, more information becomes available. The specific items that are shown can be customized by choosing **View > Display Options**.

**i** For more information on display options, please see ["Display Options" on page 30](#).



When the solution detects that an account is in use, it reports a number greater than 0 in the **Count** column. As each account is expanded, the screen shows the domains where the account could be in use, as well as the possible systems the account could be used on. Each Windows computer has an in-use count. Expand an account to view each location on that system where the account is used.

Initiate password changes from this view by selecting an account and clicking **Change Passwords** in the navigation pane or by right-clicking and selecting **Change Password**. For Windows accounts, basic account maintenance may also be performed from this view, including enabling, disabling, or deleting accounts.

 **Note:** Refreshing this display can take some time depending on many factors, such as the systems being refreshed and the number of accounts being discovered.

Windows Systems	1 Windows Systems	Windows Systems
1 Windows systems shown	31 Windows Accounts	Windows Server 2012 R2 Domain Con...
isc.ent	31 Windows Accounts	
Windows Accounts		
13 Windows accounts shown (18 H...		
LSC\arcweb	1 Trusting Domains	Windows user account
LSC\erpmdcfproc	1 Trusting Domains	Windows user account
LSC\erpnmweb	1 Trusting Domains	Windows user account
LSC\scadmin	1 Trusting Domains	Windows user account
LSC\scvcaect	1 Trusting Domains	Windows user account
1 valid domains shown	10/6/2016 10:14:18 AM	
All Discovered Uses	19 Total Uses	
(MSDB) MSSQLSERVER	10/6/2016 10:14:41 AM	SQL Reporting Services
(A2K332-PT) LSC Protection ...	2/14/2017 12:14:07 AM	Windows Services
(A2K332-PT) Backup Program	2/14/2017 12:14:07 AM	Tasks
(A2K8R2-PT) LSC Protection ...	2/14/2017 12:14:07 AM	Windows Services
(A2012R2-PT) LSC Protecbo...	2/14/2017 12:14:07 AM	Windows Services
(A2K8R2-PT) Backup Program	2/14/2017 12:14:08 AM	Tasks
(A2012R2-PT) Backup Program	2/14/2017 12:14:08 AM	Tasks
(A2K8R2-PT) PWCWebSDK	2/14/2017 12:14:08 AM	COM+ Applications
(A2012R2-PT) PWCWebSDK	2/14/2017 12:14:08 AM	COM+ Applications
(A2K332-PT) PWCWebSDK	2/14/2017 12:14:10 AM	COM+ Applications
(A2K8R2-PT) LSCAppPool	2/14/2017 12:14:12 AM	IIS 7 Account Info
(A2K8R2-PT) Anonymous Au...	2/14/2017 12:14:12 AM	IIS 7 Account Info
(A2K8R2-PT) PhysicalPathCr...	2/14/2017 12:14:12 AM	IIS 7 Account Info
(A2012R2-PT) LSCAppPool	2/14/2017 12:14:13 AM	IIS 7 Account Info
(A2012R2-PT) Anonymous A...	2/14/2017 12:14:13 AM	IIS 7 Account Info
(A2012R2-PT) PhysicalPathC...	2/14/2017 12:14:13 AM	IIS 7 Account Info
(A2K332-PT) IIS directory L...	2/14/2017 12:14:19 AM	IIS 6 Metabase Account Info
(A2K332-PT) IIS directory L...	2/14/2017 12:14:19 AM	IIS 6 Metabase Account Info
(A2K332-PT) AppPool: LSCA...	2/14/2017 12:14:19 AM	IIS 6 Metabase Account Info

## Hide Unused System Types

Use either of the following options to hide unused system types:

- In the account store view, right-click a system type and select **Hide Type**.
- To hide multiple system types, click **View > Display Options > Store Types**. Deselect any store types you don't want displayed.
- To hide custom account store types, click **View > Display Options > Store Types**. Click the **Extension Types** button at the bottom of the pane, highlight the store types to hide, and then click the **Hide/Show** button.

## Show Previously Hidden System Types

- Click **View > Display Options > Store Types**. Check the box for any store types you want to display.
- To show custom account store types, click **View > Display Options > Store Types**. Click the **Extension Types** button at the bottom of the pane, highlight the store types to show, and then click the **Hide/Show** button.

## Windows Systems View

To switch to the Windows Systems view, click **View > Windows Systems View**.

The Windows Systems view shows only Windows computer systems from the current management set. If any other system type is added to this view and then later managed as a non-Windows system type, it will be removed from the Windows Systems view and recategorized as the new system type and will be visible in the **Account Store View**.

The Windows Systems view shows all the Windows systems and various bits of information about the system such as its name, role in the network IP address, and so on. This information is queried directly from the target system and the information displayed is the result of that query. The name shown in the **System** column to locate the system and then perform the various queries.

The information presented in these columns is also historical data. That is, it is only relevant as of the last time a system refresh was performed (see "Checked" column). System refreshes are performed during password change jobs or system refresh operations (F5, right-click refresh).

Status is either **<OK>** or a hex error such as **0x00000035**. Errors are provided in hexadecimal format (as provided by the operating system) and should be converted to decimal for proper processing. As Privileged Identity does not perform its own networking or authentication, it is important to note that returned status messages are returned by the host operating system, and reported to you by Privileged Identity. All troubleshooting should start outside of Privileged Identity for errors listed here. A simple test is trying the UNC path to an administrative share, such as **\\ServerName\admin\$**. This test validates ability to resolve the name of the machine, admin access, and proper service status on the target system.

When in Windows Systems View, and systems are refreshed, system information for each system contacted will be retrieved and shown in the display. The columns available for display are:

- **Role:** WS for workstations and SRV for servers. This shows the primary role of the system.
- **Version:** 2008, 2012, 2016, etc. This shows the operating system version.
- **Resolve By:** SN (System Name), NB (NetBIOS), or IP (IP Address). This shows what mechanism is used to contact the computer on the network.
- **NetBIOS Name:** Discovered NetBIOS name of the host.
- **IP Address:** Discovered IP address of the host (query to the system).
- **Subnet Mask:** Discovered subnet mask of the host.
- **DHCP:** Shows whether or not the IP address for this system is assigned through DHCP.
- **MAC Address:** The hardware address of this computer.
- **Checked:** The last time this computer was successfully contacted
- **Status:** The last result message or error code for any operations on this computer.

System	Role	Version	Resolve By	NetBIOS Name	IP Address	Subnet Mask	DHCP	MAC Address	Checked	Status
A2012R2-PT	SRV	2012 R2	SN	A2012R2-PT	192.168.253.8	255.255.255.0	YES	00:15:SD:FE:08:01	3/8/2017 10:43:03 AM	<OK>
A2012-PT	SRV	2008	SN	A2012-PT	192.168.253.11	255.255.255.0	YES	00:15:SD:FE:08:0F	3/8/2017 10:43:06 AM	<OK>
A20R2-PT	SRV	2008R2	SN	A20R2-PT	192.168.253.6	255.255.255.0	YES	00:15:SD:FE:08:0E	3/8/2017 10:43:03 AM	<OK>
DC	PDC	2012 R2	SN	DC	192.168.253.200	255.255.255.0	NO	00:15:SD:FE:08:0A	2/16/2017 10:17:04 AM	<OK>
LAUNCHERGW	SRV	2012 R2	SN	LAUNCHERGW	192.168.253.1	255.255.255.0	NO	00:15:SD:FE:08:03	2/16/2017 10:17:04 AM	<OK>
LSCPROD5	SRV	2012 R2	SN	LSCPROD5	192.168.253.202	255.255.255.0	NO	00:15:SD:FE:08:06	2/16/2017 10:17:04 AM	<OK>
HSDB	SRV	2012 R2	SN	HSDB	192.168.253.201	255.255.255.0	NO	00:15:SD:FE:08:07	2/16/2017 10:17:04 AM	<OK>
GRAWEB	SRV	2008R2	SN	GRAWEB	192.168.253.230	255.255.255.0	NO	00:15:SD:FE:08:0C	2/16/2017 10:17:04 AM	<OK>
SCSIEM	SRV	2012 R2	SN	SCSIEM	192.168.253.221	255.255.255.0	NO	00:15:SD:FE:08:08	2/16/2017 10:17:04 AM	<OK>
SCWEB	SRV	2012 R2	SN	SCWEB	192.168.253.220	255.255.255.0	NO	00:15:SD:FE:08:09	2/16/2017 10:17:04 AM	<OK>
SPOIT	SRV	2012 R2	SN	SPOIT	192.168.253.232	255.255.255.0	NO	00:15:SD:FE:08:02	2/16/2017 10:17:04 AM	<OK>
SMAIL	SRV	2008R2	SN	SMAIL	192.168.253.231	255.255.255.0	NO	00:15:SD:FE:08:0B	2/21/2017 5:05:45 AM	<OK>
WB164	WS	Win8.1	SN	WB164	192.168.253.10	255.255.255.0	YES	00:15:SD:FE:08:0D	2/16/2017 10:17:04 AM	<OK>

Each column can be sorted by ascending or descending order, resized, or hidden. The display of specific columns can also be customized through the display options.

## System Name Resolution

When adding systems to a management set, you can resolve computer names using several methods. This product supports simple (NetBIOS) names, fully-qualified domain names (FQDN) and IP addresses. There are valid reasons to use each depending on network configuration.

**IMPORTANT!**

*Adding the same system by multiple names can result in duplicate licenses being utilized for the same system.*

IP addresses can be used, but they have two problems: (1) They do not necessarily provide a meaningful identification for a machine, and (2) IP addresses can be re-assigned using DHCP. These problems could result in an administrator making changes to the wrong machine.

With a DNS name, a machine can be specified in both an easily identifiable way, and a way that is insensitive to changes to the machine's IP address via DHCP as long as DHCP and dynamic DNS are linked together.

To check if a name is resolvable, try pinging the machine by name from the command line interface. If the ping resolves to the correct machine, Privileged Identity may be able to use that name to manage the machine (because it uses the same resolution mechanism as ping does).



**Note:** *Being able to ping a computer is not an indication that the computer will be manageable. It only indicates that name is responsive on the network. Management of the computer is dependent on other systems, such as SSH, RPCs, and so on that are not tested with a simple ping.*

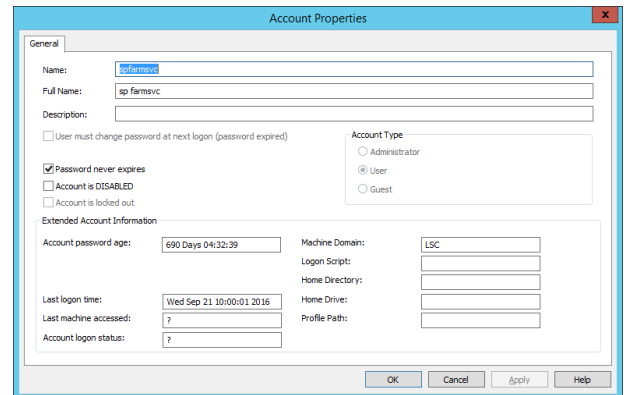
When the program does a Get Role/Version (Refresh) operation, it retrieves the NetBIOS name and IP address of each managed machine. By default, the computer is resolved by whatever name is in the System column (which can be a NetBIOS name, an IP address, or a DNS name). The resolution method can be changed by right-clicking on the computer(s) and selecting a Resolve By option. This will cause the product to use the alternate name of the computer for name resolution. In most cases, however, the computer name should be sufficient for name resolution. In addition, the other information can then be examined to make sure operations will affect the correct system(s).

## Windows Accounts View

To switch to Windows Accounts View, click **View > Windows Accounts View**.

### About Windows Account View

The Windows Accounts view shows information about Windows account on the managed Windows system(s). It is not used for other platforms. This view shows that the accounts exist and shows how old their passwords are. This is useful for reporting. To get more information about an account, right-click the account, and then select **Windows Account Properties**. When system and account information is refreshed, account information for each system contacted is retrieved and shown in the display.



### Refresh System and Account Information

Use either of the following methods to retrieve a list of accounts from the computers in the current systems list:

- Right-click in the white space and choose **Refresh System and Account Information**.
- From the menu, select **SystemsList > Refresh Information > Refresh System and Account Information**.

#### **IMPORTANT!**

*Once the Accounts View list is created and refreshed, if new systems are added, you will not be able to view accounts for the new computer(s), even if the list is refreshed. To view the accounts on the new computer(s), open Windows System view by selecting **View > Windows Systems**, then choose **SystemsList > Refresh System and Account Information**.*

### Display Account Usage Information

Follow these steps to obtain account usage information (meaning where the accounts are used on the managed systems):

1. Select one or more accounts. (Click **Shift** + click to select a range of list items and **Ctrl** + click to select/deselect individual list items.)
2. Right-click and select **Refresh system and discover local account usage**.
3. Right-click and select **Refresh List of Trusted Domains for this Account**.

This command retrieves from the domain controller all trusting domains where the domain accounts may be valid. If this step is not performed at least once, you cannot expand accounts or examine their usage. You only need to perform this step when new domains are added or removed.

Once the account and usage information is refreshed, the screen will display the new information.

## Customize Which Information is Displayed

To customize which columns are displayed, choose **View > Display Options**.

### Windows Accounts View Table Columns

- **Account:** The account name pre-fixed with either the domain name or system name as is appropriate to a domain or local systems account, respectively.
- **System:** The system where the account is located.
- **Account Type:** Identifies if the account is the built-in administrator (RID 500), a user (not RID 500), or the built-in guest account (RID 501).
- **In-Use:** This column will show **0** until account usage is refreshed, then the number will change to reflect the number of objects on the managed systems that the account is used for.
- **Password Age:** The number of days since this account's password was last changed. Be sure to refresh system and account information to see the latest data.
- **Password Status:** Status label based on password age.
  - **Threat:** Password age is 90 days old or greater.
  - **Stale:** Password age is 30 days old or greater, but less than 90 days old.
  - **OK:** Password age is less than 30 days old.
  - **[N/A]:** The account does not have a password. This is typical for built-in accounts, such as guest or ASP.NET.
- **Last Login:** Time stamp showing the last time this user was authenticated by the system. For domain users, this value is not replicated between domain controllers and thus will be dependent based on the domain controller used in the systems list.
- **Last Managed Change:** Time stamp showing the last time the password for this account was changed.

Each column can be sorted in ascending or descending order, resized, or hidden. The display of specific columns can also be customized through the display options.

Account	Account Type	In Use	Password Age	Last Login	Last Managed Change
LSC\arcweb	User	2	643 Days	9/20/2016 4:05:13 AM	[never]
LSC\erpmdefproc	User	1	720 Days	9/21/2016 2:52:58 AM	[never]
LSC\erpmweb	User	9	720 Days	9/20/2016 8:20:09 AM	[never]
LSC\scadmin	User	12	720 Days	9/21/2016 12:35:32 AM	[never]
LSC\scsvcaacct	User	19	13 Days	9/15/2016 5:01:38 AM	[never]
LSC\prcweb	User	1	48 Days	8/3/2016 4:13:33 AM	[never]
LSC\scmsvcaacct	User	2	714 Days	9/21/2016 12:32:37 AM	[never]
LSC\scmsvcaacct	User	2	714 Days	9/21/2016 1:44:31 AM	[never]
LSC\spfarmsvc	User	7	690 Days	9/21/2016 3:00:01 AM	[never]
LSC\spsvcaacct	User	6	690 Days	9/21/2016 12:47:11 AM	[never]
LSC\susan	User	1	279 Days	8/15/2016 4:05:45 AM	[never]
LSC\svcpool-2	User	6	12 Days	9/8/2016 3:43:44 AM	[never]
LSC\umpdefproc	User	1	32 Days	9/18/2016 11:53:05 PM	[never]
W8164\anicu	User	1	721 Days	10/1/2014 6:41:33 PM	[never]

## Other Actions Available From This View

- A password change job can be initiated by selecting the account and clicking **Change Passwords** in the **Actions** pane.
- Basic account maintenance can be performed on any account from this view including enabling or disabling accounts, as well as deleting the accounts. Select one or more accounts, right-click, and then select a command.

## Linux/Unix Systems View

To switch to Linux/Unix Systems View, click **View > Linux/Unix Systems View**.

The Linux/Unix Systems view shows only Linux, Unix, and related computer systems.

This information is queried directly from the target system and the information displayed is the result of that query. The name shown in the **System** column to locate the system and then perform the various queries.

The information presented in these columns is also historical data. That is, it is only relevant as of the last time a system refresh was performed (see "Checked" column). System refreshes are performed during password change jobs or system refresh operations (F5, right-click refresh).

Status is either **<OK>** or a hex error such as **0x00000035**. Errors are provided in hexadecimal format (as provided by the operating system) and should be converted to decimal for proper processing. As Privileged Identity does not perform its own networking or authentication, it is important to note that returned status messages are returned by the host operating system, and reported to you by Privileged Identity. All troubleshooting should start outside of Privileged Identity for errors listed here.

When systems are refreshed in this view, Privileged Identity contacts each system, retrieves its information, and outputs the information to the display.

The columns available for display are:

System	OS	Distribution	Resolve By	Hostname	IP Address	Subnet Mask	DHCP	MAC Address	Checked	Status
centos7	Linux	CentOS 7	SN	centos7.lac.ent	192.168.253.100	255.255.255.0	YES	00:15:9D:FE:10:00	3/8/2017 10:38:05 AM	<OK>
centosssh	Linux	CentOS 7	SN	centosssh.lac.ent	192.168.253.98	255.255.255.0	YES	00:15:9D:FE:10:01	3/8/2017 10:38:05 AM	<OK>
UBLDAP	Linux	Ubuntu 14.04	SN	ubldap	192.168.253.97	255.255.255.0	NO	00:15:9D:FE:10:01	3/8/2017 10:38:12 AM	<OK>
ubuntu14	Linux	Ubuntu 14.04	SN	ubuntu14	192.168.253.99	255.255.255.0	NO	00:15:9D:FE:10:00	3/8/2017 10:38:11 AM	<OK>

- **System:** The name entered for the system. This value can be an IP address, a simple name, a fully-qualified domain name (FQDN).
- **OS:** Shows the operating system and version.
- **Distribution:** Shows the operating system distribution information if applicable.
- **Resolve By:** Shows what mechanism is used to contact the computer on the network. Either **SN** (System Name) or **IP** (IP Address).
- **Hostname:** Discovered name of the host. This value is obtained by contacting the system.
- **IP Address:** Discovered IP address of the host (query to the system).
- **Subnet Mask:** Discovered subnet mask of the host.
- **DHCP:** Shows whether or not the IP address for this system is assigned through DHCP.
- **MAC Address:** The hardware address of this computer.
- **Checked:** The last time this computer was successfully contacted
- **Status:** The status of the last performed operation. Displays the last result message or error code for any operations on this computer.

Each column can be sorted by ascending or descending order, resized, or hidden. The display of specific columns can also be customized through the display options (**View > Display Options**).

## SSH Keys View

To switch to SSH Key View, click **View > SSH Key View**. This view displays SSH keys discovered on Unix, Linux, and similar systems regardless of the current management set.

SSH keys found here can potentially be managed in an SSH key rotation job. For SSH key rotation jobs, Privileged Identity supports OpenSSH keys of type:

- **RSAv1**: note that if rotated, v1 keys will be updated to be v2 keys.
- RSAv2
- DSA
- EC (Elliptic Curve)
- ED (Edwards Curve 25519)

However, while all key types can be discovered and cataloged, only RSAv2 and DSA key types can currently be mapped and used for login sessions when managing target SSH systems.

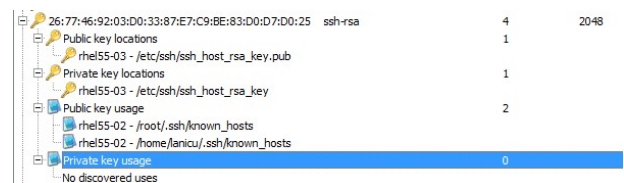
When discovery is performed and keys subsequently imported via the discovery process, if only a private key is found, Privileged Identity will also construct the corresponding public key. An public key that is constructed in this manner will be annotated with a comment of "RED-IM@HOSTNAME". The location of the private portion of the key will appear as RED IM\_Imported\_Private\_Key\_DD MM YYYY\_HH:MM:SS.MS and the same except with `_Public_` for the public portion of the key.

For each key this view also provides:

- Information about the originating host
- Account information
- A list of other systems where the key is present

## SSH Key View Display

Depicted in the screen shot below is an SSH key that is owned by a system called rhel55-03. The screen lists where the key is physically stored on the host. The **Public key usage** node and the **Private key usage** node list other systems that also know and make use of those SSH keys and where those keys are located on those other systems as depicted under the public key usage node.



Keys depicted in this view have four possible icons indicating different status information. Keys will either be green, indicating that this is a stored private key, or yellow, indicating that this is a public key with a stored private key. Each key color may be present with a shield indicating that this key is for the "root" account.

Key Identifier	Type	Length	References
36:6F:69:AD:FE:72:35:B2:9A:40:2E:24:0F:27:FA:C0	ssh-rsa	2048	1
6E:96:ED:D0:C0:5C:F0:0C:B1:A5:5C:3F:28:29:7D:FF	ecdsa-sha2-nistp...	Unknown	2
85:6C:04:85:9E:C9:7B:AB:7D:FC:9D:71:CD:7E:1A:EF	ecdsa-sha2-nistp...	Unknown	2
97:03:A8:9C:B3:52:2E:C7:45:4A:42:84:ED:91:B1:DD	ecdsa-sha2-nistp...	Unknown	1
centosssh	ssh-rsa	2048	1
F3:3B:0A:07:4E:22:C7:3F:38:4F:68:FB:C2:03:3D:FF	ecdsa-sha2-nistp...	Unknown	2

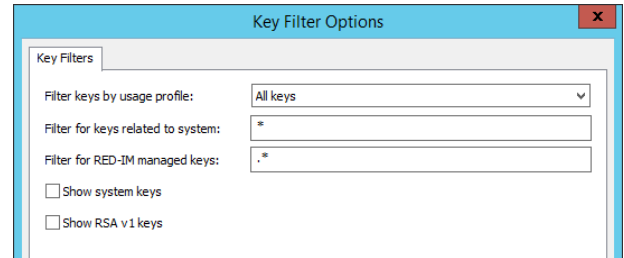
- **Green key with shield**: Private key imported providing access to the "root" account.
- **Green key without a shield**: Private key imported not providing access to the "root" account.
- **Yellow key without a shield**: Either a public key reference or a private key discovered for a system or user account that is not "root," and the private key is not stored in Privileged Identity, possibly due to requiring a password for the key or because the key is not OpenSSH compatible.
- **Yellow key with shield**: Either a public key reference or a private key discovered for the "root" account but the private key is not stored in Privileged Identity, possibly due to requiring a password for the key or because the key is not OpenSSH compatible.



## Customize SSH Key View Display Options

Initially, the SSH Key View may be intimidating because of the amount of information available, and because of how it is initially displayed. The following can help:

- To aid in sorting the display of the keys, the keys may be labeled. To label a key, right-click it and select **Label Key**.
- The view can also be filtered to show keys of a certain type. To set display filters, choose **View > Display Options**.



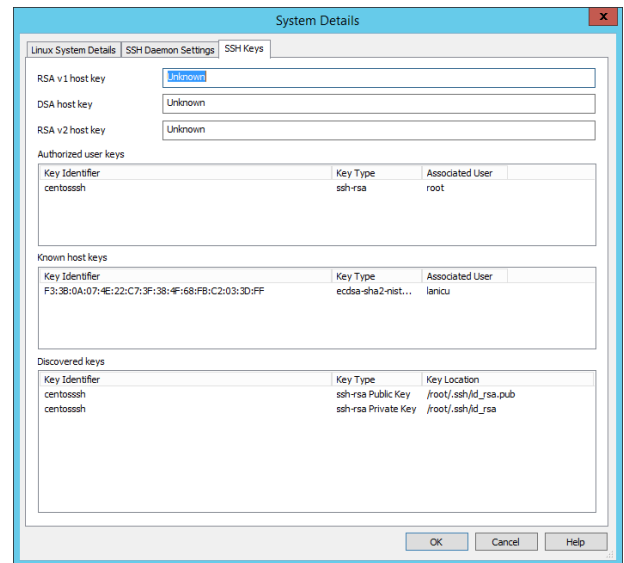
## View SSH Key Details

Each key has details that can be viewed by right-clicking the key and selecting **Key Details**.

The following is another way to view discovery keys and known host keys, along with other useful key information.

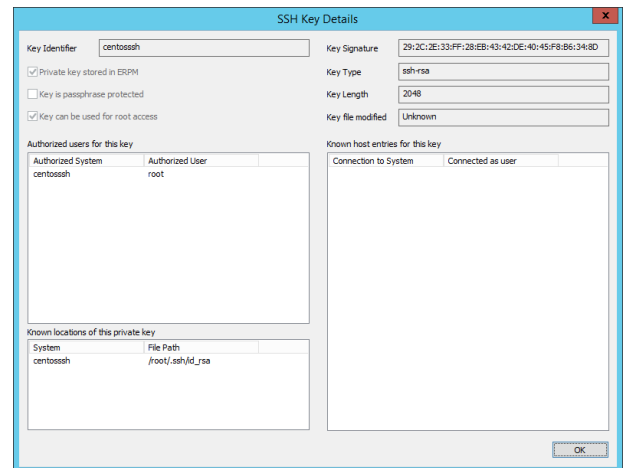
1. Perform a discovery. (See SSH Key Discovery for details.)
2. Right-click the host and select **System Details**.

When viewing system details on a Linux/Unix host, the **SSH Keys** tab includes information about the discovered keys and key information.



From the SSH Keys view, information about the particular keys may be viewed by right-clicking on the key and choosing to view Key Details:

- **Key Identifier:** The label assigned to the key. Initially, it will be set to the key's signature.
- **Private key stored in ERPM:** Indicates if the private key has been imported into Privileged Identity.
- **Key is passphrase protected:** Indicates if the SSH key has an associated passphrase or not. This is not an indication of whether or not the key has a passphrase.
- **Key can be used for root access:** Indicates if the key is associated with the root account.
- **Key signature:** The signature of the SSH key.
- **Key Type:** The type of the key, for example, RSA, DSA, and so on.





- **Key Length:** The bit-length of the key.
- **Key file modified:** If the key file's attribute is present is set in the host's file system, the last modified date value will be present.
- **Authorized users for this key:** This table indicates what account on a particular system may use this key. This may display different systems and different users depending on how the target systems are configured to use the key.
- **Known locations of this private key:** Indicates any physical path information that has been discovered regarding the physical location of the key across any number of systems. This value may not be present if the SSH key was manually imported.
- **Known host entries for this key:** Indicates the known hosts as defined in the known-hosts file for the target system. This is based on a key signature comparison to the entries in the known hosts file.

## Generate a Report From SSH Key View Data

Reports can be generated using SSH Key View data by choosing **SystemsList > Create Report from Display List**.

Alternatively, reports can be generated from the database directly. Discovered key information is found in the program data store in the following tables:

- **tbl\_KeyData\_Asymmetric\_Private:** Private keys.
- **tbl\_KeyData\_Asymmetric\_Public:** Public keys.
- **tbl\_KeyData\_SourceInfo:** Key physical location and file permissions.
- **tbl\_KeyData\_Asymmetric\_SecurityInfo:** All discovered key information (type, length, creation data).
- **tbl\_KeyData\_Asymmetric\_Correlation:** Correlation of public and private keys.



**Note:** Discovered keys may not be imported for a number of reasons, including when the key cannot be decoded (possibly due to password protection) or because the key is not OpenSSH compatible.

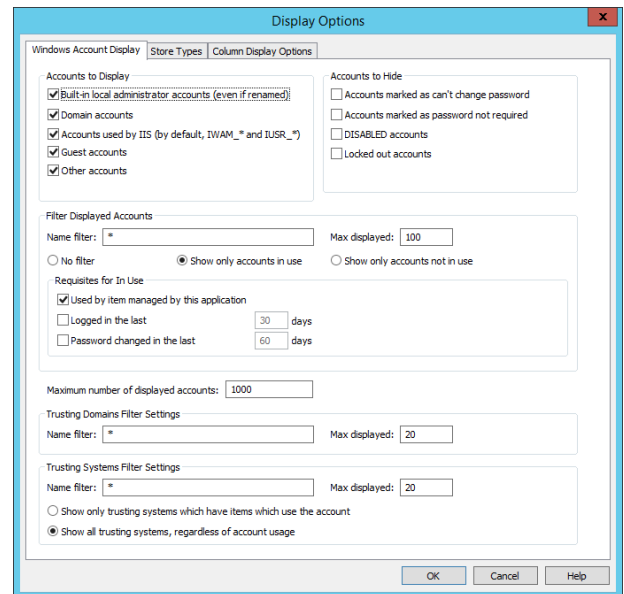
## Display Options

To modify how information is displayed in any of the five views, first open the view from the **View** menu (for example, **View > Account Store View**), and then select **Display Options** from the menu (**View > Display Options**). Depending on which view is selected, the display options dialog will display different tabs. Each tab is displayed in the following topics.

## Windows Accounts Display

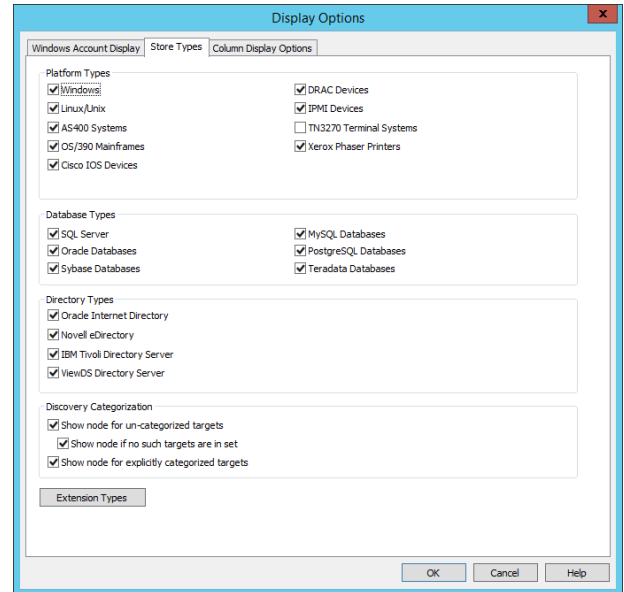
Use this tab to configure the account types that the console displays in Account Store View and Windows Accounts View.

- Select from the options listed in the **Accounts to Display** and the **Accounts to Hide** sections. **Accounts to Hide** takes precedence over **Accounts to Display** if a user account matches both criteria.
- Use the **Filter Displayed Accounts** section to further filter which accounts are displayed.
  - By default, the **No filter** option is selected, which means just about all accounts are displayed.
  - The **Show only accounts in use** or to **Show only accounts not in use** options require further definition of what it means for an account to be considered "in use." You can define whether an account is "in-use" based on if the account is managed by the application, if the account has logged in in n-number of days, if the account's password has changed in the last n-days, and/or if the account is used by services, tasks, or COM applications.
  - The **Max displayed** setting represents a firm limit on the number of accounts that will be displayed in the accounts list or Account Store View. Displaying a large number of accounts slows down the display and also requires higher memory usage.
- Use **Trusting Domains Filter Settings** to filter the domain names to include, as well as the maximum number of trusting domains to display. This setting does not stop Privileged Identity from enumerating or scanning these domains for trusting systems, only from displaying them.
- Use **Trusting Systems Filter Settings** to filter the system names to include, as well as the maximum number of trusting systems to display. This setting does not stop Privileged Identity from enumerating or scanning these domains for trusting systems, only from displaying them.



## Store Types

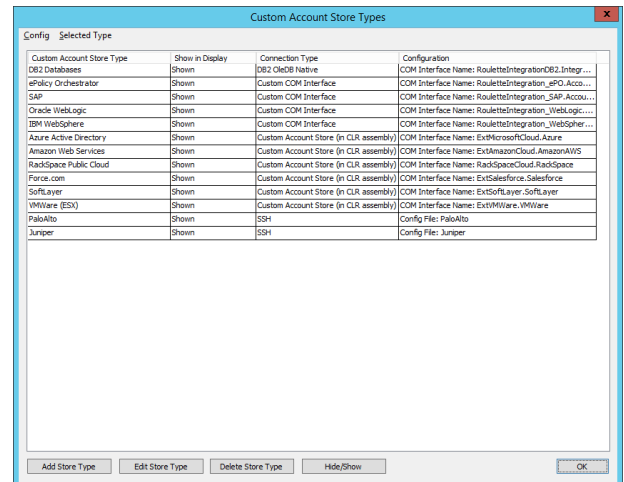
Use the **Store Types** to choose the account store objects that are visible in Account Store View.



## Custom Account Store Types Dialog

Click **Extension Types** at the bottom of the **Store Types** tab to open the "Custom Account Store Types" dialog. Use this dialog to take the following actions on custom account store types:

- **Add Store Type:** Click to open the Account Store Type Properties dialog and add a custom account store type. For details, see the Enrolling Custom Account Stores and Enrolling Cloud Service Providers topics.
- **Edit Store Type:** Select the store type to edit, then click this button to open the Account Store Type Properties dialog. For details, see the Enrolling Custom Account Stores and Enrolling Cloud Service Providers topics.
- **Delete Store Type:** Select the store type to delete, then click this button to delete. Deleting a custom account store type also removes the data for all instances of the store type and all jobs operating on targets of this account store type. Click Yes at the confirmation dialog to delete the store type; click No to cancel.
- **Hide/Show:** Select one or more store types to hide or show in Account Store View, then click **Hide/Show** to toggle between **Hidden** and **Shown** (as indicated in the **Show in Display** status column).




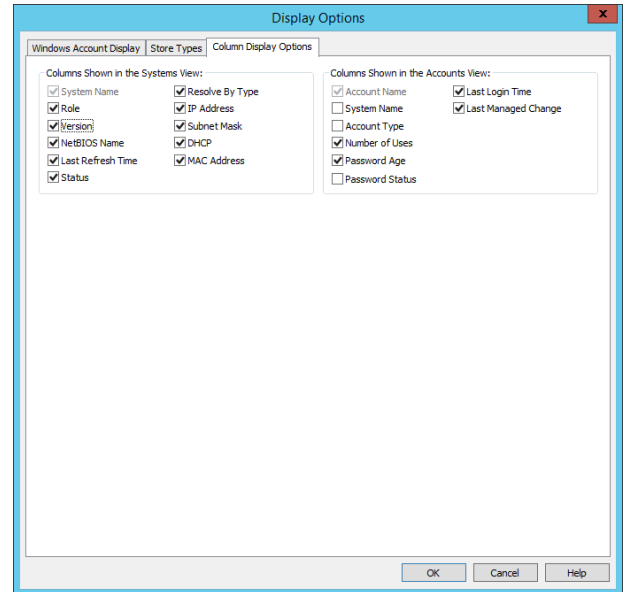
This dialog is also available by clicking **SystemsList > Custom Account Store Types**.

## Column Display Options

Use the **Column Display Options** tab to define the columns that will display in Windows Systems or Linux Systems or Windows Accounts View.

Click **OK** once the desired display options are configured.

 **Note:** These are per-user settings as opposed to global settings and affect all management sets. In the Windows registry, these settings are stored in `HKCU\Software\Lieberman\PWC`.



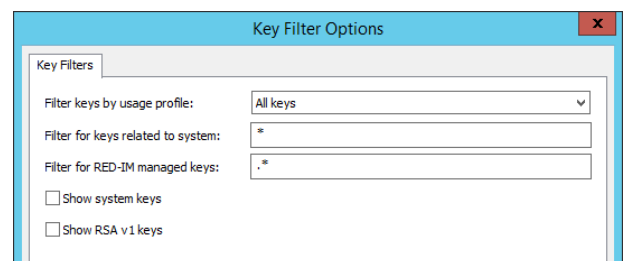
## Key Filters

Use **Key Filters** to choose which keys display in SSH Key View.

To open the **Key Filters** option, switch to the SSH Keys view (**View > SSH Key View**), then choose Display Options (**View > Display Options**).

Configure the following settings as needed:

- **Filter keys by usage profile:** Choose the keys to display from the following options
  - **Privileged Identity Managed Keys:** Show only keys which are managed by Privileged Identity.
  - **All keys:** Displays all keys.
  - **Keys with root level access:** Filters for keys that have root-level access.
  - **Keys with user level access:** Filters for keys that do not have root-level access.
  - **Keys with public/private key pair files:** This will limit the keys being displayed to those that have both a private key and public key. For instance, keys found that are only in an authorized\_keys file and nowhere else would not be shown with this option.
  - **Keys without known private key:** Filters for keys that do not have a known private key.
  - **Keys that are known weak:** Filters for low bit-length keys.



- **Filter for keys related to system:** Enter a system name to view keys stored on that system. You can also filter system names by entering the first letter or two of a name followed by an asterisk. For example, entering **ub\*** will list keys found on systems with names such as Ubuntu14 and ubldap. Similarly, entering **\*7** will list keys found on systems with names such as CentOS07 and CentOS17.
- **Filters for RED-IM managed keys:** will only find keys if they are also managed by Privileged Identity.
- **Show system keys:** Displays system or host keys stored in `/etc/ssh/ssh_host_*` files.
- **Show RSA v1 keys:** Enable to view any old OpenSSH keys. The default is to not show these keys.

## Program Options

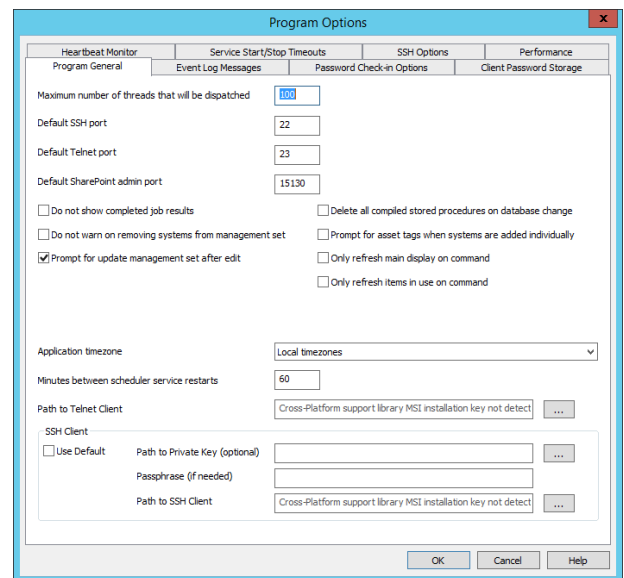
Program Options are found at **Settings > Program Options**.

The options described in the following pages discuss settings that will be applied following a password randomization such as heartbeat monitor configurations or service start/stop timeouts.

## Program General Tab

The **Program General** tab has settings for the management console and management operations.

- Maximum number of threads that will be dispatched:** Configures the multi-threading properties of the product. The higher the thread count, the more systems can be managed at once, CPU, network, and memory permitting. Privileged Identity will attempt to throttle thread dispatching when CPU or memory usage is too high (over 95%). Higher thread counts will also increase network traffic. The default thread count is 100 threads which works well for most system and database configurations. In Windows Server 2012 R2 with a dedicated database instance, 200-300 threads is generally fine for most networks. In Windows Server 2008 R2, it is not recommended to exceed 200 threads as the SMB stack is not as robust as that on Server 2012 R2. Threading problems show themselves as missed operations or database timeouts due to lack of resources.
- Default SSH port:** Configures the default port to be used when "OpenSSH Console", context menu option, is selected for a Linux/Unix system. Default is 22.
- Default Telnet port:** Configures the default port to be used when "Open Telnet Console", context menu option, is selected for a Linux/Unix system. Default is 23.
- Default SharePoint admin port:** Configures the port that will be used to discover and manage Microsoft SharePoint instances. Only one configuration can be set per Privileged Identity installation.
- Do not show completed results:** When enabled, stops the job results dialog from displaying job results following an interactive job run. Default is not enabled.
- Do not warn on removing systems from management set:** when enabled will not alert the user that they are about to remove a system from the management set.
- Prompt for update management set after edit:** When enabled and management set properties are edited, you will be prompted to update the management set. Default is enabled.
- Delete all compiled stored procedures on database change:** When enabled and the database is upgraded or the management console is re-connected to a database, all existing compiled stored procedures will be deleted (and subsequently re-created). This helps ensure the latest stored procedures are used and other programmatically generated stored procedures which may no longer be needed are removed. This helps ensure performance over time. Default is not enabled.
- Prompt for asset tags when systems are added individually:** When enabled, adding any system to any node will cause a prompt for an asset tag to appear. This asset tag is visible in system properties (both GUI where applicable and web service/PowerShell), as well as in the web application systems view.
- Only refresh main display on command:** When enabled, the management console will only refresh the main display on command (filters updated or re-applied) or when switching between management sets. If the option is disabled, the main display will be refreshed after changing program options, viewing the jobs dialog, or viewing other dialogs that could affect the main display. Default is enabled.



- **Only refresh items in use on command:** When enabled, the main display will not update the items in use (service accounts and system expansion) when the user clicks the refresh button (magnifying glass) or switches between management sets. Default is not enabled.
- **Do not warn on removing systems from management set:** When not enabled and you manually remove a system from a management set, you will receive a warning asking you to verify the removal and to also check the list of static inclusions for the management set. Default is not enabled.
- **Application time zone:** set to define the scheduling timezone for all scheduled jobs. When set to Local timezones (default) the scheduling behavior is based on the local timezone of the component running the job. This behavior is the same as seen in all versions of Privileged Identity from version 5.5.2.2 and earlier.

This has a potential side effect of running a job twice or more based on the time relationship between the console scheduling the job and the component running the job. Consider a console in timezone -6 hours GMT. In this case a job scheduled for 12PM would be seen as 10AM in the -8 GMT timezone. The job would be seen as past due and would run immediately, but then rescheduled for 10AM (per the job settings) and would run again two hours later. After that, the job would run at 12PM (-8 GMT) per the job settings. If another component in another timezone runs the job on the next interval, the job could run twice again.

To avoid this problem, set a program timezone. All jobs will be calculated based on a timezone value. This means that when scheduling the job, you must be aware of the application timezone. Thus if the intent is for a zone processor to run a job at 12PM when the component is -8GMT and the console is -6GMT, the job should be scheduled for 2PM.

- **Minutes between scheduler service restarts:** Configures the interval for Deferred and Zone Processor restarts (job scheduling services). The restart should be configured to a frequency higher than Kerberos ticket (ST and TGT) expirations which defaults to 600 minutes or 10 hours. The default service restart frequency is 60 minutes. Setting a value that is too high can lead to errors during Windows Service management such as error 1305: Revision Level Unknown. This feature was added to handle Kerberos ticket refresh errors surrounding SCM access on Windows Server 2012 R2. Service restarts will be logged in the host's system log, the product's PWCJobLog.txt file, and will also trigger event sink numbers 5000: EVENT\_ID\_SCHEDULER\_STARTED and 5007: EVENT\_ID\_SCHEDULER\_STOPPED (if configured).
- **Path to Telnet Client:** You can right-click on a system and select Open Telnet Console after which a Telnet program will open. This defines the path to the Telnet client. You may use the built-in Telnet client at %programfiles(x86)%\Lieberman\Cross Platform Support Library\TelnetConsole.exe or specify the path to a local PuTTY executable.
- **SSH Client:** this is not used for SSH management or discovery. This setting is used for right-clicking on a target and electing to connect with SSH console.
  - **Use default:** When enabled, the default SSH/Telnet client found in the default installation path will be used, which is OpenSSH.
  - **Path to Private Key:** (Optional) If an SSH key will be used to connect to the target system (limited to RSAv2 and DSA keys for the default SSH client), specify the path to the SSH key.
  - **Passphrase (if needed):** (Optional) If a private key is specified that requires a passphrase, place the passphrase here. If the key requires a passphrase and the passphrase is not specified, the user will be prompted for the SSH key's passphrase.
  - **Path to SSH Client:** You can right-click on a system and select OpenSSH Console after which an SSH program will open. This defines the path to the SSH client when the program is not configured for Use Default. You may use the built-in SSH client at %programfiles(x86)%\Lieberman\Cross Platform Support Library\SSHConsole.exe or specify the path to a local PuTTY executable.

The Path to SSH Client field can now be populated with path to the application and arguments. In addition to arguments supplied by the user, other arguments can also be supplied by the console application in the form of a directive character prefixed by a percent sign. If the file explorer is used to find an application, a %h will be appended to the end which specifies the target being connected to. The list of directives available and their meanings are

- **%h** - Host name
- **%u** - User name

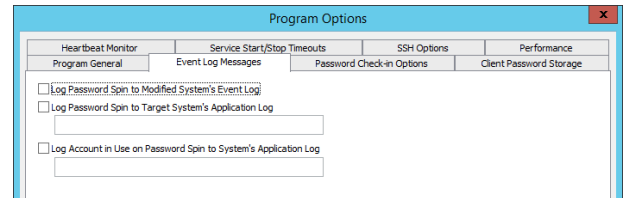
- **%p** - Port number
- **%k** - Private key file path
- **%P** - Key passphrase
- **%j** - Java bin install directory



## Event Log Messages

The deferred processing service can take multiple logging actions when performing password spins for passwords that are recovered through the web interface.

These options only apply to jobs created by instances of the web application in conjunction with the auto-spin password on recovery option. The logging options allows input of an event log message in a Windows computer's application event log. The message has an Event ID of 17 and a source of Privileged Identity. One or more of the options can be enabled to track password spins being performed by the deferred processor. Options to log to the target system's event log or a specific system's event log as well as logging a message if the deferred processor service detects that the account is in use on the system when the password change happens are available.



Note that these event log messages are not generated if the password spin is prevented or has not happened yet. It is possible to log an event at the actual time of the recovery/check-in using the web application settings for event logging.

- No options are enabled by default.
- **Log Password Spin to Modified System's Event Log:** When enabled, places the event in the target system's application event log. This option is for valid when the target system is a Windows system.
- **Log Password Spin to Target System's Application Log:** When enabled, places the event in the specified system's application log.
- **Log Account in Use on Password Spin to System's Application Log:** If the solution is configured to check for account in use (password change job option) prior to password rotation, this log message will be triggered and will write the event to the specified system's application log.

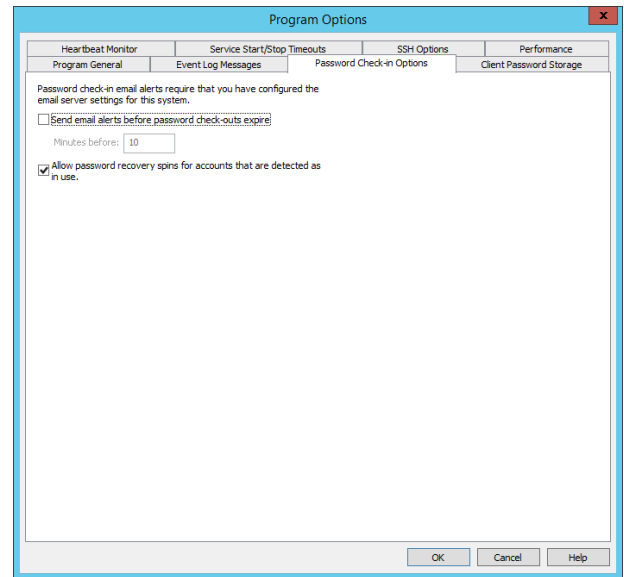
Typically, these items are left unconfigured in lieu of using event the following event sinks:

- 2000-2071 for password changes (platform specific).
- 3024 for passwords edited via the web application.
- 2072 for account in use failures.

## Password Check-in Options

These options control the password optional check-in and spin features of the deferred processor that relate to the web application.

- Send email alerts before password check-outs expire:** With this option enabled, the deferred processor will check for password checkout expiration in the web application and send an email notification to the Windows user that has checked out the password informing them that their password checkout is about to expire. The email address destination for the user is pulled from Active directory. If the user who has checked out the password is not a Windows user, the web application is not running in an active directory domain, or if the user has no registered email address in AD, then this warning will be ignored. Email settings must be configured for this to work.
- Allows password recovery spins for accounts that are detected as in use:** The application will check for existing logon sessions using the account, and if one is found, the password change operation will fail normally and be set to retry according to the defined retry policy. This check happens at the time when the password is spun by the deferred processing service, not when the password is actually checked in to the user interface.

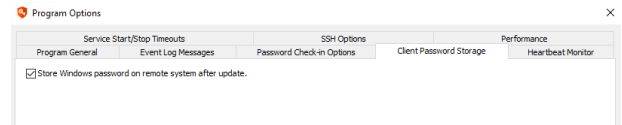


## Client Password Storage

This option allows enabling of client password storage settings for Windows systems only. This is generally used in conjunction with the Legacy Software Development Kit (SDK) - deprecated - for local or programmatic offline password retrieval.

This option should only be enabled if it desired to store credentials in the registry of the system where the password is being changed. This has no effect for distributed credentials such as service accounts.

If this option is enabled, passwords for accounts that exist on remote systems will be pushed to the registry of the remote system in an encrypted format that can be decrypted using the remote agent SDK. This option is useful if recovery of the password on the remote system while it is offline and doesn't have access to the web application recovery system is required.



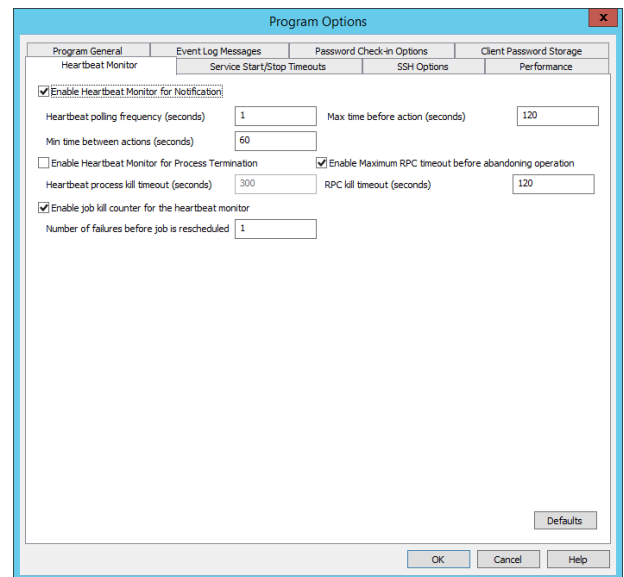
Privileged Identity can also cache credentials in a Java Client installed on desired target systems. The Java client can be deployed via a right-click option on the target system and functions on both Windows and non-Windows operating systems. This is done with **Local Cache for Java Client Update** propagation. Privileged Identity can also use the Java client SDK.

Encryption for passwords when using the Java client is provided via AES encryption. Communication with the Java client is protected via Rc4.

## Heartbeat Monitor

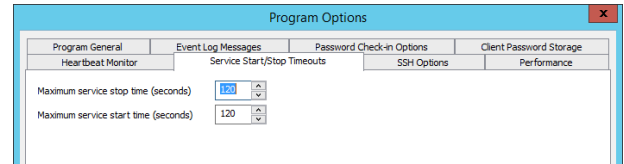
The Heartbeat Monitor monitors the processes in Privileged Identity. Specifically, when jobs are run interactively, this means the process representing the management console (`roulette.exe`) and when jobs are running on a scheduled basis this means the job processor executable (`rouletteproc.exe`). The Heartbeat Monitor consists of two action types: Normal and Critical. The normal action is simply a logging message. If the job is being run interactively (immediately) via the console, this goes to the main program log. If the job is running via schedule, the Heartbeat Monitor logging messages will be logged to the job log's text file. The critical action is to kill the process in its entirety.

- **Enable Heartbeat Monitor for Notification:** When enabled will log heartbeat messages (e.g. Long running operation detected) to the job log.
  - **Heartbeat polling frequency (seconds):** How often the heartbeat monitor will check to see if there is a heartbeat.
  - **Max time before action (seconds):** The amount of time without a heartbeat before the heartbeat monitor will take the default action.
  - **Min time between actions (seconds):** The amount of time that must pass before the heartbeat monitor will take the default action again for the same process.
- **Enable Heartbeat Monitor for Process Termination:** When enabled, will terminate the running process (typically the scheduler process) when the Heartbeat Process kill timeout value is reached. Default for the feature is not enabled.
  - **Heartbeat Process kill timeout:** The amount of time for a non-responsive process before the host process (`Roulette.exe` - the console, or `RouletteProc.exe` - the job processing process) is killed. Default for the kill timeout is 300 seconds.
- **Enable Maximum RPC timeout before abandoning operation:** Thread specific for Windows systems. The amount of time to wait for a non-responsive thread to simply abandon RPC connections (Windows). Default is 120 seconds.
- **Enable job kill counter for the heartbeat monitor:** When enabled, will increment each time the job is automatically unlocked after a crash. If the kill counter for a job exceeds whatever threshold is specified in this setting, then the job will be failed with no retry and rescheduled instead of marked as partially complete and put back into the queue. Default is 1.
- **Defaults:** To quickly return all settings to their default value, click the **Defaults** button. This will reset every component attached to the Privileged Identity instance to the default heartbeat monitor values.



## Service Start/Stop Timeouts

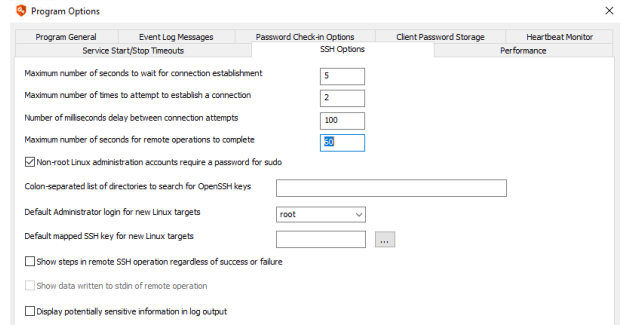
The **Service Start/Stop Timeouts** tab is used to define the wait times for Windows services to start or stop when performing password change jobs with propagation against Windows services. The default wait duration is 120 seconds. This means a service must start or stop within 120 seconds or that propagation step will be labeled as job failed. If there are scenarios where services may take three or four minutes or longer to shutdown or startup, adjust these values to account for those services.



## SSH Options

The **SSH Options** tab is used to configure certain aspects of SSH-based connectivity and timeouts.

- Maximum number of seconds to wait for connection establishment:** If a connection attempt does not succeed within this time, the connection attempt is considered a failure. The default is 5 seconds.
- Maximum number of times to attempt to establish a connection:** The number of times a connection attempt is made if there is a failure. The default is 2 attempts.
- Number of milliseconds delay between connection attempts:** The number of milliseconds to wait between each successive connection attempt. The default is 100ms.
- Maximum number of seconds for remote operations to complete:** If a job does not complete successfully within this time, the job is considered a failure. The default is 60 seconds.
- Non-root Linux administration accounts require a password for sudo:** When enabled, if a non-root account performs a refresh against a target Linux/Unix system, it requires the use of a password to run sudo.
- Colon-separated list of directories to search for OpenSSH keys:** Choose which directories to include when discovering SSH keys on Unix, Linux, and similar systems. If left blank, all directories are searched.
- Default Administrator login for new Linux targets:** Populates the list with alternate administrators. This list is populated with any administrators found in the database for existing machines, limited to the 20 most commonly found.
- Default mapped SSH key for new Linux targets:** Choose a default SSH key to use when creating a new Linux target.



Program Options

Program General | Event Log Messages | Password Check-in Options | Client Password Storage | Heartbeat Monitor

Service Start/Stop Timeouts | SSH Options | Performance

Maximum number of seconds to wait for connection establishment: 5

Maximum number of times to attempt to establish a connection: 2

Number of milliseconds delay between connection attempts: 100

Maximum number of seconds for remote operations to complete: 60

Non-root Linux administration accounts require a password for sudo

Colon-separated list of directories to search for OpenSSH keys:

Default Administrator login for new Linux targets: root

Default mapped SSH key for new Linux targets:  ...

Show steps in remote SSH operation regardless of success or failure

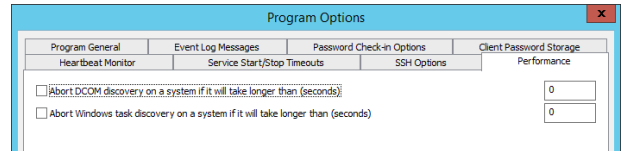
Show data written to stdin of remote operation

Display potentially sensitive information in log output

## Performance

The **Performance** tab is used to configure aspects of propagation discovery and management.

- Abort DCOM discovery on a system if it will take longer than (seconds):** This setting affects the propagation type DCOM Object RunAs identities. The timeout case uses a performance check to determine how long it will take to enumerate the DCOM applications on a target system and if that timeout will be exceeded. If the timeout would be exceeded, this operation will be skipped. Healthy, well connected systems typically take less than 20 seconds to refresh the entire DCOM catalog, per system. If this value is not defined (default) an unhealthy system can cause a discovery or management job to pend indefinitely as each call may take an undisclosed amount of time. While a long running operation may be noted by the heartbeat monitor, depending on how quickly each RPC call takes to respond, the job may not terminate as their does in fact appear to be active operations.
- Abort Windows task discovery on a system if it will take longer than (seconds):** This setting affects the propagation type Windows Scheduled tasks. The timeout case uses a performance check to determine how long it will take to enumerate the scheduled tasks on a target system and if that timeout will be exceeded. If the timeout would be exceeded, this operation will be skipped. If this value is not defined (default) an unhealthy system can cause a discovery or management job to pend indefinitely as each call may take an undisclosed amount of time. While a long running operation may be noted by the heartbeat monitor, depending on how quickly each RPC call takes to respond, the job may not terminate as their does in fact appear to be active operations.



## Control Access to the Management Console

Privileged Identity has the ability to control which users have the rights to launch the management console. Following installation, any user who is an administrator of the system where the console is installed who also has rights to the SQL database, will have the ability to launch the application.

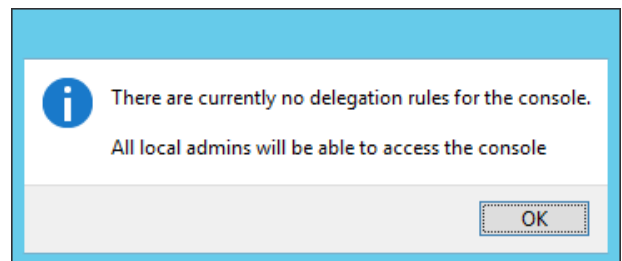
### Create a White List of Allowed Console Users

To require administrators to be on a white list of allowed console users, navigate to **Delegation > Delegate Console Access**. If no delegations are present, the following dialog will display:

Users added to this console delegations list must still meet the prerequisites in order to launch the console:

- Local administrator
- Sufficient database access

The console delegation simply adds an extra filtering and authentication model. When users are added to this list, their permissions within the console can also be restricted.



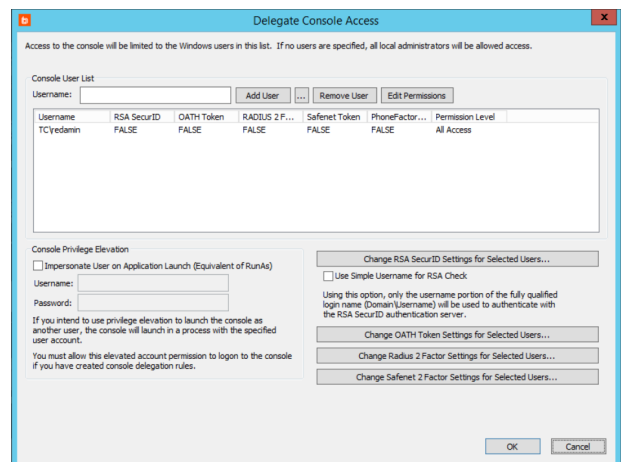
**i** For more information on these delegations, please see the next section, "[Delegate Console Access](#)" on page 45.

From this page either type in the names of users who can launch the tool by clicking the **Add** button or by browsing the domain for users and selecting the accounts to allow access to. The proper format of user names will be **DomainName\UserName**.

Once a single user is added to the list, any other administrators not included in the list, will be unable to open the console.

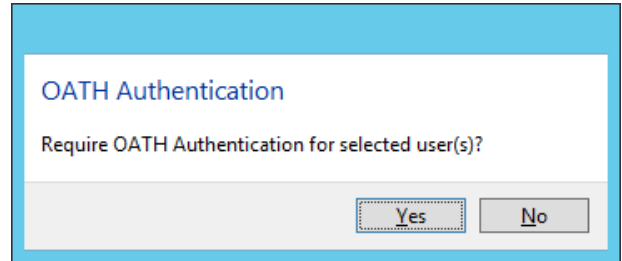
When adding users to the delegated users list, it is possible to also require the users to provide two factor authentication via:

- **RSA**
- **OATH**
- **SafeNet**
- **RADIUS 2 Factor**

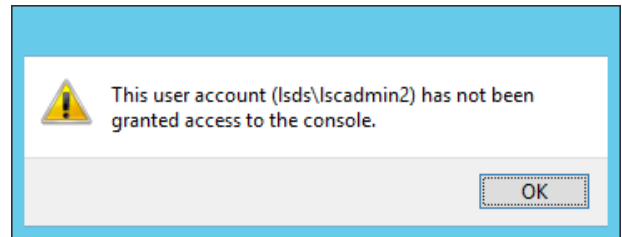




To change the settings for a particular user, select the user, and then click the **appropriate** button to **Change XXX Requirement Settings for Selected Users**. A prompt will appear asking to require the particular two factor authentication for the user. Simply choose **Yes** or **No**.



If an all access model is later desired, where all administrative users are able to launch the tool, simply clear all entries in the above dialog. If a user tries to access the console that has not been granted the rights to do so, the management console will not launch and will display the following warning:



## Additional Security Considerations

These delegations are stored in the database in a table called **tbl\_DelegatedConsoleUser**. If using these delegations, access to this table should be secured. Moreover, in the event of complete lockout of the tool, a DBA for the database can clear all the entries from this table and Privileged Identity will default to allowing all administrators of the system to launch the console application.

Further, as changes are made to these delegations from the management console, those events will be logged to the system's application logs as event id 22 when users are added or event id 23 when users are removed with a source of **Privileged Identity**.

## Launch the Console With Elevated Permissions

Privileged Identity provides a facility to always launch the console with elevated privileges. To configure this option, enable the **Run with elevated privileges option and supply a proper administrative account**. This permits an otherwise low powered user account to launch the console and perform management actions. **These credentials cannot be managed automatically by Privileged Identity**. Proper name input is in the form of either **User\_Account\_Name** or **Domain\_Name\User\_Account\_Name**.



**Note:** This option does not work when UAC is enabled.



### IMPORTANT!

*If using the option to run the console as an elevated user is enabled, do not perform other delegations on this dialog as they will not work. The console will never prompt for the other users and if the elevated user account is not on the list, the console will also fail to open due to delegation constraints.*

## Delegate Console Access

To open the management console, users must meet the following prerequisites:

- Local administrator
- Sufficient database access

The console delegation adds an extra filtering and authentication model. When users are added to this list, their permissions within the console can also be restricted.

By default, users added to the console delegation list will have **All Access** to the console, which means they can see and do anything in the console. The delegation permissions allows granting or denying access to certain parts of the management console.

To lock out administrators from portions of the console, first add them to the console delegations list as described in "[Create a White List of Allowed Console Users](#)" on page 44, and then select the user and click **Edit Permissions**.

- **All Access:** Grants all access to the console, default. If this bit is set, all other deny permissions are ignored.
- **Edit Database Settings:** Change primary program database configuration.
- **Edit Encryption Settings:** Change program encryption settings.
- **Edit Web Application Settings:** Change default and specific web site configuration settings.
- **Edit Web Delegation Settings:** Change delegation settings for web site and web service access from the management console.
- **Edit Console Delegation Settings:** Change console delegation list and access permissions.
- **Edit Logging Settings:** Change system logging settings.
- **View Console Logs:** View program logs via the console.
- **Edit Deferred Processor Settings:** Change, install, or remove the deferred processor.
- **Edit Job Retry Policy:** Change the failed job retry policy settings.
- **Edit Restricted Systems List:** Add or remove systems to the restricted systems list.
- **Edit Compliance Reporting Database Settings:** Change the compliance reporting database configuration.
- **Gather Compliance Reporting Database Snapshot:** Interactively gather or schedule a gather of information for the compliance database via the console.
- **Run Compliance Reports:** Run compliance reports from an existing capture.
- **Edit Client Agent Settings:** Edit or push out the SDK client agent settings.
- **Edit Event Sink Settings:** Edit event sink settings.
- **Edit Email Server Settings:** Edit email server settings.
- **Edit Program Options:** Change program options.
- **Edit Discovery And Propagation Defaults:** Edit discovery and propagation defaults. Affects what information is collected during an account usage discovery job and what is included in password change jobs by default.
- **Edit Alternate Administrator Accounts:** Add, remove, or edit entries from the alternate administrators list.
- **View Password Store:** Gain access to all credentials in the password store.
- **Edit Authentication Servers:** Edit the authentication servers list. Affects web site logins.
- **Edit 2 Factor Authentication Settings:** Edit multi-factor authentication settings. Affects console and web site logins.
- **Edit User Impersonation Settings:** Edit user impersonation settings. Used for integrations such as McAfee EPO.
- **Edit User Lockout Status:** Unlock users who are locked out of the web site.
- **Import Passwords:** Add or overwrite passwords into the password store
- **Edit Password Compartmentalization Settings:** Change password compartmentalization access, potentially granting a single user more access to a password segment or removing segment permissions.
- **Edit Shared Credential Lists:** Add, edit, or remove shared credential lists.
- **Create Remote Sessions:** Right-click option on systems to establish RDP or SSH sessions.
- **Edit License Info:** Change the currently installed license. Could drastically affect whole system functionality.

- **Create Management Sets:** Add new management sets.
- **Edit Management Sets:** Edit the properties of existing management sets. Improper editing could grant users access to systems and accounts they should not be able to access in the web site or possibly manage credentials that should not be managed.
- **View Management Sets:** Open the list of all management sets so the console user can select and change the currently active management set.
- **Delete Management Sets:** Delete management sets.
- **Start And Stop Deferred Processors:** Stop and start deferred processors. This does not stop the user from using other Windows mechanisms to control the deferred processors.
- **Create Password Change Jobs:** Create new password change jobs and set all settings.
- **Edit Password Change Jobs:** Edit existing password change jobs.
- **Delete Password Change Jobs:** Delete existing password change jobs.
- **Run Password Change Jobs:** Run existing password change jobs.
- **Create Refresh Jobs:** Create new refresh/discovery jobs.
- **Edit Refresh Jobs:** Edit existing refresh discovery jobs.
- **Delete Refresh Jobs:** Delete existing refresh discovery jobs.
- **Run Refresh Jobs:** Run existing refresh discovery jobs.
- **Create Elevation Jobs:** Create new account elevation jobs.
- **Edit Elevation Jobs:** Edit existing account elevation jobs.
- **Delete Elevation Jobs:** Delete existing account elevation jobs.
- **Run Elevation Jobs:** Run existing account elevation jobs.
- **Run Management Set Update Jobs:** Run existing management set update jobs.
- **Edit Management Set Jobs:** Edit existing management set update jobs.
- **Delete Management Set Jobs:** Delete existing management set update jobs.
- **Create Stored Password Test Jobs:** Create new stored password test jobs.
- **Edit Stored Password Test Jobs:** Edit existing stored password test jobs.
- **Delete Stored Password Test Jobs:** Delete existing stored password test jobs.
- **Run Stored Password Test Jobs:** Run existing stored password test jobs.
- **Create Administrator Activity Report Jobs:** Create new admin activity reports.
- **Edit Administrator Activity Report Jobs:** Edit existing admin activity reports.
- **Delete Administrator Activity Report Jobs:** Delete existing admin activity reports.
- **Run Administrator Activity Report Jobs:** Run existing admin activity reports.
- **Edit Enrolled Certificates:** Edit existing certificates used for identity authentication.
- **Edit Domain Discovery Restrictions:** Edit domain discovery OU restrictions.
- **Edit Custom Account Store Types:** Edit, add, delete custom account stores.
- **Create Windows Accounts:** Create new accounts on Windows systems.
- **Delete Windows Accounts:** Delete discovered accounts on Windows systems.
- **Edit Remote Applications:** Edit, add, or delete remote launch applications.
- **Edit Gateway Servers:** Edit, add, or delete gateway/bastion/jump servers for remote launch applications.
- **Execute IPMI Power Commands:** Run IPMI power commands against IPMI devices.
- **Edit BMC Remedy Configuration:** Edit the BMC Remedy integrations settings.
- **Edit Microsoft System Center Configuration:** Edit the Microsoft System Center Service Manager integrations settings.

- **Edit HP Service Center Configuration:** Edit the HP Service Manager integrations settings.
- **Edit Service Now Configuration:** Edit the ServiceNow! integrations settings.
- **Edit OTRS Configuration:** Edit the OTRS integrations settings.
- **Edit CA Service Desk Manager Configuration:** Edit the CA Service Desk integrations settings.
- **Edit JIRA Configuration:** Edit the JIRA integrations settings.
- **Edit Custom CLR Configuration:** Edit the Custom CLR configuration settings.
- **Edit Clustered Resource Types:** Edit the Clustered Resources dialog.
- **Edit Custom Communication Types:** Edit custom communications configurations.
- **Edit System And Account Store Settings:** Edit system and account store settings.
- **Edit Token Configuration:** Edit settings for OATH authentication tokens.
- **Open Job Monitor Dialog:** View the Jobs dialog.
- **Open System Information View:** Open the Windows systems view.
- **Open Account Information View:** Open the Windows accounts view.
- **Open Account Store Information View:** Open the account store view.
- **Edit Orphaned Systems List:** Edit the orphaned systems list.
- **View Target And Account Details:** View the properties of systems and accounts.
- **Ping System** - ping a system.
- **Enable Windows Account:** Enable a disabled Windows account.
- **Edit Java Agent Configuration:** Edit Java agent configuration.
- **View SSH Key Information:** View SSH key information.
- **Create SSH Key Change Job:** Create a new SSH Key rotation job.

To define a delegation permission, select the user from the console delegation list, and then click **Edit Permissions** at the top of the dialog. To modify their permissions, select the desired entries then choose to **Allow Selected** or **Deny Selected**. If the **All Access** permission is allowed, that will override any other permissions that may not otherwise be allowed or are explicitly denied.

Once settings are changed for a specific user, that user's session will be affected by removing or disabling buttons and menu items.

## Management Set Introduction

Management sets are used to define lists of systems and devices. These lists can be used for the following functions:

- **Job boundaries:** All jobs can target an entire list of systems as well as individual systems.
- **Password propagation scope:** When performing a password change job where the account is used by multiple machines, propagation scope defines those secondary machine's that could be using the account and must be updated after the password for the account is changed.
- **Delegations:** Identities can be granted access to an entire list of systems and devices for the sake of password retrieval, privileged sessions, account elevation, etc.
- **Management zones:** Distributed networks may leverage zone processors to distribute the work load or manage disjointed networks. Management sets are the foundation of configuration for these management zones.

You may create as many management sets as are needed to suit your management needs. You may find it is necessary to have the same system represented in multiple management sets. Adding the same machine [name] to multiple management sets will NOT consume additional licenses.

Typically, management sets are first deployed by platform, such as Windows or Linux. Management sets may represent specific application farms such as SharePoint or clustered servers. Management sets may also represent geographic dispersion (see management zones above). Management sets may in fact be combinations of one or more criteria.

One important takeaway regarding information in Privileged Identity is that management sets represent only a loose association of systems. That is to say, a system may be moved from one management set to another without loss of password or account data or the need to reproduce any of that data. So if you start with one deployment model and later change to another or needs simply evolve, just add or move (or remove) the systems as necessary to any of your management sets.

Management sets may be created in any of the following ways:

- Click **Management Sets** from the **Actions** pane in the management console .
- Use PowerShell with an **All Access** account and the **New-LSManagementSet** operation.
- Use the **ManagementSet** interface as POST from the REST-based web service.
- Use the **ManagementSetOps\_CreateManagementSet** method from the SOAP-based web service.

## Overview of Management Sets in the Management Console

You can access the **Manage Management Sets** dialog from the **Actions** pane by clicking **Management Sets**.

Select the **Import/Export** menu item to import a list of management sets from or to an XML file. This list contains the names of the management sets and all of their configured settings.

Click the **Add** or **Delete** buttons to add or delete a management set. You cannot delete the management set you are currently viewing.

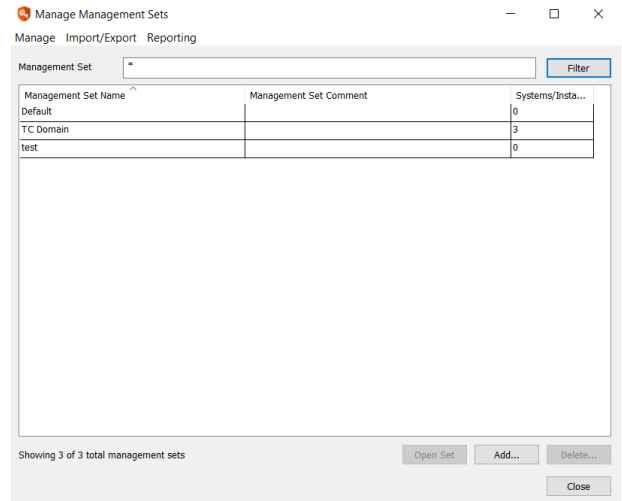
**!** **IMPORTANT!**

*Clicking the **Add** button, though tempting, causes Privileged Identity to contact the host systems local Active Directory, enumerate every single organization unit (OU) and create a management set for it and populate each management set with the contents of the OU and all of its child OU system objects.*

The **Reporting > Generate Report on Config** menu option generates a report on the management sets in the list. The output is the management set's name, comment, and instance information (number of systems/devices in the management set).

To view an existing management set, either double-click the management set or highlight it, and then click **Open Set**.

To find a system in a management set, click on any single management set, then click **CTRL+F** to open the **Search for System** dialog. Type in the name or partial name (use DOS teststyle wild cards if needed, e.g. server\*), and then click **OK**. All management sets containing the system(s) will be highlighted.



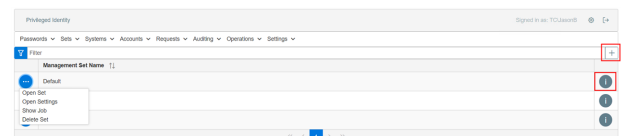
## Overview of Management Sets in the Web Application

You can access the **Management Sets** page by selecting **Sets > Management Sets** from the top menu.

To add a new management set, click the **+** button, and then give the management set a name.

To add systems to a management set, modify settings for the set, view jobs for a management set, or to delete a management set, click the **ellipsis** button for the set, and then select the applicable menu option. Options are:

- **Open Set**
- **Open Settings**
- **Show Job**
- **Delete Set.**



To view quick details for a management set, such as its name and comment, and the number of account stores in the set, click the **i** button for the management set.

**i** For more information on viewing and configuring management sets from the web application, please see "[Configure Management Sets in the Web Application](#)" on page 451.

# Scheduled Jobs and Deferred Processing Introduction

In Privileged Identity, jobs can be run immediately or can be scheduled to occur at a later point in time. Any job scheduled to be run at a later time will be run by a Deferred Processor or a Zone Processor. A zone processor is a way of indicating a deferred processor which handles a discrete list of systems.

When jobs are run interactively from the management console, the job is run by the user that is currently logged in. For jobs targeting Windows systems or Active Directory, this typically means it will be this user's credentials that are used for any of the operations within that job. For other platforms, other credentials may be used, such as SSH keys.

When jobs are scheduled, these jobs run in the context of the deferred processing service account. Even a job configured from the web application, web service, or PowerShell set to run immediately will be scheduled to run now, rather than being run interactively.

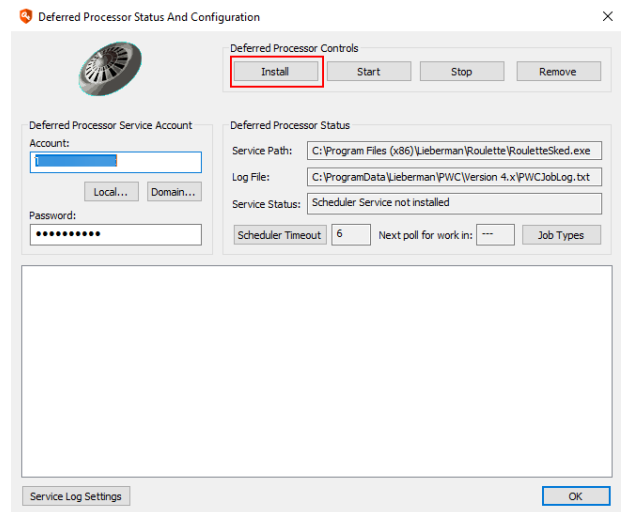
## Install Deferred & Zone Processors

Privileged Identity makes use of two distinct types of service for processing jobs:

- **Default Deferred Processing Service (or Default Deferred Processor):** Refers to the service installed with the management console and can be configured from the **Deferred Processing** dialog. This service has no concept of zones. That means it will process any and all configured jobs that become available for any system any where.
- **Zone Processor:** Refers to a deferred processing service that is configured to target a specific management set. That means it will only process jobs for systems which appear in any management set the zone processor is configured to manage.

The default deferred processor may be installed at installation time or may be installed post program installation. To install the deferred processor after installation:

1. From the **Actions** menu, click **Jobs**.
2. In the **Stored Jobs** dialog, click **Deferred Processor**.
3. Specify the name of the account and its password, and then click **Install**. This installs the service, grants **Logon As a Service** to the service account, and starts the service.

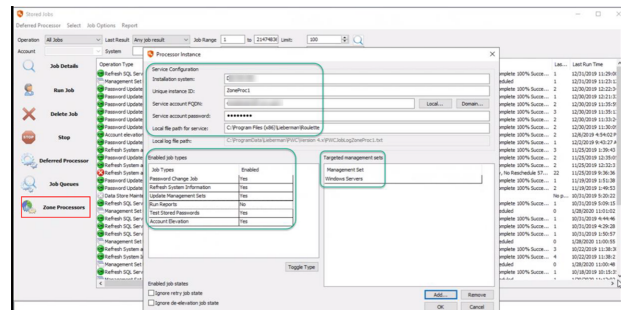


Zone Processors may be installed at any time after Privileged Identity is installed and are not dependent on the default deferred processor. To install a zone processor:

1. From the **Actions** menu, click **Jobs**.

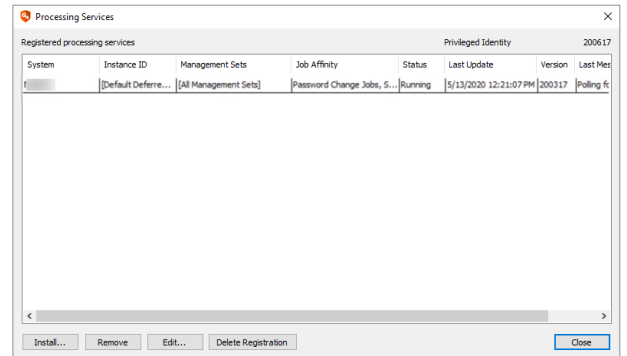


2. In the **Stored Jobs** dialog, click **Zone Processors**.
3. Click **Install**, provide the required information, and then click **Add**.



**Note:** When the zone processor is installed, the service is setup, however, rights will not be automatically granted and no attempt to start the service will be made. The rights must be granted via a system or group policy. The service may be started via the services snap-in on the zone processor host or by right-clicking and selecting start service from the **Processing Services** dialog in the management console.

In the case of either a deferred or zone processor, if the services do not start, or rather start and immediately stop, this is a sign that the service account has no access to the Privileged Identity data store or is not a local administrator on the host. Other errors may arise such as those related to a bad password. Fix those errors and try again.



## View the Job Queues

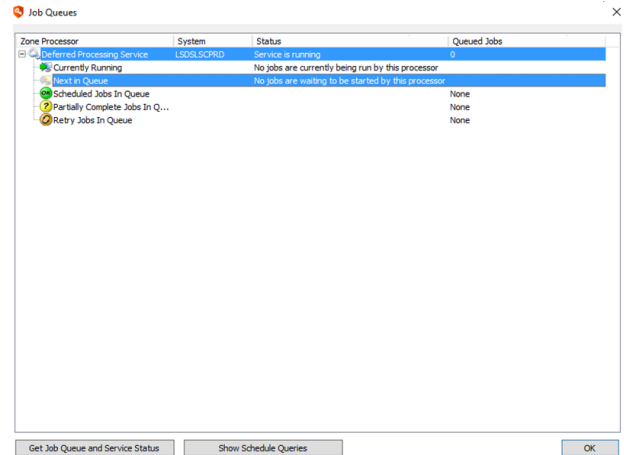
All jobs run on a first in first out basis. This means that if there are two jobs, both configured to run at exactly the same time, the first job created (lower job ID), will be run first. This also means if there is a long running job, it may cause a backlog of other jobs.

To view the jobs queue in the management console:

1. From the **Actions** pane, click the **Jobs**.
2. On the **Stored Jobs** dialog, click **Job Queues**.



- In the **Job Queues** dialog, select one or more services, and then click **Get Job Queue and Service Status**. This will attempt to query the service to see if it is running and try to determine what jobs are currently running, what is next in the queue to run and what other jobs are scheduled.

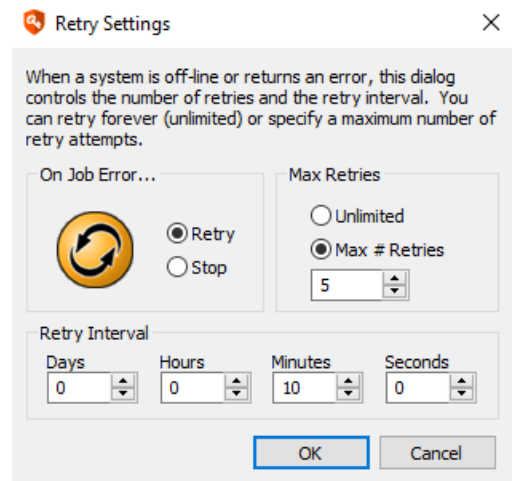


## Set Retry Settings for a Job

When a job is run but fails, it will go into a state of retry. These retry settings are configured in the management console. Go to **DeferredProcessing > Retry Policy**.

The default setting is to retry jobs on failure every 10 minutes until successful or five retries have been attempted.

The default settings may not meet the needs of a production environment. Use the **Retry Settings** dialog to define a retry policy for failed jobs and tailor it to your organizational and business needs.



## View and Edit Existing Jobs

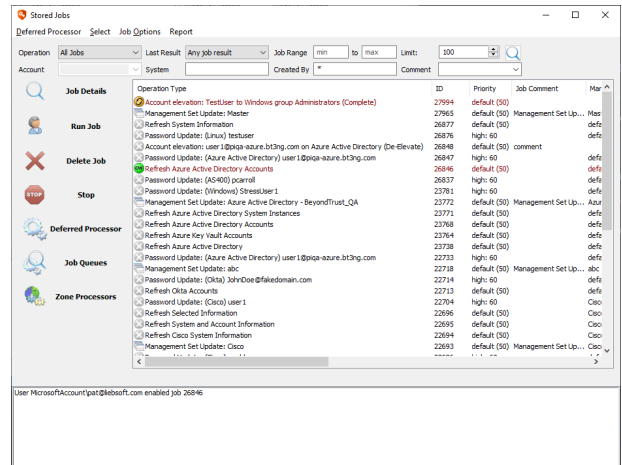
Any job that has been previously scheduled can be viewed, edited, deleted, copied, disabled or enabled by opening the **Stored Jobs** dialog in the management console. This includes jobs set to run immediately. This is useful when attempting to troubleshoot existing jobs or if password policy changes.

The **Stored Jobs** dialog shows all jobs that have been created and the current status of those jobs. Filter the type of job that is displayed by choosing a job type from the **Operation** menu, and then click the magnifying glass at the top right of the dialog. Only jobs of that operation type will be shown in the display. This dialog can also be filtered to display jobs that include a specific system or account.

Wildcards work like DOS-style wildcards: use Server\* to specify all system names that begin with "Server" and Admin\* to specify all accounts that begin with "Admin." Wildcards can be used in the **System** and **Account** fields. The result set is limited to the number of items shown in the Limit field (max is 10,000). To update the display, click the **refresh** button, which shows an image of a magnifying glass.

The **Stored Jobs** dialog lists job information in the following columns:

- **Operation Type:** the type of job and account being targeted.
- **ID:** The ID number for the job.
- **Priority:** The priority for the job. A relative number used for prioritization if two or more jobs would otherwise be scheduled to run at the same time.
- **Job Comment:** An optional comment entered by the admin who created the job.
- **Management Set:** the management set the user was in when the job was created.
- **Created By:** The user who created the job.
- **Last Result:** The most recent status of the job.
- **Last Run Duration:** Run time in seconds of the last job run.
- **Last Run Time:** The last time the job was run.
- **Next Scheduled Time:** The next time the job is scheduled to run.
- **Next Retry Time:** If the job is in a retry state, the next time the job will be retried.



Each column is sortable, but sorting operations take place after filter operations. If a particular job cannot be located, double check the filter settings.

Use the control panel on the left side of the dialog to view, run, stop, or delete the selected job. These operations are also available from the context menu by right-clicking on a specific job. Use Run Job to interactively run any scheduled jobs. The log file information will be shown in the bottom panel and all associated alerting and reporting actions will still take place. Job details (schedule, system inclusion, password settings, and so on) can be viewed or edited by viewing the details pages of the selected job.

**Note:** The Deferred Processing Service does not need to be installed for scheduled jobs to be configured or to set failed jobs to auto-retry. Any scheduled job will not run, however, until the service is installed and started.

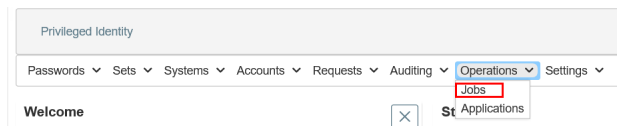
**Note:** Jobs with a status of "Do Not Delete" are indicated in red.

From the menu select **Deferred Processor** to configure the Deferred Processor and **Job Options** to configure the retry settings for failed jobs.

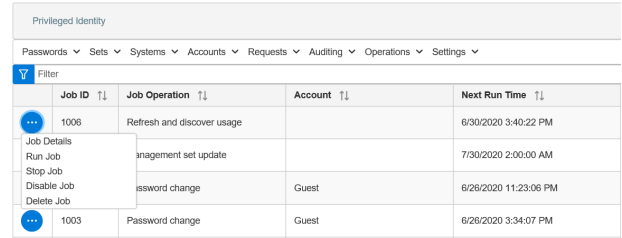
Highlight a job, and then click **Job Details** to view or edit a job's settings.

You can also view stored jobs and their job details from the web application:

1. Select **Operations > Jobs** from the menu.



- All of the jobs are listed in the grid with their ID, their next run time, and the last result for that job. To view details for a job in the queue, click the ellipsis button, and then select **Job Details**.



Job ID	Job Operation	Account	Next Run Time
1006	Refresh and discover usage		6/30/2020 3:40:22 PM
	Management set update		7/30/2020 2:00:00 AM
	Password change	Guest	6/26/2020 11:23:06 PM
1003	Password change	Guest	6/26/2020 3:34:07 PM

## Deferred Processor Status and Configuration

The deferred processor is the default service for all scheduled jobs in Privileged Identity. If the deferred processor was not installed during the initial setup, the service can be installed from the management console.

### Install the Deferred Processor

- From the **Actions** pane in the management console, click **Jobs**.
- On the **Jobs** dialog, click **Deferred Processor**.
- Supply an account name and password then click **Install**.

The account specified must be an administrator on the host system and should also be an administrator on target Windows systems.

### Other Deferred Processor Configurations

- Service Log Settings:** Click this button to view or clear the job scheduler service log file or change its path.
- Scheduler Timeout:** The scheduler timeout represents the database polling frequency. The default is 6 seconds. Every 6 seconds, the database will be polled to check for jobs that need to be run.
- Job Types:** Configure job types to set the job affinity for this particular deferred processor. The deferred processor will perform the enabled job types for any system in any management set. Typically changing job affinity will only be done when you are also working with zone processors.

## Password Change Jobs Introduction

There are many types of jobs that can be created in Privileged Identity. The most used job type is a password change job. There are multiple types of password change jobs:

- **Static password change jobs:** This type of job sets a password to a known value. This password will not ever be set to a different value without administrative intervention. This type of password change job is configured when it is desired to use the same password on a number of different accounts or when it is desired to never have this password automatically change, ever.
- **Random password change jobs:** This type of job sets a random password on the target account(s). The password is never known to anyone, not even the person who set the job up, until it is retrieved from the solution. By default, random passwords will also be triggered for automatic re-randomization once retrieved via the web application.
- **Password change jobs with propagation:** Propagation is applicable to service or process accounts. These types of accounts are used by services, tasks or applications in order to connect to a different system or service or run with a specific privilege level or scope of access.
- **Password change jobs with Account Pooling:** Account pooling is used in concert with propagation. In this scenario, Privileged Identity will rotate the identity used to run the item (service, task, etc.) and rotate the password of the new account. This option provides the benefit that should a system be unreachable during a password rotation, it will not disable or lockout the services or the account.

There are many ways to create a password change job. Select the system(s), or expand the system and select the account(s) and click the **Change Password** button on the left action pane of the management console. There are also PowerShell cmdlets and web service calls which provide this functionality as well.



*For more information, please see "[Manage Passwords and SSH Keys](#)" on page 244.*

Once a password is successfully changed, the value is encrypted and the encrypted value is stored in the PI database. Passwords may be retrieved through the web application, web service, or PowerShell.

To start a password change job, select the system(s) or target account under a system, and then click **Change Password** from the **Actions** pane.

## Password Retrieval Introduction

Once a credential has been stored in Privileged Identity, it may be retrieved in a number of ways, provided the identity has been granted appropriate access:

- **Web application:** Use the managed passwords page in the web application to retrieve or request access to the password.
- **Privileged Session via the Web Application:** Use the application launcher to create a session as a managed credential using a specific tool to connect to a specific system.
- **PowerShell:** Use the Get-LSPassword cmdlet to retrieve the password.
- **Web Service:** Use the StoredCredential method (REST) or AccountStoreOps\_StoredCredential\_CheckOut method (SOAP) to retrieve the credential.

Delegations can be performed via the management console under the Delegation menu, via the web application, web service, or PowerShell. There are many levels at which delegations may be performed:

- **Global:** This is a "many to many" delegation. This means a given user may belong to one or more groups or roles. Each may be assigned one or more permissions to one or more management sets. The cumulative set of permissions is granted to the user. All permissions assigned at this level will apply to all management sets and thus system and accounts in those management sets.
- **Per Management Set:** An identity may be granted a specific set or permissions to a specific management set. All permissions assigned to the management set apply to the systems in that management set and thus the accounts on those systems. Use this to assign different permissions to the same identity for different management sets.
- **Per System:** An identity may be granted a specific set of permissions to a specific system. All permissions assigned to the system apply to all accounts managed on that system. Use this to assign different permissions to the same identity for different systems.
- **Per Account:** An identity may be granted specific permissions to a specific account on a specific system.



**Note:** Permissions are cumulative; therefore, looser permissions assigned at higher levels, for example, Global vs Per Account, will grant a higher level of access than may be desired.



For more information, please see "[Manage Identities and Delegations for Password and System Access](#)" on page 374.

## Auditing and Alerting Introduction

Every action taken in Privileged Identity will be logged to one or more logs and may also be sent to external systems such as syslog servers (Splunk, QRadar, ArcSight, etc.), help desk incident management systems (ServiceNow, Remedy, etc.), and many more.

Auditing of Privileged Identity may occur from the web application, web service, PowerShell, or other logs depending on the item being examined.

Compliance reports exist to provide pre-canned reports of most requested information such as password under management or privileged identity access. Additionally, an framework exists called Event Sinks that enabled Privileged Identity to connect to external systems and provide this same information in multiple consumable formats. Finally, if none of the built-in mechanisms are providing the level of information or reporting needed, the data store may be mined directly using SQL queries from any program which can connect to a Microsoft SQL database.

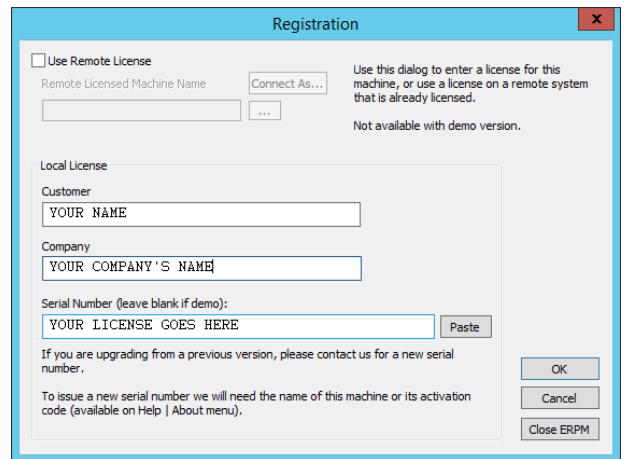


For more information, please see *"Audits and Alerts"* on page 525.

# Registration, Licensing, and Help

## Registration

Once the product is installed, go to **Help | Register** to register your product. If you have multiple management consoles tied to the same database, install a license on only one of those machines (hint: licenses are tied to specific machines by name), then click **OK**. Once the first license is installed, secondary management consoles may be installed and connected to the same data store. They will use this license automatically.

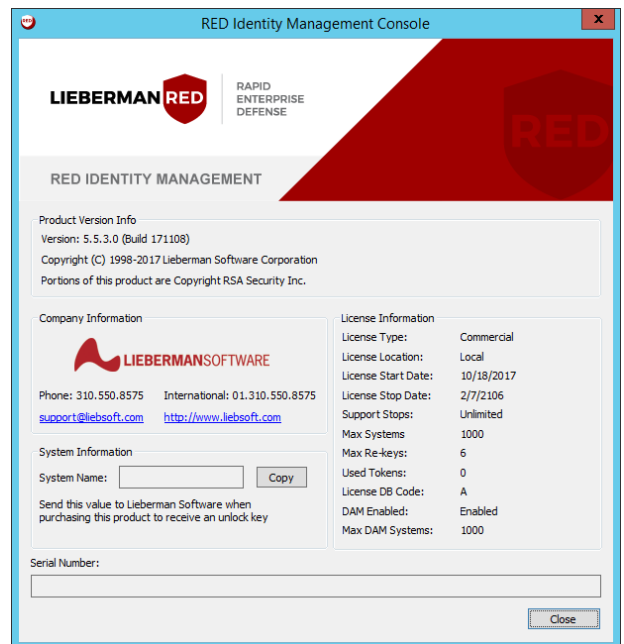


## About

Under **Help > About**, you may see information about the installed license, used token, system name and more.

Information of interest to you includes your license start and stop date as well as the maximum number of systems and used tokens counters.

Information of interest to support is the version and build information.

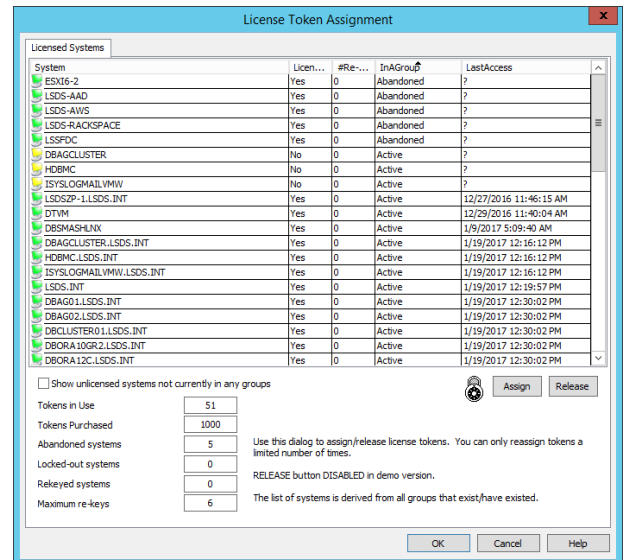


## License Keys

Go to **Help > License Keys** to open the "License Token Assignment" dialog.

The **License Token Assignment** dialog shows how many license tokens are currently in use and which systems those tokens are assigned to. Under normal operation, this dialog is not necessary. If some systems are replaced with others, this dialog can be used to release license tokens from one or more of the obsolete systems and optionally manually license the new systems.

Each system that is stored in the program's data store is listed on the left of the list. The "Licensed" column shows whether or not that system has a license token assigned to it. The **#Rekeys** column shows the number of times a license token has been removed and re-added from a specific system. This number correlates to the **Maximum Rekeys** and **Locked-out Systems** fields. If a system is re-keyed too many times, it will be locked out and not allowed to be managed by this tool.



**Note:** There is a limit to the maximum number of re-keys for each system before that system becomes locked out and can no longer be licensed. The **InAGroup** column shows whether or not the system is a current member of any managed group. Systems listed as abandoned are currently not members of managed groups. The **LastAccess** column shows the last time a specific system was successfully contacted.

The button boxes show the number of abandoned systems, locked-out systems, and re-keyed systems. Abandoned systems are systems that are not members of any managed groups. These systems may or may not be licensed. Locked-out systems are systems that have exceeded their re-key count and can no longer be assigned a license token. Re-keyed systems are systems that have had their license token removed and are no longer licensed, but can still have a license token assigned to them. There is also the maximum number of re-keys shown. The Maximum number of re-keys indicates how many times a license token can be removed from a system and re-licensed before it is locked out and can no longer be licensed.

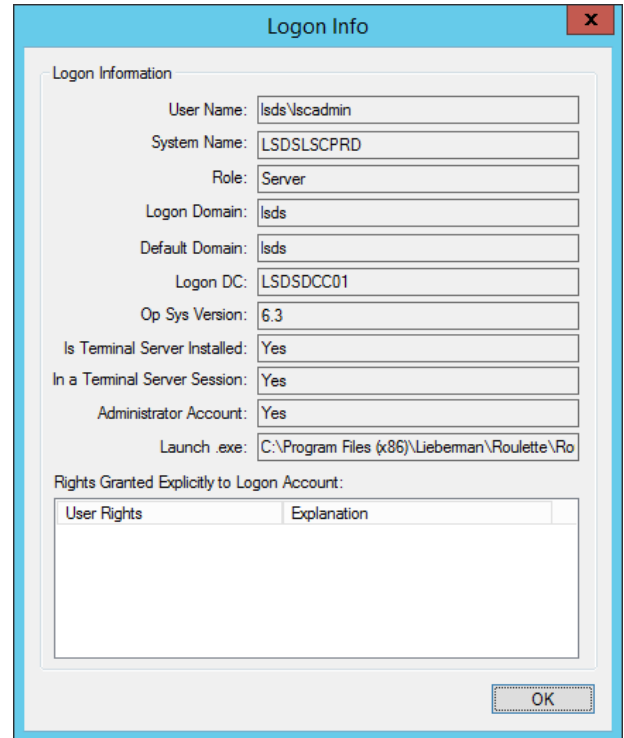
To manually remove a license token from a system, highlight the system in the list and click the **Release** button. This will remove the license token from the system, decrement the Tokens in Use count, and increment the total re-key count. The license column will show that the key has been removed and the system re-key count will not be incremented until the system is licensed again either manually or by performing an operation on it. **License tokens are assigned to machine names. If replacing a system and using the exact same name, do not release its license token.**

To manually assign a license token to a system, highlight the system in the list and click the **Assign** button. The number of Tokens in use will increment by 1 and the license column will indicate that the system has been licensed.



## Logon Info

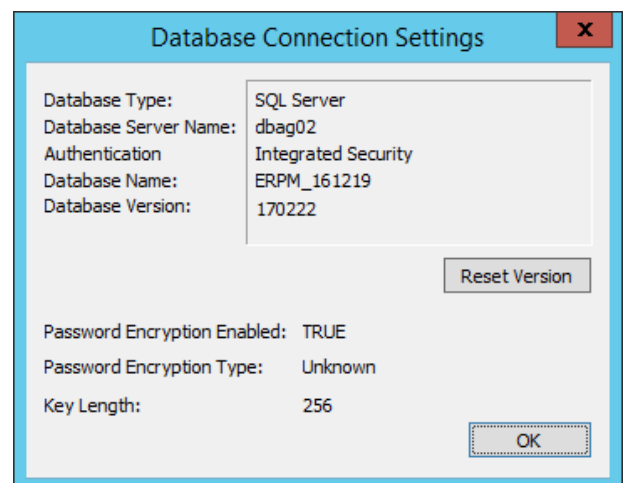
Use the **Logon Info** dialog under **Help > Logon Info** to see information about the currently logged on user, host system, and rights.



## Database Connection Settings

Use **Help > Database Configuration** to see information about the currently configured data store.

If there are problems with views, tables, or stored procedures, as can occur during a failed upgrade or other mismanagement of the data store, use the **Reset Version** button to clear out certain settings in the database and restart the management console. This will force Privileged Identity to re-evaluate the tables, views, and stored procedures and re-create anything that is missing. This is not something that should be done as part of general maintenance or troubleshooting as the re-verification procedure can cause disruption to a production installation.



## Enroll New Systems and Devices

Before any credential management or discovery can occur in Privileged Identity, you must create a management set. A management set is a list of systems or devices. Jobs and access delegations can target management sets.

You can add systems and devices to a management set in a number of ways, as follows:

- Use automated system and device discovery such as, Active Directory query, IP Scan, CMDB Query, and more.
- Use orchestration via web service (SOAP or REST) or PowerShell.
- Import via text file.
- Manually add systems one at a time using the management console or the web app.

Once systems are added to a management set, you can discover the local accounts on the systems or devices, as well as discover account meta data and other system information. You can run discovery jobs on an interactive basis or you can schedule them. This discovered information may be useful for reference and reporting requirements.

The following pages describe how to add management sets and add systems and devices to the management sets. Unless explicitly noted, steps outlined in the following pages are performed from the **Account Store View** in the management console.

In the management console, you can select machines or accounts or other objects in any view using typical selection methods such as click, **CTRL + click**, or **SHIFT + click**. Operations are typically performed against the selected object.

# Create Management Sets and Enroll Systems

A management set is a logical collection of systems and devices that you create as needed to organize system discovery and management tasks. There is no hard limit to the number of management sets you can create. A single system may appear in one or more management sets to address the necessary job and delegation requirements. Having a single system in one or more management sets will not consume multiple licenses when that one system is added by the exact same name.

When creating management sets, consider the following:

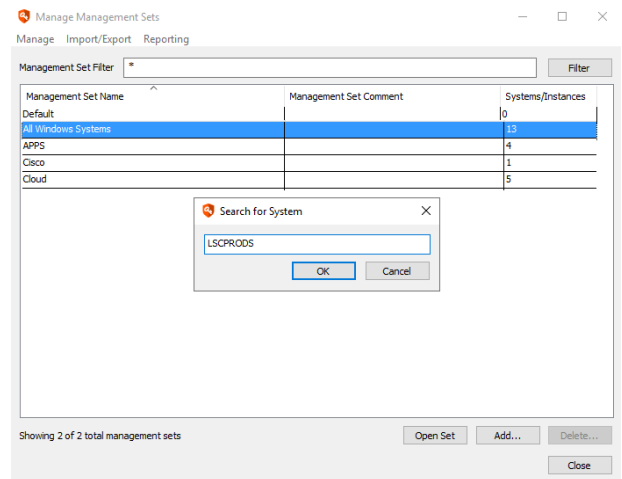
- Management sets function as zone processor boundaries (zone processors are covered later in this guide). A zone processor is assigned to one or more management sets which in turn establishes the list of systems the zone processor will be responsible for.
- A job (password rotation, discovery, etc.) can target individual systems or a single management set. Recurring jobs allows the management set to update automatically to account for any additional systems or removed systems automatically, without human intervention.
- Delegations can target individual management sets. Granting identity permissions to a management set, such as view accounts and recover passwords allows that identity to view all accounts on all systems in that management set and to retrieve their managed/stored passwords. As the management set updates itself, what the identity has access to automatically changes.

One important thing to note, is that the management set is present simply to establish a list of systems. That means a system may be moved between management sets and there will be no loss of stored credentials or related data. Thus, if you start managing and grouping your systems in one manner and later realize that a different model must be established or multiple models must be maintained, you need only to re-organize your management sets.

The only static property of a management set is the name. Once named, the management set name may not be changed.

Large lists of management sets can make it difficult to find a system within the management console. To quickly find a system:

1. From the **Actions** pane in the management console, click **Management Sets**.
2. Select any management set, and then click **CTRL + F** to open the **Search for System** dialog.
3. Type the full name of the system, and then click **Ok**. This will highlight any management sets containing this system.



## Create Management Sets

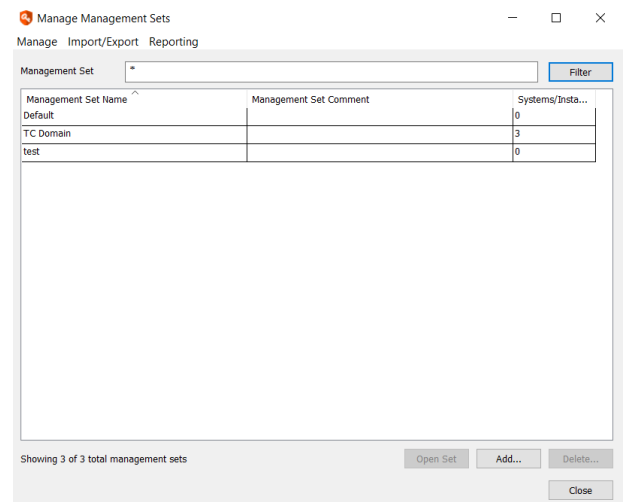
Using the management console, you can create management sets in the following ways:

- From the **Actions** pane in the management console, click **Management Sets**:
  - Click **Add** and give the management set a name.
  - Select **Manage > Automatic Set Population** from the menu, and then click **Ok** to confirm the operation. Select a domain to instruct the product to contact its domain controller and create management sets for every container found in Active Directory. The name of the auto-created management set is mapped to the container's distinguished name.
- Programmatically:
  - Use the **New-LSManagementSet** PowerShell cmdlet.
  - Use the SOAP **ManagementSetOps\_CreateManagementSet** command.
  - Use the REST **ManagementSet** command as a POST.

The first management set, called **Default**, is added automatically when Privileged Identity is installed. Initially it contains the system hosting that management console in it, but no other dynamic properties are set.

Once lists are created, you may perform the following options from the **Manage Management Sets** dialog:

- **Import/Export**: Export the list of management sets (and all of their properties) to an XML file. This would allow the full recreation of the management set on a separate unrelated management console (not sharing the same database).
- **Delete Management Set**: Select the management set, and then click **Delete**.
- **Filter the list**: Use the **Management Set Filter** (DOS style filters) to search for one or more management sets.
- **Report**: Generates a report of the management sets. This report will contain the names, comments, and instance counts for the management sets, but not their extended properties.
- **Open a management set**: Either double click the management set or highlight it, and then click **Open Set** to open the desired management set.



Once a management set has been created, add systems to the management set in one of the following ways:

- **Dynamically**: Query various sources such as Microsoft Active Directory, LDAP, CMDB, IPScan, and more, management sets can be populated and in many cases, automatically categorize the systems found.
- **Manually**: Add one system at a time through the management console, web application, web service, or PowerShell.
- **Import**: Import a list of systems from a text file.
- **Programmatically**: Use the SOAP, REST, or PowerShell commands to manage management sets without human intervention or to integrate with other system deployment models and orchestration images.

The following pages show you how to set management set properties to dynamically add systems to management sets.

**i** You can also add and configure management sets from the web application. For more information on adding and configuring management sets using the web application, please see "[Configure Management Sets in the Web Application](#)" on page 451.

# Manage Restricted and Orphaned Systems

## Add Restricted Systems

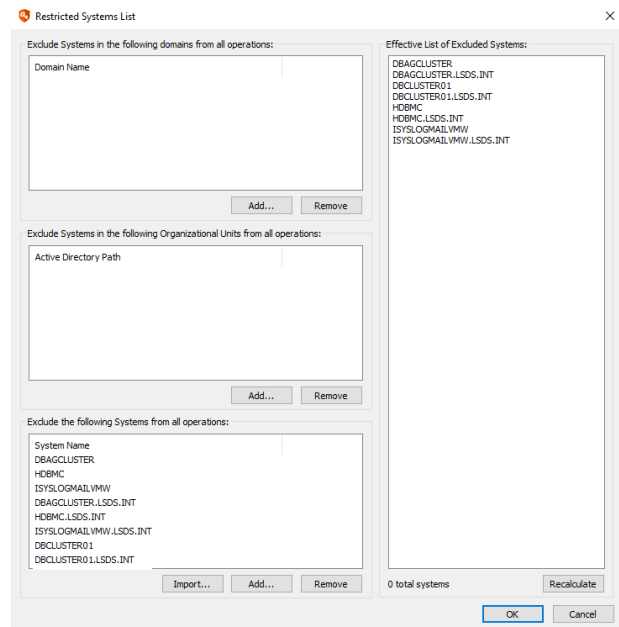
Management set properties can exclude systems on a per-management set basis. However, if there are some systems that should never be included in any operation, they need to be added to the restricted systems list. The restricted systems list is a global system exclusion list that, when a system is added to it, will prevent operations against that system (though it may still appear in a management set).

To setup a restricted list of systems, select **Systems List > Restricted Systems List** from the menu.

Systems may be excluded by entire domains (Domain Name as NetBIOS domain name), Active Directory Path (distinguished name for path to exclude) or by individual system name (NetBIOS, FQDN, or IP).

Once your systems have been added, click **Recalculate** to update the list, and then click **OK**.

When Privileged Identity performs an operation against the system, the system will be skipped and a log message indicating the system was skipped due to inclusion on the restricted systems list.

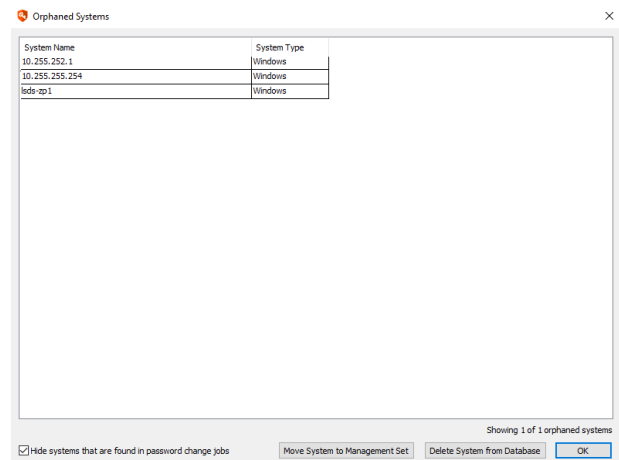


## Manage Orphaned Systems

If a system gets removed from all management sets, the system is not removed from Privileged Identity. Rather, it is removed only from the management set(s). The system may still be present in jobs and its stored passwords will still be present in the secure password store. This is done to ensure no information is lost from the product without explicit administrator intervention. To see a list of systems that have been removed from all management sets, select **Systems List > Orphaned Systems** from the menu.

When the orphaned systems list opens, it will display only systems that are not in any management sets and not included in any jobs. To show systems still in jobs, clear the check box to **Hide systems that are found in password change jobs**.

To fully remove the system (not including passwords) from the database, highlight the system(s), and then click **Delete System from Database**. To re-add the system to a management set, highlight the system(s) and click **Move System to Management Set**.




## Configure Management Set Properties

To configure the properties of a management set use the management console or PowerShell or API. Use the `Set-LSManagementSetInfo` cmdlet when using PowerShell. Use `ManagementSetOps_SetManagementSetInfo` for SOAP or REST management. This section discusses how to use the management console.

To configure a management set's properties, open the management set, and then from the **Actions** pane, click the **Set Properties**.

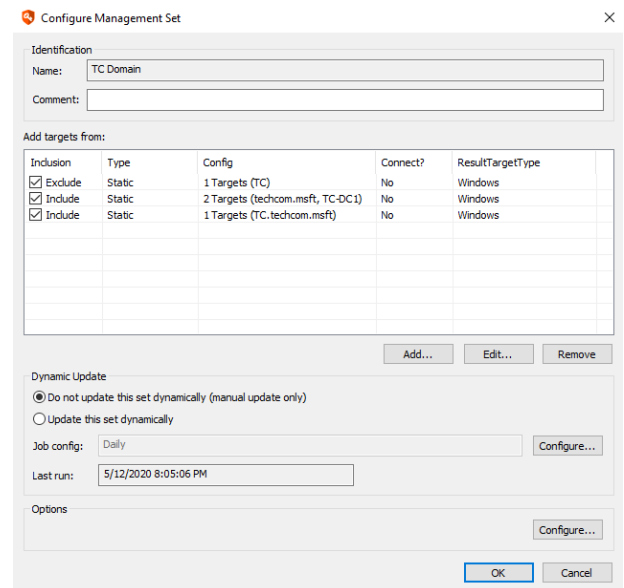
The **Configure Management Set** dialog is empty when it is first opened. Take note of the four sections on this dialog:

- **Identification:** Shows the name and comment.



**Note:** *The name of management set cannot be changed, but a comment for the management set can be provided. This comment should help define what is in the management set such as "London SharePoint Servers".*

- **Add targets from:** Lists the inclusion or exclusion properties of the management set. Those properties are configured using the **Add**, **Edit**, and **Remove** buttons just under the **Add targets from** table. The columns in this section indicate a primary property of the management set attribute:
  - **Inclusion:** Defined as **Include** or **Exclude**, identifies if the property will include or exclude systems found with that query. Include statements are processed first from top to bottom. Exclude statements are processed from top to bottom. If the check box next to the Include/Exclude property is not selected, it means the attribute is not currently enabled. This allows you to enable or disable attributes of a management set without having to delete the attribute.
  - **Type:** Identifies the type of search being performed such as AD Query or Static.
  - **Config:** Lists properties such as the AD query or CMDB query being performed.
  - **Connect?:** Identifies if the product will attempt a connection to the target systems as part of the discovery process for inclusion in the discovery range.
  - **ResultTargetType:** Tells how a target found with this criteria will be classified. Note, once a system has been found and classified, the system classification will not change again without human intervention.
- **Dynamic Update:** Defines the frequency with which this management set will (or will not) auto-update. The default time frame for a management set update job is to run every 30 days, indexed from the original time of management set's creation. Use the **Configure** button to set additional job properties including:
  - **Job Schedule:** Set the job to run as frequently as every hour.
  - **Pre-Run Alerts:** Before the management set update runs, an email may be triggered to a defined email address or addresses (semi-colon delimited list).
  - **Pre/Post Run Steps:** Define additional scripts to run before or after the management set update runs.
  - **Log:** The non-verbose log for the management set update job.



Inclusion	Type	Config	Connect?	ResultTargetType	
<input checked="" type="checkbox"/>	Exclude	Static	1 Targets (TC)	No	Windows
<input checked="" type="checkbox"/>	Include	Static	2 Targets (techcom.msft, TC-DC1)	No	Windows
<input checked="" type="checkbox"/>	Include	Static	1 Targets (TC.techcom.msft)	No	Windows

- **Options:** Additional filter options for Windows systems being added to the management set based on name or OS type. We do not recommend you change these settings when querying from any source that already keeps track of this information, such as Active Directory. Configuring these options causes the product to create a secondary connection to the target to query its name and operating system values. Thus a query that could take 20 seconds from Active Directory can take several minutes or more for every management set update.



*For more information, please see:*

- *"Management Set Job Options" on page 69*
- *"Dynamic Discovery of Systems and Devices" on page 72*



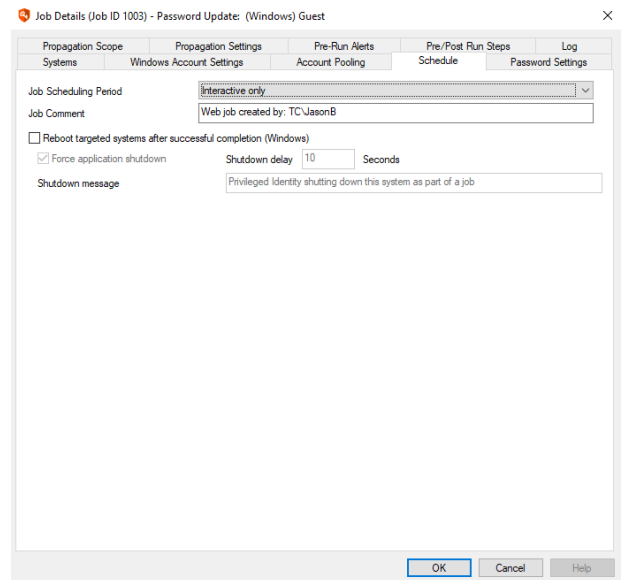
## Management Set Job Options

Every management set has a scheduled job associated with it. If the job is deleted, it will be automatically recreated. These jobs may be viewed and edited by opening the job from the management set properties or through the **Jobs** dialog.

### Schedule

Use the **Schedule** tab to define the job's scheduling period and a comment for the job.

- **Job Scheduling Period:** Choose how often the job should run. Jobs that run immediately and jobs that run once are not saved to run on a recurring basis.
- **Job Comment:** (Optional) Enter a brief description. This comment is displayed on the "Stored Jobs" screen.
- **Only run job if within time window:** Select to run the job within a given window of time. For example, suppose you set the job to run at midnight and you specify a 60 minute window within which the job can run. If the job launches at 12:45AM but takes 30 minutes to run, the job will run for fifteen minutes (until 1 AM), then pause until the next job run-time, and then finish running. The job will then reschedule.
- **Minutes into hour:** For jobs that are scheduled to run at a set time, sets the minutes portion. Enter a number between 0 and 59. Applies to the following scheduling periods: One Time, Every hour, Every day, Every week, Every month, Every year, Every N days, Every N hours.
- **Hour of day:** Sets the hour for jobs that are scheduled to run at a set time. Enter a number between 0 and 23, where 0 = midnight and 23 = 11 PM. Applies to the following scheduling periods: One Time, Every day, Every week, Every month, Every year, Every N days.
- **Day of week:** Sets the day of the week for the following scheduling period: Every week. You cannot select multiple days.
- **Day of the month:** Sets the day of the month for the following scheduling periods: Every month, Every year.
- **Month:** Sets the month for the following scheduling periods: Every year. Enter a number between 1 and 31.
- **Day Period:** For jobs that are scheduled to run every N days, sets the value for N.
- **Hour Period:** For jobs that are scheduled to run every N hours, sets the value for N.



### Pre-Run Alerts

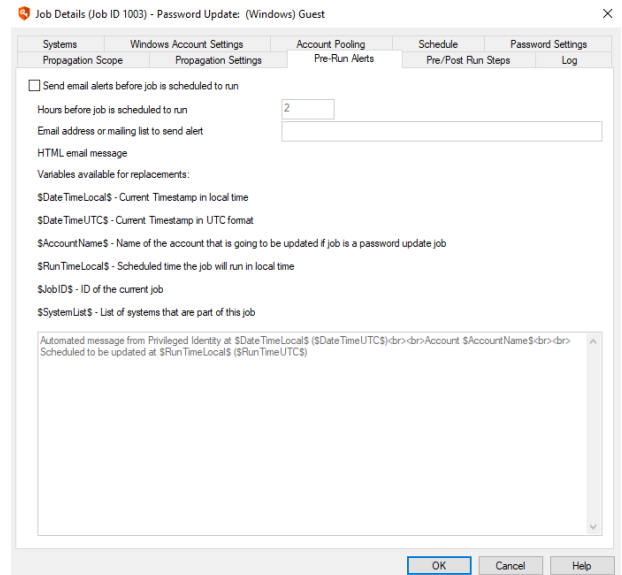
Pre-run alerts are used to send an email alert to a single email address prior to the job running.

Configure the following options:

- **Send email alerts before job is scheduled to run:** (Optional)  
Select if an alert email should be sent prior to running the job.
- **Hours before job is scheduled to run:** Enter the number of hours in advance of the job running that the alert message should be sent.
- **Email address or mailing list to send alert:** Enter the email address that the alert message should be sent to. The address can be for a user or a distribution list.

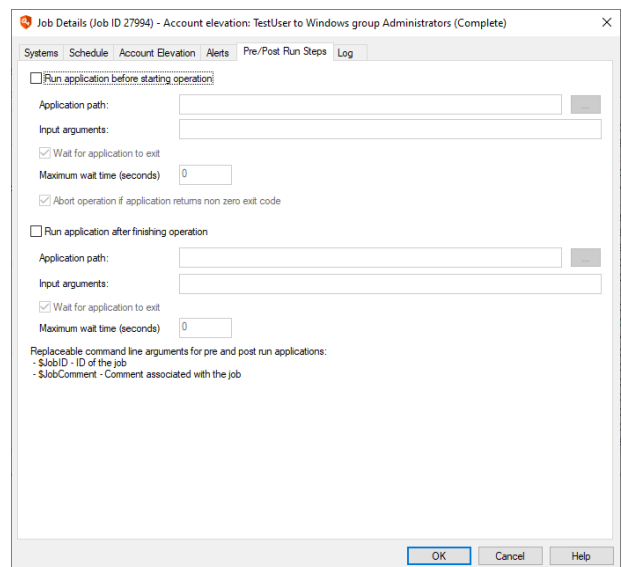
The email body is formatted in HTML and supports standard tags such as **<br>**. Each job, if not cloned from another job, will need to define its own message. The following variables are supported:

- **%DateTimeLocal%:** Current time stamp in local time relative to the deferred or zone processor running the job.
- **%DateTimeUTC%:** Current time stamp, adjusted to UTC time, relative to the deferred or zone processor running the job.
- **%AccountName%:** Name of the account that is going to be updated if the job is a password update job. This value is not used for SSH key update jobs.
- **%RunTimeLocal%:** The targeted run time of the job, in local time relative to the deferred or zone processor running the job.
- **%JobID%:** The ID of the job being run.



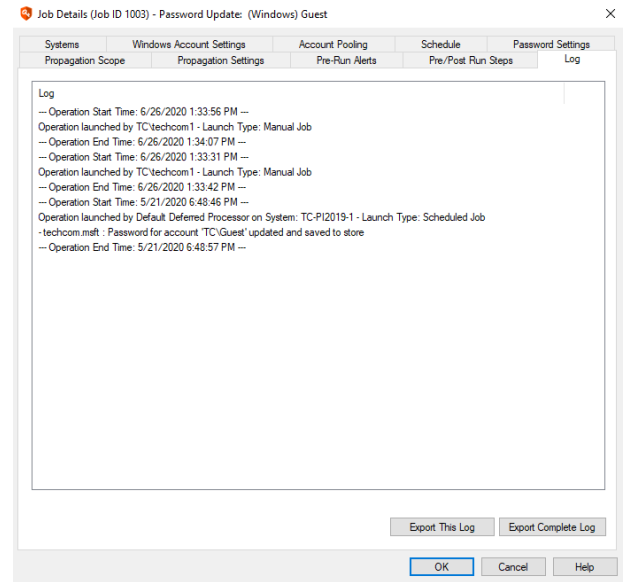
## Pre/Post Run Steps

Use the **Pre/Post Run Steps** tab to run scripts and other applications before and/or after a job runs. For example, open a software-defined networking (SDN) connection just prior to running a job, and then immediately close the connection when the job finishes.



## Log

The **Log** tab shows the operations performed during the job run. Management set jobs will have very little detail beyond the start and stop time of the job. The verbose log can be seen by right-clicking on the job via the jobs dialog and select View Text log.



## Dynamic Discovery of Systems and Devices

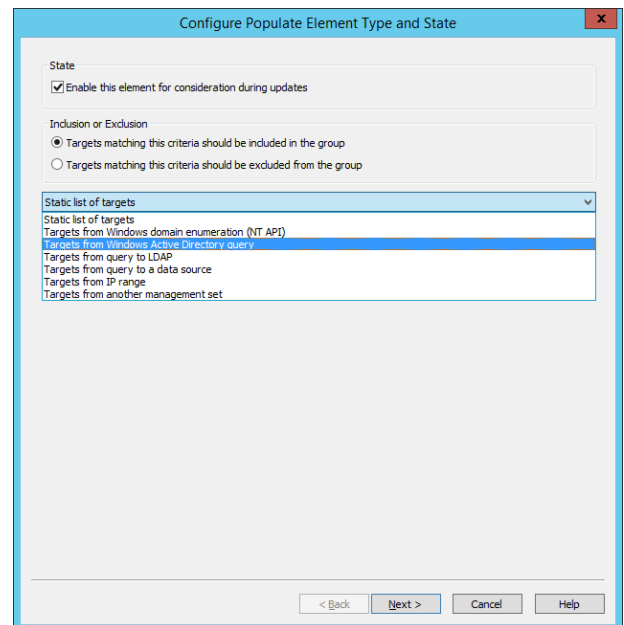
This section describes how to configure dynamic discovery options. Every platform that Privileged Identity can manage can be added by manual or programmatic methods. Currently, only some platforms, such as Windows and Linux and more, can be discovered and automatically categorized using the dynamic options.

If the target network is relatively static, or if only a small number of systems need be grouped together, manually choosing those systems may be easier than defining dynamic selection criteria. However, if the target network is highly dynamic and constantly adding and/or removing systems, dynamic selection offers many time saving and accuracy benefits.

To start configuring the inclusion or exclusion ranges of any management set, open the management set, click on Set Properties from the left action pane of the management console, then click the **Add** button to add a new property and define some basic options.

Basic on this dialog include:

- **State:** When enabled (default), this property will be actively evaluated for inclusion or exclusion of systems.
- **Inclusion or Exclusion:** Identify if this criteria will cause found systems to be included or excluded from the management set. The default property is set for inclusion.
- Discovery Properties include:
  - **Static list of targets.**
  - **Targets from Windows domain enumeration (NT API).**
  - **Targets from Windows Active Directory query:** this could include non-Windows systems as well.
  - **Targets from query to LDAP.**
  - **Targets from query to a data source:** This could be any data source for which there is a valid database provider installed on the host system. Customers have used other management systems such as Microsoft System Center Operations Manager or BMC Asset Management systems.
  - **Targets from IP range:** Define one or more IP ranges to scan for systems.
  - **Targets from another management set:** this option is useful when combining management sets for the purposes of delegated access or zone processor affinity.




**Note:** When systems are added using the manual methods, they are added to the dynamic group properties (system set properties) as **Explicit Inclusions**.

Once the inclusion/exclusion element has been defined, the last step will be to define any addition sub-discovery attributes or classification attributes.



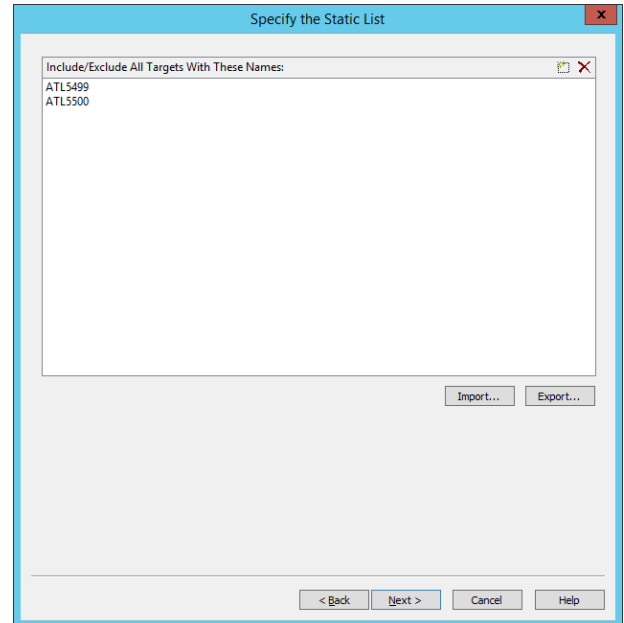
For more information, please see "[Map Scanned Targets](#)" on page 85.

## Add a Static List of Targets

Any system manually added to any node in the management console is also added as a static target to the static list of the management set. Similarly, any system added to the static list will be added to the management set. However, the classification of the system when adding the system to the management set's static list first, will be limited to Windows, Linux, or Cisco devices, unless additional steps are taken later to re-classify the system.

Static lists of this kind are useful when lists of systems are short and the environment rarely changes.

Once the inclusion/exclusion element has been defined, the last step will be to define any addition sub-discovery attributes or classification attributes. For more information, see "[Map Scanned Targets](#)" on page 85.

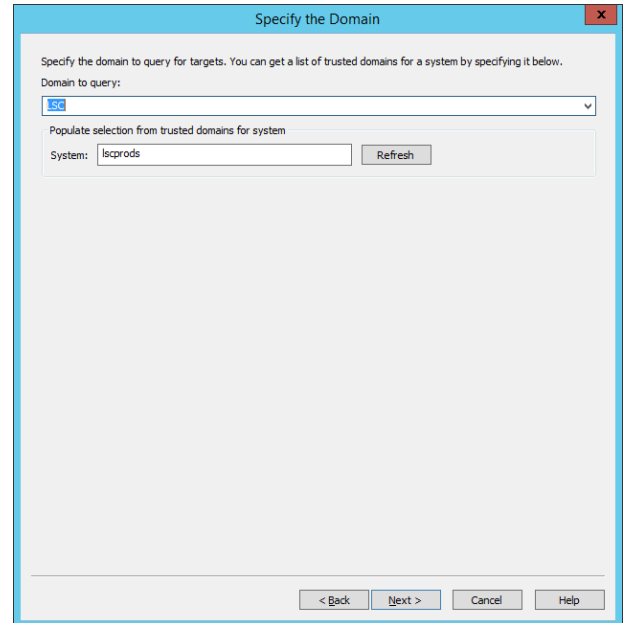


## Add Targets from Windows Domain Enumeration

Adding targets from the Windows Domain Enumeration (NT API) uses the API introduced with legacy Windows NT-based domains. This options is generally not desirable when attempting to query an Active Directory domain as the API is slow and provides no filtering capabilities and the authentication methods are limited only to integrated authentication.

Either type in the name of the domain to query (NetBIOS domain names only!) or specify the name of a system to query and click refresh to obtain a list of trusted domains. Select the target domain then click **Next**.

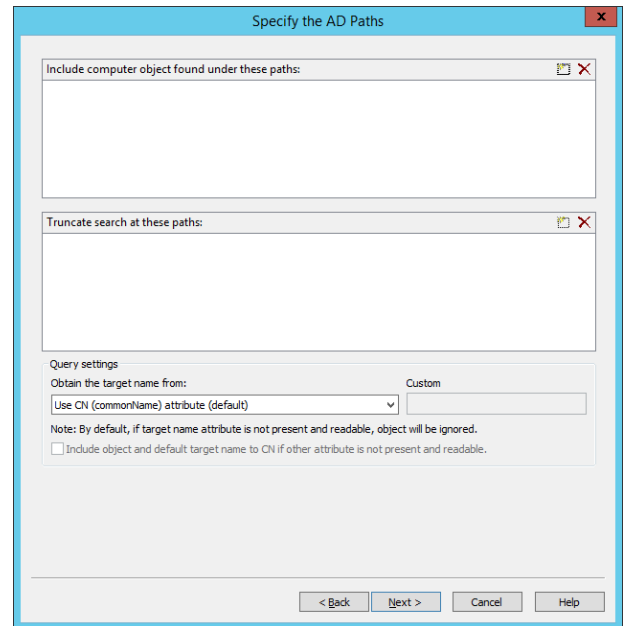
Once the inclusion/exclusion element has been defined, the last step will be to define any addition sub-discovery attributes or classification attributes. For more information, see "[Map Scanned Targets](#)" on page 85.



## Add Targets from Windows Active Directory Query

The AD Paths dialog is separated into three fields:

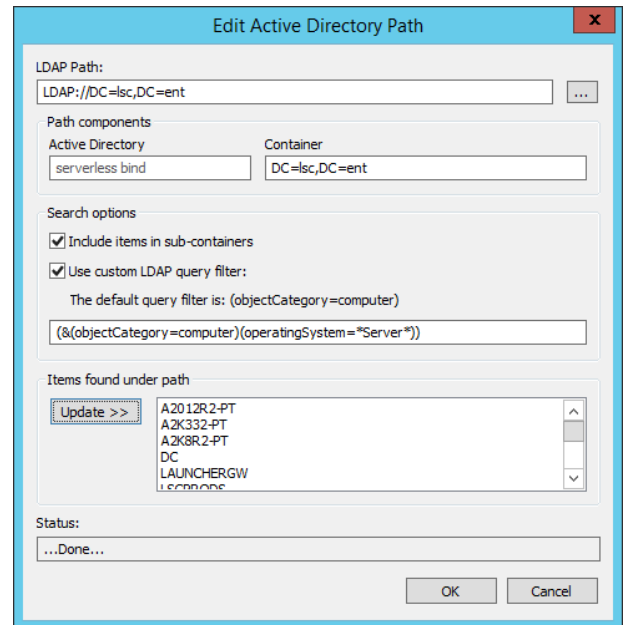
- **Inclusion:** Include computer objects found under these paths.
- **Exclusion:** Truncate the search at the defined paths. Setting a path here will exclude the AD path and child containers at the defined path.
- **Query Settings:** Defines the computer attribute to query as the discovered computer's name.
  - **CN or commonName** is the default value. This returns just the short name.
  - **dNSHostName or DNS name** returns the FQDN value of the object (when present, typical).
  - **Custom Value:** You may define the value to query for. Use ADSI Edit or other LDAP browsers to see potential attributes you may wish to return to Privileged Identity to be used for the target system name.
  - **Include object and default target name to CN if other attribute is not present and readable:** when enabled, if the custom attribute or DNS name is not present and readable, the CN attribute of the object will be included. If the box is not selected and the queried attribute is not present and readable, the object will not be included in discovery.



Click the **new** button (box) to add an include or exclude path. There are no hard limits on how many paths may be defined for a single management set.

Define the following information:

- **LDAP Path:** the distinguished name of the path using LDAP URL format to query. Click the Ellipses (...) to browse for the parent container to query. This path will be updated automatically if the Active Directory or Container fields are modified.
- **Active Directory:** leave blank to use a serverless bind (default) or specify an exact domain controller by name to query.
- **Container:** type in the distinguished name of the container or OU to query. This path will be filled in automatically if you used the browser option next to the LDAP path.
- Search Options:
  - **Include items in sub-containers:** Default is enabled. When enabled, the search will start at that parent object and include all child containers. When disabled, the search will include only items found directly in the container.
  - **Use custom LDAP query filter:** Default is not enabled. Default query when not enabled is `(objectCategory=computer)`. Use standard LDAP queries to populate this field to limit the search results. See below for more information.



When creating systems lists, you can use **Filter Options** to look for specific names or operating system versions as noted from the main management set dialog. Using this process consumes quite a bit more bandwidth and time than is necessary to filter out for operating system types and names, especially when the systems being queried for are to be found in AD. The reason this is an expensive operation is that each system must be contacted to determine if it meets the defined criteria. In total, it means the systems list is derived from AD first, and then imported into the systems list. Then a series of secondary connections are made to the target systems to identify if the system meets the filtered list of criteria. The systems list is then re-filtered to contain only systems that meet the filter. The larger downside is that if a system is offline during this operation, this process cannot be performed and thus the system will remain in the systems list and potentially be managed if the list is not updated prior to the job running.

If everything is in AD, the best practice is to use a custom LDAP query filter to aid in finding and filtering for systems. The most obvious benefit is the cost of this query: a single LDAP query to one domain controller to obtain all the information needed without ever contacting the target systems or performing post filtering for each system in the systems list.

When generating an LDAP query, be aware of how the query is formed. The rules are similar to those used for regular expressions but the syntax is slightly different.

\* = anything, any number of characters. For example, `joe*` would return "joe, joey, joe1234567890", and so on.

? = single character. For example, `j o ?` would return "joe, joy, jot" and so on.

| (pipe) = or

& = and

! = not

Single expressions are all grouped with parenthesis. For example:

```
(objectCategory=computer)
```

Would return every computer at the target LDAP container.

To include multiple expressions, join them with an & and a set of parenthesis. For example, to find all computers whose account name started with LA:



All computers = (objectCategory=computer)

- **Name starts with LA** = (sAMAccountName=LA\*)

Would be written as:

```
(&(objectCategory=computer)(sAMAccountName=LA*))
```

To include multiple expressions, join them with an & and a set of parenthesis. For example, to find all computers whose account name started with LA, but excludes Windows 2003 systems:

- **All computers** = (objectCategory=computer)
- **Name starts with LA** = (sAMAccountName=LA\*)
- **Windows 2003 Operating System** = (operatingSystem=Windows Server 2003)

Would be written as:

```
(&(&(objectCategory=computer)(sAMAccountName=LA*))(! (operatingSystem=Windows Server 2003)))
```

To include multiple expressions, join them with an "&" and a set of parenthesis. For example, to find all computers whose account name started with LA, but excludes Windows 2003 or Windows XP systems:

- **All computers** = (objectCategory=computer)
- **Name starts with LA** = (sAMAccountName=LA\*)
- **Windows 2003 Operating System** = (operatingSystem=Windows Server 2003)
- **Windows XP Operating System** = (operatingSystem=Windows XP)

Would be written as:

```
(&(&(objectCategory=computer)(sAMAccountName=LA*))(!(|(operatingSystem=Windows Server 2003)(operatingSystem=Windows XP))))
```

Break apart the last query to see the steps a little easier:

```
(&
  (&
    (objectCategory=computer)(sAMAccountName=LA*)
  )
  (!
    (|
      (operatingSystem=Windows Server 2003)
      (operatingSystem=Windows XP)
    )
  )
)
```

Queries can be much more or less complex than what is shown here. Any attribute present in Active Directory may be used for a possible query. Three additional and useful computer filters are:

- **Disabled account:** userAccountControl:1.2.840.113556.1.4.803:=2
- **Domain Controllers:** userAccountControl:1.2.840.113556.1.4.803:=8192
- **Global Catalogs:** (&(objectCategory=nTDSDSA)(options:1.2.840.113556.1.4.803:=1))

To find all computers and exclude all disabled computer accounts:

- **All computers** = (objectCategory=computer)
- Disabled account: (userAccountControl:1.2.840.113556.1.4.803:=2)

Would be written as:

```
(&(objectCategory=computer) (!(userAccountControl:1.2.840.113556.1.4.803:=2)))
```

Management set properties can be configured in the Set-LSManagementSetInfo PowerShell cmdlet, or through the web service

(URI:ManagementSetOps\_SetManagementSetInfo).

Once the inclusion/exclusion element has been defined, the last step will be to define any addition sub-discovery attributes or classification attributes. For more information, see "[Map Scanned Targets](#)" on page 85.

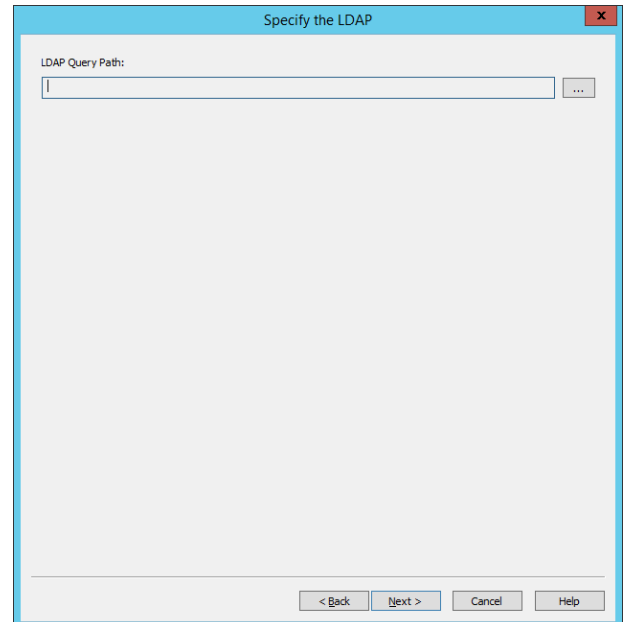
## Add Targets from Query to LDAP

Any LDAP compliant directory, including Active Directory may be added as an LDAP source to query for a list of systems. To query such an LDAP source, the LDAP source must be added as an Authentication Server.



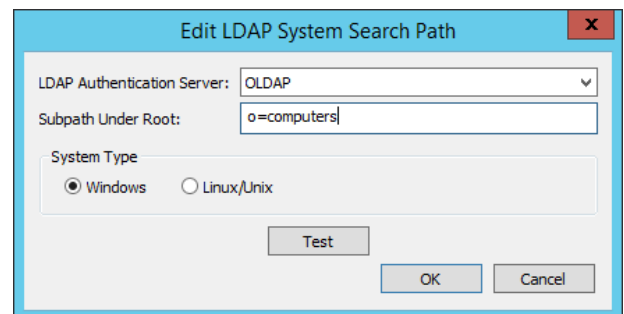
**Note:** For more information about configuring authentication Servers, see "[Configure Authentication Servers](#)" on page 310.

1. Once an authentication server exists, click the ellipses (...) to the right of the LDAP Query Path field.
2. ID Mgr
3. Select the appropriate LDAP Authentication Server entry from the LDAP Authentication Server drop list.
4. Define the query path using the distinguished name of the container to query. Note that there is no filter options on this dialog to fine tune the LDAP query. The query used is defined on the authentication server entry.
5. Set the system type as Windows or Linux/Unix.
6. Click **Test** to validate the query is valid.
7. Click **OK**.



The "Specify the LDAP" dialog box features a text input field labeled "LDAP Query Path:" with an ellipsis button to its right. At the bottom, there are four buttons: "< Back", "Next >", "Cancel", and "Help".

Once the inclusion/exclusion element has been defined, the last step will be to define any addition sub-discovery attributes or classification attributes. For more information, see "[Map Scanned Targets](#)" on page 85.



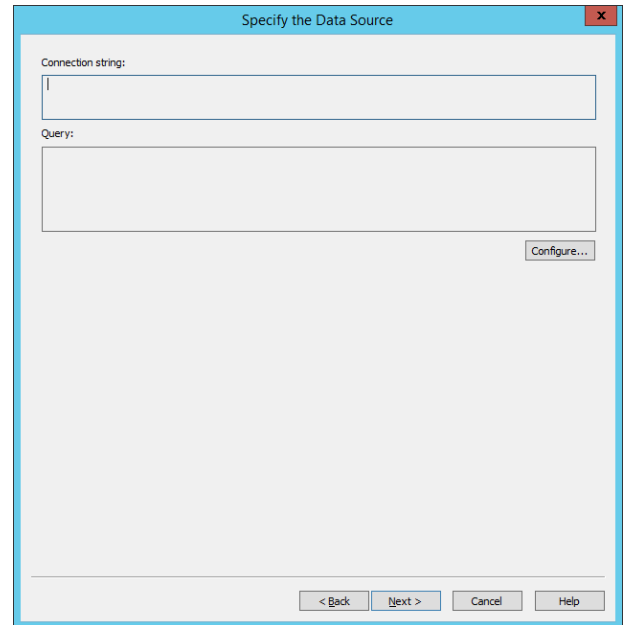
The "Edit LDAP System Search Path" dialog box contains the following fields and controls:

- "LDAP Authentication Server:" dropdown menu with "OLDAP" selected.
- "Subpath Under Root:" text input field containing "o=computers".
- "System Type" section with two radio buttons: "Windows" (selected) and "Linux/Unix".
- "Test" button.
- "OK" and "Cancel" buttons at the bottom right.

## Add Targets to Query from a Data Source

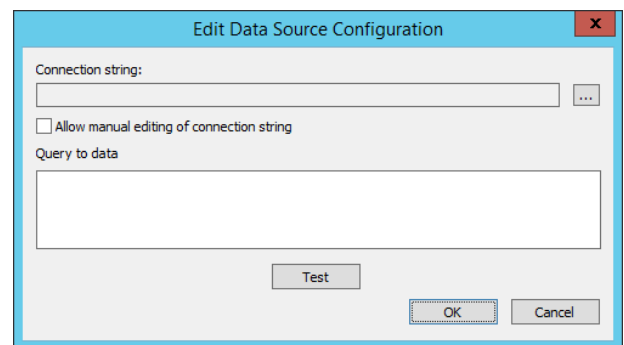
Any data source that the host system has a provider for may be queried for a list of systems. The returned list of systems should include only the system name and nothing more.

1. Click **Configure**.



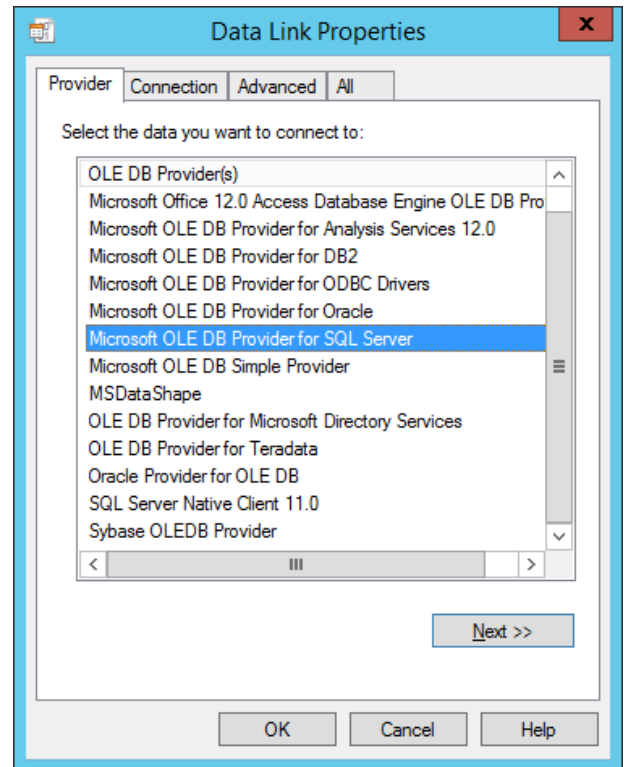
The "Specify the Data Source" dialog box features a title bar with a close button (X). It contains two text input fields: "Connection string:" and "Query:". A "Configure..." button is positioned to the right of the "Query:" field. At the bottom, there are four buttons: "< Back", "Next >", "Cancel", and "Help".

2. Click the ellipses (...) to the right of the Connection String field.

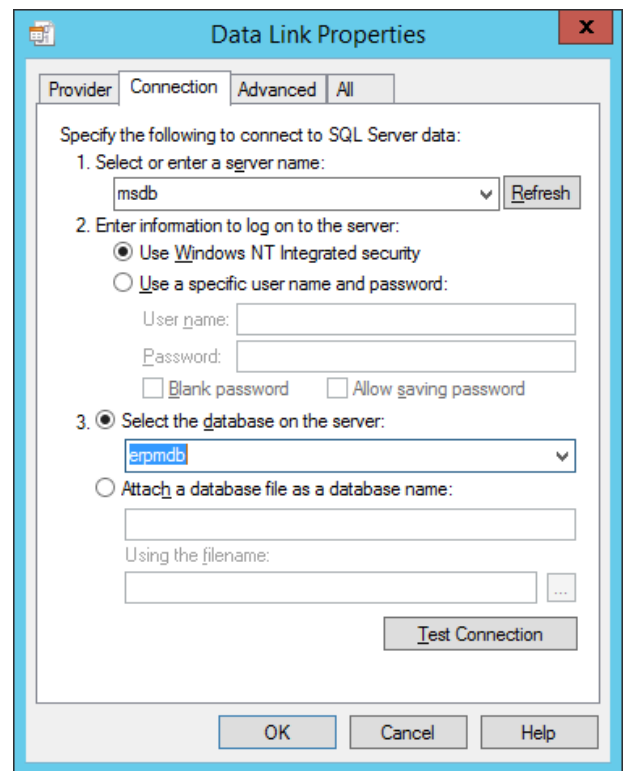


The "Edit Data Source Configuration" dialog box has a title bar with a close button (X). It includes a "Connection string:" label above a text input field with an ellipsis (...) button to its right. Below this is a checkbox labeled "Allow manual editing of connection string". A "Query to data" label is above another text input field. At the bottom, there are three buttons: "Test", "OK", and "Cancel".

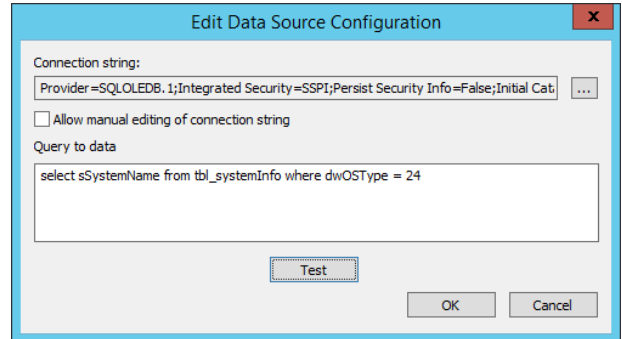
3. On the Data Link Properties (Microsoft dialog), select the appropriate provider. Note that this dialog lists only OLE DB providers. If you must use a separate ODBC provider, you will need to create a system DSN (using %systemroot%\syswow64\ODBCAD32.exe, refer to Microsoft documentation) and select Microsoft OLE DB for ODBC Drivers on this dialog.
4. Click **Next**.



5. On the connection tab specify the target server name, authentication method, and if appropriate, the target database on the server. Actual elements will vary based on the provider selected. The screen shot below depicts connecting to a Microsoft SQL Server database with Integrated Authentication.
6. Click **OK**.

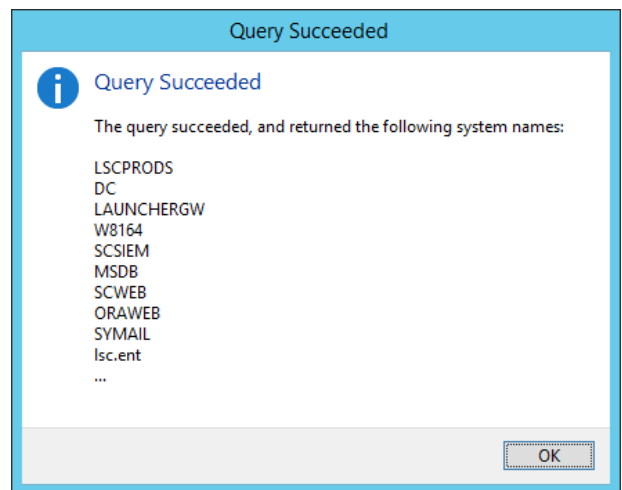


- Provide the SQL statement to query the database with. This query should return only system names.



- Click **Test** to validate the query returns data and does not fail.

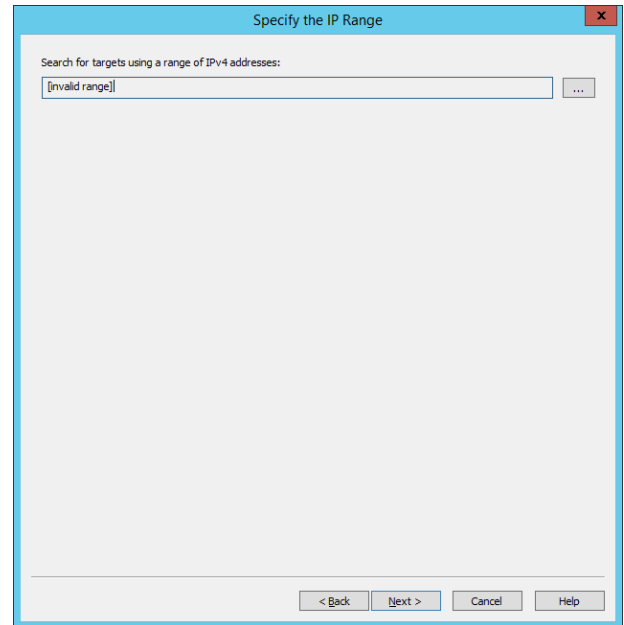
Once the inclusion/exclusion element has been defined, the last step will be to define any addition sub-discovery attributes or classification attributes. For more information, see "[Map Scanned Targets](#)" on page 85.



## Add Targets from IP Range

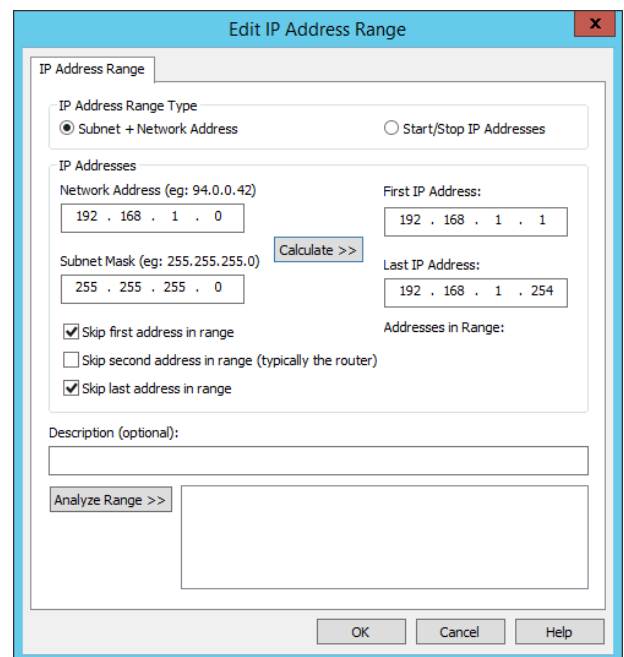
When the IP Range dialog is opened and not configured, the IP range field will display "[invalid range]". Once configured, displays the subnet information to query.

1. Click the ellipses (...) to the right of the IP range field.



2. Specify either a subnet address and subnet mask and click **Calculate** or specify a specific Start/Stop IP Address.
3. Specify to Skip the first, second, and/or last address in the range.
4. Provide a description if desired.
5. Click **Analyze Range** to determine if the range was input correctly.
6. Click **OK**.

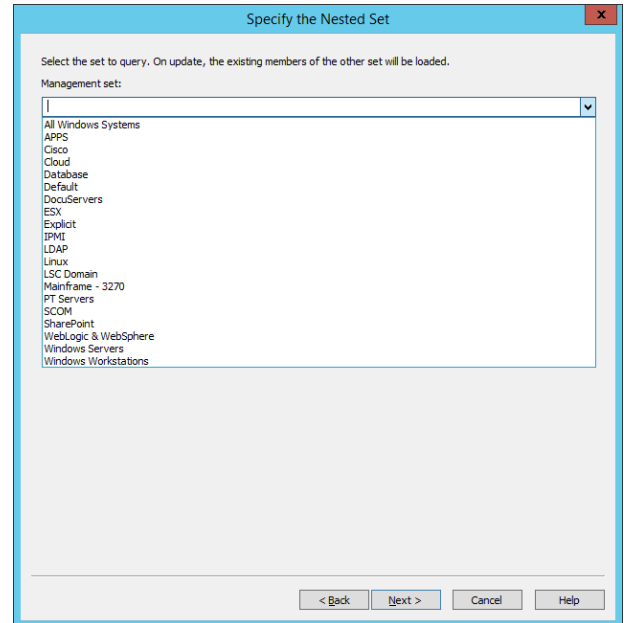
Once the inclusion/exclusion element has been defined, the last step will be to define any addition sub-discovery attributes or classification attributes. For more information, see "[Map Scanned Targets](#)" on page 85.



## Add Targets from Another Management Set

Select this option to add all targets from another management set to this management set at the time of update. This option allows you to aggregate smaller management sets for management and/or delegation purposes.

Once the inclusion/exclusion element has been defined, the last step will be to define any addition sub-discovery attributes or classification attributes. For more information, see "[Map Scanned Targets](#)" on page 85.





## Map Scanned Targets

Once a management set discovery property has been configured, the last dialog will ask how to map what is discovered to a real system and configure any additional steps to take with that system.

### 1. First define how to "type" the system:

- **Manually** - Any systems discovered through the discovery operation will be configured as this type of system. Be aware that any system previously added as a particular type of system will not be "re-typed". Options include:

- Windows
- Linux/Unix

- **Scan for Target Type** - Using one or more connection methods and authentication methods and resolution methods, classify the systems discovered through the discovery operation as this type of system. Be aware that any system previously added as a particular type of system will not be "re-typed". Options include:

- Windows
- Linux/Unix
- Cisco IOS
- SQL Server Instance
- Oracle Instance
- IPMI (Lights out, BMC, etc.)
- Xerox Phaser Printer

- **If no mapping, use** - with this option selected, if the dynamic mappings cannot categorize the system, Privileged Identity will default the found system to a known type. Options include:

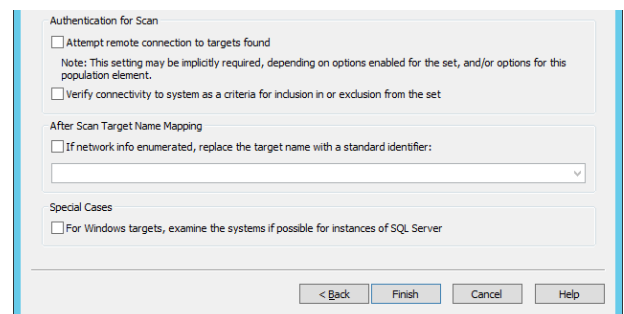
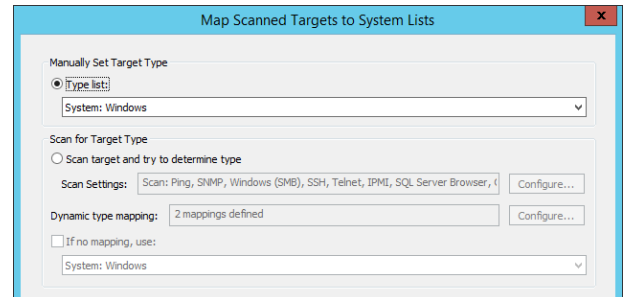
- Windows
- Linux/Unix

For more information on how to use dynamic type mappings, see the next section, "[Dynamic Mapping Settings](#)" on page 87.

For more information on how to configure Scan settings, see "[Configure Scan Settings](#)" on page 90.

### 2. Configure any authenticated scan settings:

- **Attempt remote connection to targets found** - This option may be selected automatically based on you use of the dynamic mapping settings as these dynamic mappings will require a connection to the target systems.
- **Verify connectivity to system as criteria for inclusion or exclusion from the set** - If connectivity tests will be performed, enabling this option mandates a connection must occur (based on scanner setting requirements for protocol and authentication level) in order for the system to appear. If a system is found but connectivity is not achieved,



and this option is not selected, the system will still be added to the management set based on the "If no mapping, use" setting or will be added as an "uncategorized target" to the Uncategorized Target node (hidden by default, see management set view filter options).

3. Configure name mappings. Typically not enabled, enabling **If network info enumerated, replace the target name with a standard identifier** allows you to change the system name from what was discovered in the discovery step to a different name based on information resolved by querying the target for alternate names. For example, if an AD query returns only the CN name, but upon connecting to the system you are able to glean the DNS Name when querying the system, the DNS name will be used instead when adding the system to the management set.
4. Configure Special Cases. Windows target may be scanned for installed instances of SQL Server installed on the Windows host. This makes a remote registry call to the target system to obtain installed SQL Server installation information.

## Dynamic Mapping Settings

Use the "Manage Dynamic Type Mappings" dialog to configure dynamic type-mapping rules.

There are two ways to open the Dynamic Type Mappings dialog:

- Choose **Systems List | Configure Dynamic Type Mapping for Target Discovery**. These entries will be available to any and all management sets.
- Configure management set properties. Add a discovery property to the management set. The last dialog will allow configuration of the

## About Dynamic Type Mappings

Dynamic type-mapping rules can identify new systems and assign them the proper target classification. In this release, the following target classifications are available:

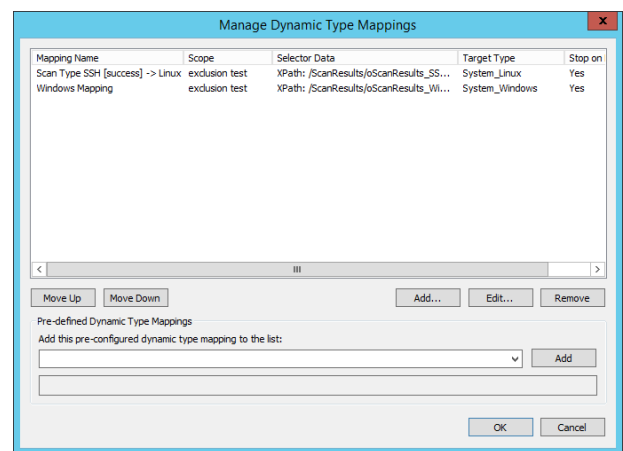
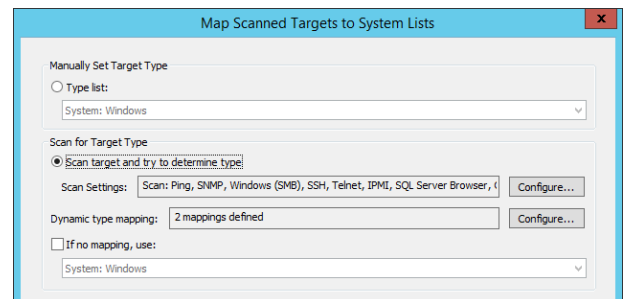
- Windows
- Linux/Unix (through SSH)
- Cisco IOS
- SQL Server instance
- Oracle instance
- IPMI
- Xerox Phaser Printer

If the system scanner cannot match a newly added system to a known type, create one or more rules that can make the proper match.

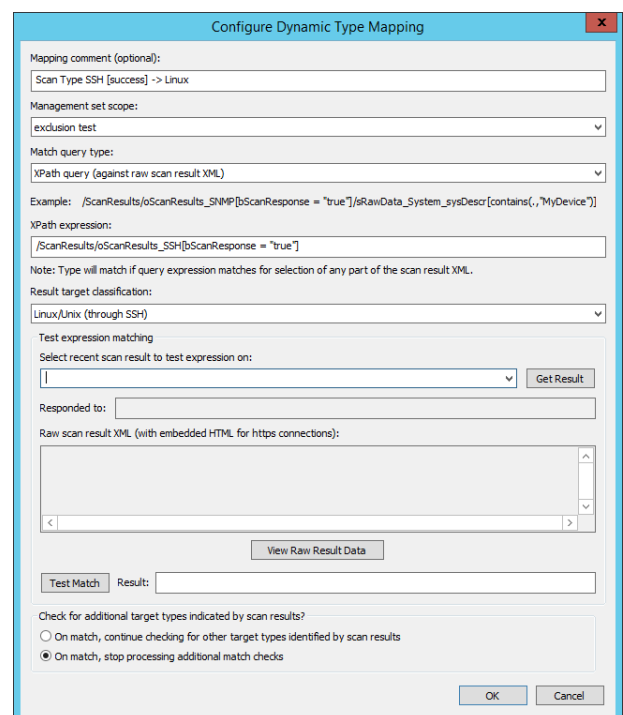
Scan results are written to an XML file any time a management set is updated dynamically. Dynamic type mapping rules parse the scan results and, when a match is made, the system is assigned the target classification specified by the rule. Create rules that identify systems based on matching attribute values - for example, a known keyword in a system description field. You can use either basic pattern matching, XPath-based mapping, or regular expression matching.

- XPath-based mapping allows you to query for data in XML nodes or node sets that meet specific criteria. XPath-based mapping is covered later in a section titled "Understanding XPath Expressions."
- Basic pattern search matching is less precise and is more likely to return false-positive matches. For example, if you want to find NETBIOS computer names that have "Test" in their name, the query \*Test\* will return a false-positive match based on the extraneous XML element <CredentialDataTestResultList/>.
- When creating basic pattern search expressions, use asterisks before and after the search phrase. Basic pattern search matching is not case sensitive.
- Regular expression matching allows you to use regular expression search operators to search for a matching value anywhere in the XML scan result record.

1. To use dynamic mappings when configuring a management set, on the final dialog of the management set properties, set the Scan for Target Type to Scan target and try to determine.
2. Next configure Scan Settings and configure Dynamic type mapping.
3. If necessary configure the **If no mapping use** settings. When enabled, if a discovered system cannot be typed using the mappings, the system will be added as Windows or Linux.
4. To configure a mapping, click **Configure** next to **Dynamic Type Mapping**.
5. From the **Manage Dynamic Type Mappings** dialog, click **Add** to add a new scan setting or click **Edit** to edit an existing one.
6. Use the **Move Up** or **Move Down** buttons to re-order the scanner settings.



7. Provide a mapping comment. This label will be seen only in the scanner.
8. Provide a **Management set scope**. This identifies that the scanner setting should be used for any management set or an explicit management set.
9. Define the Match query type. Options for this settings are:
  - **Basic Dos-Style pattern search** - XML scan results will be parsed using DOS style search filters, e.g. \*Linux\*.
  - **XPath query** - XML scan results will be parsed using XPath nomenclature to navigate the XML scan results. Hint, it will be helpful to perform a scan first to see what the scan looks like as results will vary based on connection and authentication options selected for the scan.
  - For more information see the next sub-section, "[Help with XPath](#)" on page 89.
  - **Regular expression** - XML scan results will be parsed using regular expressions.
10. Configure Result target classification. This Setting identifies what a discovered system will be classified as (and what node it is added to) if a scanner match is made.
11. To test your expression and see the results, if a scan has been run previously, select a recent scan from **Select recent scan result to test expression on** and click **Test Match**. Use the View Raw Result Data to see the full XML output of the scan. This can be helpful as moving the XML to another editor may let you see the full scan results.



12. Finally configure the **On match** settings. The options are:
  - On match, continue checking for other target types identified by scan results.
  - On match, stop processing additional match checks.
13. Click **OK** to add the mapping.

## Help with XPath

An XPath expression allows you to query for data in a specific XML element. If you are new to XPath expressions, note that XPath expressions as implemented in Privileged Identity specify the absolute path required to reach a target node or node-set in an XML document. If a matching node is found for the XPath expression then the rule evaluates to true.

For example, consider a scan that returns the following SNMP data for a Xerox Phaser printer:

```
<ScanResults>
  ...
  <bScanResponse>>true</bScanResponse>
  ...
  <oScanResults_SNMP>
    ...
    <ConnectionData_sCommunityString>public</ConnectionData_sCommunityString>
    <sOnError_ErrorMessage>Timeout.</sOnError_ErrorMessage>
    <sRawData_System_sysDescr>Xerox Phaser</sRawData_System_sysDescr>
    <sRawData_System_sysName>Xerox Color Printer B07F03</sRawData_System_sysName>
    <sRawData_System_sysLocation/>
    ...
  </oScanResults_SNMP>
</ScanResults>
```

A rule based on the following XPath expression will match the scan result because the `<sRawData_System_sysDescr>` node contains the "Xerox" target text:

```
/ScanResults/oScanResults_SNMP[bScanResponse = "true"]/sRawData_System_sysDescr[contains(., "Xerox")]
```

Compare the sample XPath expression with the sample XML scan result and take note of the following:

- The expression starts with the root node (`ScanResults`), descends one level down to the next matching child node (`oScanResults_SNMP`), then descends to the target node (`sRawData_System_sysDescr`).
- Fragments enclosed in square brackets are used to specify nodes that meet some specified condition. (These fragments are called predicates.) The predicate `[bScanResponse = "true"]` specifies that, to be true, the `oScanResults_SNMP` node must contain the `bScanResponse` element and that element must evaluate to true.
- The predicate `[contains(., "Xerox")]` specifies that, to be true, the `sRawData_System_sysDescr` node must contain the string `Xerox`.
- The `contains` function evaluates if the first argument string (`.`) contains the second argument string (`Xerox`). The dot (`.`) in the first argument position indicates the current string.

For more information about XPath, see the following resources:

- XPath syntax: [https://www.w3schools.com/xml/xpath\\_syntax.asp](https://www.w3schools.com/xml/xpath_syntax.asp)
- XPath functions: <https://developer.mozilla.org/en-US/docs/Web/XPath/Functions>

## Configure Scan Settings

Use the **Configure Scan Settings** dialog to configure the system scanner. As part of a management set update operation, the scanner uses various network and application-layer protocols to probe each system it finds. The scanner writes response details in XML format to a scan results file. Depending on configuration, the scanner uses Ping, SMB, SSH, Telnet, SNMP, IPMI, and other protocols to determine if a system matches a known type (for example, Linux/Unix, Windows, IPMI, SNMP, SQL Server, or Oracle database). If the **Enable access checking** option is selected, the scanner can test if it can access each system using known credentials.

To open the "Configure Scan Settings" dialog, click **Configure** next to the **Scan Settings** box on the "IP Scanner Target Type and Name Lookup Options" dialog.

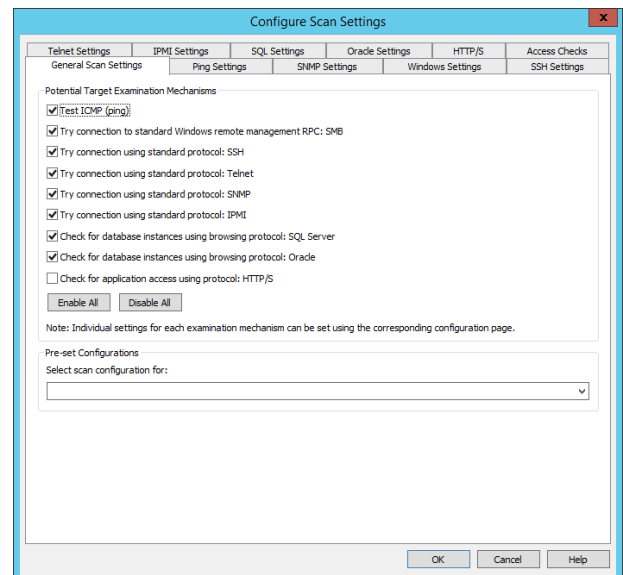
### General Scan Settings

The General Scan Settings tab is used to configure which mechanisms will be used during the scan to collect results that will be parsed via the dynamic mapping settings.

Configure the connectivity options to perform your scan with. The more items selected, the slower the scan will be and the more configuration will be required. It is recommended to use only the methods relevant to the targets you are attempting to find. For example, you do not need to include Telnet when searching only for Windows machines.

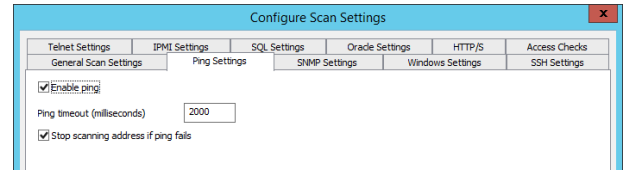
- **Test ICMP (ping)** - Select to use ping to test if a host is reachable at an IP address.
- **Try connection to standard Windows remote management RPC: SMB** - Select to try to connect to Windows hosts using the Microsoft Server Message Block protocol.
- **Try connection using standard protocol: SSH** - Select to try to connect to hosts using the Secure Shell network protocol.
- **Try connection using standard protocol: Telnet** - Select to try to connect to hosts using the Telnet application layer protocol.
- **Try connection using standard protocol: SNMP** - Select to try to connect to managed devices using the Simple Network Management Protocol.
- **Try connection using standard protocol: IPMI** - Select to try to connect to hosts using the Intelligent Platform Management Interface protocol.
- **Check for database instances using browsing protocol: SQL Server** - Select to try to connect to SQL Server database instances.
- **Check for database instances using browsing protocol: Oracle** - Select to try to connect to Oracle database instances.
- **Check for application access using protocol: HTTP/S** - Select to try to connect to hosts that present a web interface.
- **Enable All** - Click to select all of the **Potential Target Examination Mechanisms** option boxes.
- **Disable All** - Click to clear all of the **Potential Target Examination Mechanisms** option boxes.

Use the Pre-set Configurations dropdown list to quickly configure this dialog. Choose from a menu of pre-configured scan settings: **Scan for Windows (only)** and **Scan for SSH (only)**.



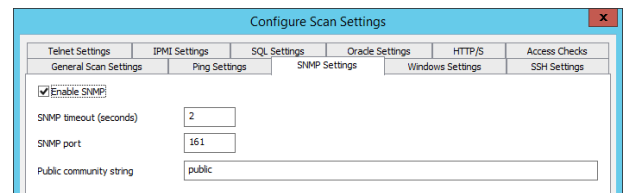
## Ping Settings

- **Enable ping** - Select to use ping to test if a host is reachable at an IP address. Changing this option also updates the **Test ICMP (ping)** setting on the **General Scan Settings** tab.
- **Ping timeout** - Enter the period of time that ping should wait for a response before the request times out. The default value is 2000 milliseconds.
- **Stop scanning address if ping fails** - Select this option if the scanner should abort further scanning of an address if it does not receive a ping response. (The scanner skips to the next address.) Clear this option if the scanner should proceed with additional scan tests if the ping test does not receive a response.



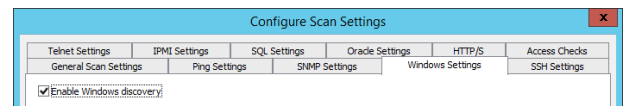
## SNMP Settings

- **Enable SNMP** - Select to scan for managed devices using the Simple Network Management Protocol. Changing this option also updates the **Try connection using standard protocol:SNMP** setting on the **General Scan Settings** tab.
- **SNMP timeout** - Enter the number of seconds that the scanner should wait before the SNMP connection attempt times out. The standard value is 2 seconds.
- **SNMP port** - Enter the port number to use to attempt the SNMP connection. The standard value is 161.
- **Public community string** - Enter the public community string to include in the request to allow access to the device. Most network vendors use the default value public.



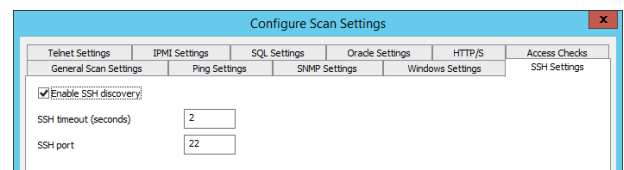
## Windows Settings

- **Enable Windows discovery** - Select to try to connect to Windows hosts using the Microsoft Server Message Block protocol. Changing this option also updates the **Try connection to standard Windows remote management RPC: SMB** setting on the **General Scan Settings** tab.



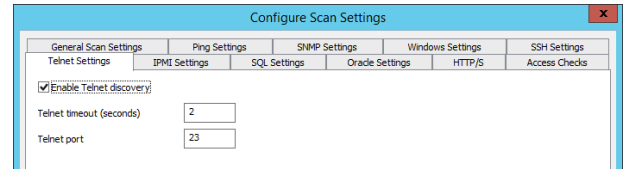
## SSH Settings

- **Enable SSH discovery** - Select to try to connect to hosts using the Secure Shell network protocol. Changing this option also updates the **Try connection using standard protocol: SSH** setting on the **General Scan Settings** tab.
- **SSH timeout** - Enter the number of seconds that the scanner should wait before the SSH connection attempt times out. The standard value is 2 seconds.
- **SSH port** - Enter the port number to use to attempt the SSH connection. The standard value is 22.



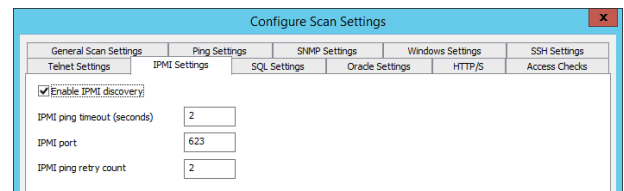
## Telnet Settings

- **Enable Telnet discovery** - Select to try to connect to hosts using the Telnet application layer protocol. Changing this option also updates the **Try connection using standard protocol: Telnet** setting on the **General Scan Settings** tab.
- **Telnet timeout** - Enter the number of seconds that the scanner should wait before the Telnet connection attempt times out. The standard value is 2 seconds.
- **Telnet port** - Enter the port number to use to attempt the SSH connection. The standard value is 23.



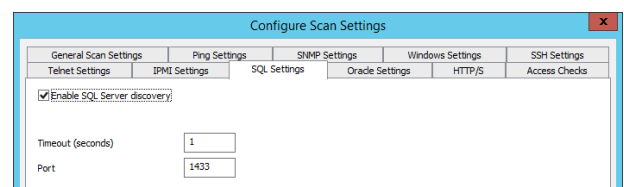
## IPMI Settings

- **Enable IPMI discovery** - Select to try to connect to hosts using the Intelligent Platform Management Interface protocol. Changing this option also updates the **Try connection using standard protocol: IPMI** setting on the **General Scan Settings** tab.
- **IPMI ping timeout** - Enter the number of seconds that IPMI ping should wait for a response before the request times out. The standard value is 1 second.
- **IPMI port** - Enter the port number to use to attempt the IPMI connection. The standard value is 623.
- **IPMI ping retry count** - Enter the number of times that ping should reattempt to connect to a host that timed out.



## SQL Settings

- **Enable SQL Server discovery** - Select to try to connect to SQL Server database instances using the SQL Server Browser Service. Changing this option also updates the **Check for database instances using browsing protocol: SQL Server** setting on the **General Scan Settings** tab.



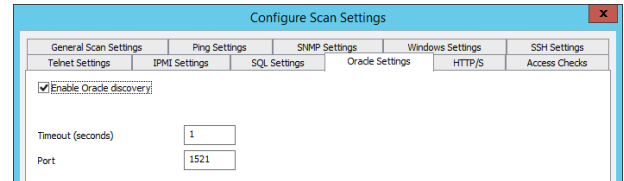

**Note:** You can also search registry data on Windows targets for instances of SQL Server by enabling an option on the "IP Scanner Target Type and Name Lookup Options" dialog. In the "Special Cases" section, select this option: **For Windows targets, examine the systems if possible for instances of SQL Server.**

- **Timeout** - Enter the number of seconds that the scanner should wait before the SQL Server database connection attempt times out. The standard value is 1 second.
- **Port** - Enter the port number to use to attempt the SQL Server connection. The standard value is 1433.



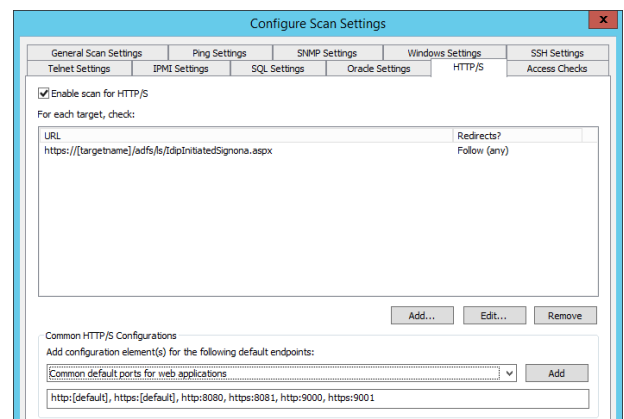
## Oracle Settings

- **Enable Oracle discovery** - Select to try to connect to Oracle database instances. Changing this option also updates the **Check for database instances using browsing protocol: Oracle** setting on the **General Scan Settings** tab.
- **Timeout** - Enter the number of seconds that the scanner should wait before the Oracle database connection attempt times out. The standard value is 1 second.
- **Port** - Enter the port number to use to attempt the Oracle database connection. The standard value is 1521.



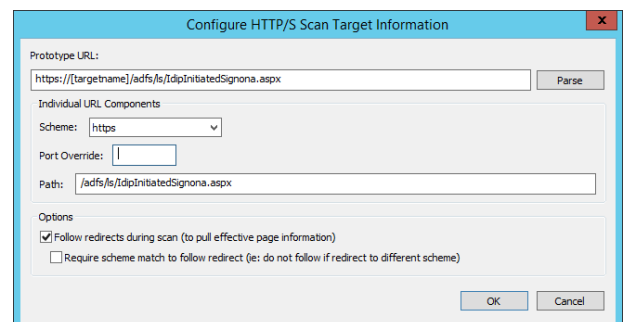
## HTTP/S Settings

- **Enable scan for HTTP/S** - Select to try to connect to hosts that present a web interface. Changing this option also updates the **Check for application access using protocol: HTTP/S** setting on the **General Scan Settings** tab.
- **For each target check** - Table lists URL schemes that have been added to the scanner.
  - **Add** - Click to open the "Configure HTTP/S Scan Target Information" dialog and add an HTTP or HTTPS endpoint to the scan table.
  - **Edit** - Click to open the selected endpoint for editing.
  - **Remove** - Click to delete the selected endpoint from the scan table.
- **Common HTTP/S Configurations** - Choose from the menu to add common web interface URLs to the scan table. To customize a URL, select it and click **Edit**.
  - **Add configuration element(s) for the following default endpoints** - To quickly add one or more URL schemes to the scan table, choose from the dropdown menu and click **Add**.



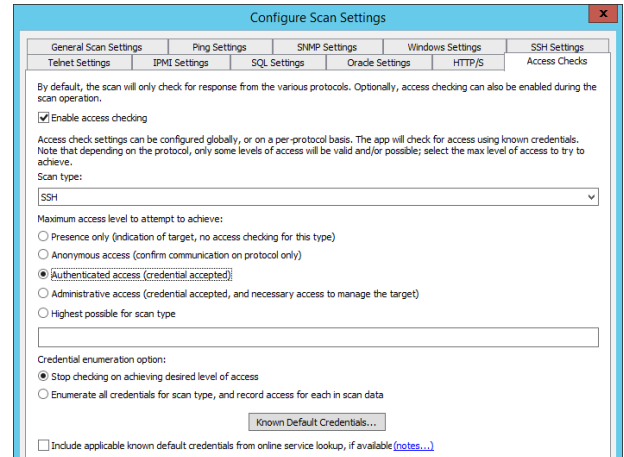
When adding or editing a new HTTP/S elements, click **Add** or **Edit**. Complete this dialog to add or modify an endpoint URL in the scan table.

- **Prototype URL** - Type the URI scheme of the endpoint to test.
- **Parse** - Click to add values to the "Individual URL Components" fields based on the URL entered in the **Prototype URL** field.
- **Scheme** - Specify **http** or **https**.
- **Port Override** - If required, enter a custom port number for the endpoint.
- **Path** - Enter the resource portion of the URL.
- **Options**
  - **Follow redirects during scan** - Select if the scanner should follow page redirects.
  - **Require scheme match to follow redirect** - Select to prevent the scanner from following a redirect that uses a different URI scheme than the one specified.



## Access Checks

- **Enable access checking** - Select if the scanner should check if the target system can be accessed using known credentials. Results are recorded in the scan results file. If this option is cleared, the scanner only probes for the presence of targets using the selected protocols and does not check access.
- **Scan type** - Choose the protocol for which you are configuring the following settings: Maximum access level to attempt to achieve and Credential enumeration option.
- **Maximum access level to attempt to achieve** - For the selected Scan type, choose from the following:
  - **Presence only** - Tests if the target is present using ping.
  - **Anonymous access** - Tests if the target is present and communicating on the selected protocol.
  - **Authenticated access** - Tests if the target system will accept a credential from the "Known Credentials" list.
  - **Administrative access** - Tests if a credential from the "Known Credentials" list has administrator privileges on the target system.
- **Credential enumeration option** - For the selected Scan type, choose from the following:
  - **Stop checking on achieving desired level of access** - Upon successfully achieving the access level configured above, stop checking credentials from the "Known Credentials" list and move to the next host.
  - **Enumerate all credentials for scan type and record access for each in scan data** - Check all applicable credentials on the "Known Credentials" list before moving to the next host. Record access for each in the scan data.
- **Known Default Credentials** - Click to open the "Configure Known Credentials" dialog. See ["Configure Known Credentials" on page 95](#) for details.



## Configure Known Credentials

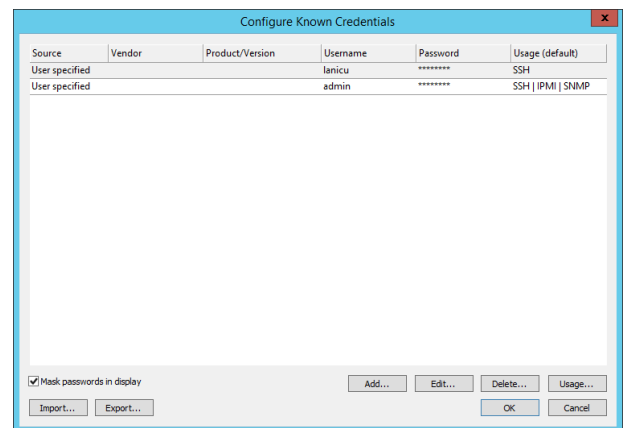
Configure Known Credentials to define the list of credentials that the system scanner uses to perform access checking. If access checking is enabled, the scanner uses this credential list to test if the target system will accept the supplied credential. For details, see "[Configure Scan Settings](#)" on page 90.

There are two ways to open this dialog:

- In the console, choose **Settings | Known Default Credentials**.
- Open the "Configure Scan Settings" dialog, click the Access Checks tab, and click **Known Default Credentials**.

This dialog lists the following information:

- **Source** - Shows the source of the credential.
- **Vendor** - Shows the value entered in the Product Vendor field
- **Product/Version** - May indicate a specific product and version for which the credential is valid, or may provide a general description of the scope where the credential is valid.
- **Username** - User name portion of the credential.
- **Password** - Password portion of the credential.
- **Usage** - Indicates the access mechanisms for which the credential should be tried. See Known Credential Usage later in this topic for more information.



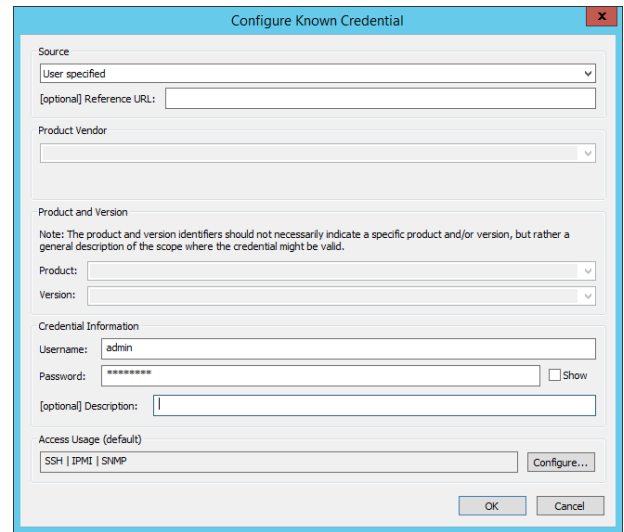
Buttons and Options on this dialog include:

- **Mask passwords in display** - Select to mask passwords displayed in this dialog with asterisks.
- **Import** - Click to import credential data in XML format from an export file.
- **Export** - Click to save credential data in XML format to an export file.
- **Add** - Click to open the "Configure Known Credential" dialog to add a new credential.
- **Edit** - Click to select a record in the table, then click **Edit** to make modifications.
- **Delete** - Click to delete the selected credential.
- **Usage** - Click to open the "Known Credential Usage" dialog and select the protocol(s) / access mechanism(s) for which the credential is valid.

Click **Add** to add a new credential or **Edit** to edit an existing credential entry.

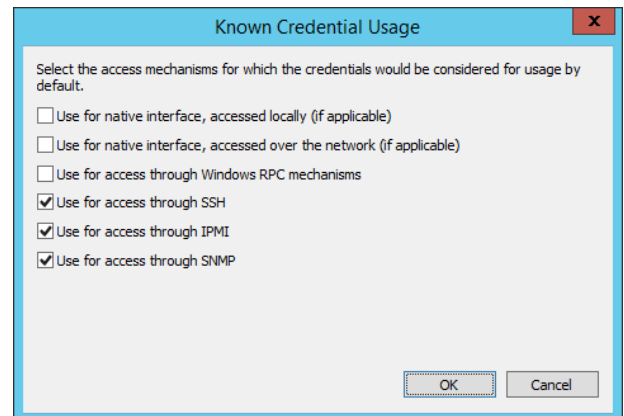
When configuring a known credential, the following items must be specified:

- **Source** - Choose from the following:
  - **User specified** - Enter credentials that are used in your organization.
  - **Reference list** - Enter credentials for system and device accounts.
  - **Reference URL** - Enter a link to an address that contains information relevant to this credential record.
- **Product Vendor** - (Optional) Enter a manufacturer name (or similar) that you can use to group product-related credentials.
- **Product** - (Optional) Specify either a specific product for which the credential is valid, or provide a general description of the scope where the credential is valid.
- **Version** - (Optional) Specify either a product version or a range of versions for which the credential is valid, or provide a similar value.
- **Username** - Enter the user name portion of the credential.
- **Password** - Enter the password portion of the credential.
- **Description** - (Optional) Enter a brief note or description of the credential to help you or another administrator identify it later.
- **Access Usage** - Click the **Configure** button to configure access usage. This setting determines what access mechanisms are valid for the credential (i.e. use this credential for SSH connections).



When configuring access usage, these protocol options determine if/when the credential will be used. As depicted in the screen shot, the credential would only be used for connection attempts made via SSH, IPMI, or SNMP.

- **Use for native interface, accessed locally** - This option is for future use and not supported in this release. The intended use case is for platforms/services/middleware/etc. that when included in the scanner in the future provides app specific providers and methods.
- **Use for native interface, accessed over the network** - This option is for future use and not supported in this release. The intended use case is for platforms/services/middleware/etc. that when included in the scanner in the future provides app specific providers and methods.
- **Use for access through Windows RPC mechanisms** - Select to use the credential on computers and systems that are identified using Microsoft Remote Procedure Call.
- **Use for access through SSH** - Select to use the credential on computers and systems that are identified using the secure shell (SSH) protocol.
- **Use for access through IPMI** - Select to use the credential on systems and devices that are identified using the Intelligent Platform Management Interface (IPMI) protocol.
- **Use for access through SNMP** - Select to use the credential on network devices that are identified using the Simple Network Management Protocol (SNMP) such as servers, printers, hubs, switches, routers, and other network devices.



## Enroll Linux, Unix, Mainframe and Related Systems

Any non-Windows operating system can be managed as long as it can be accessed via SSH or Telnet. Linux, Unix, and mainframe systems have explicit nodes laid out in the account store. These nodes have names indicating their recommended use, but the nodes are not type-definitive. If you need to manage a device or OS type that is not expressly mentioned, you can add it to almost any node that supports an SSH or Telnet option, such as the Linux/Unix node.



**Note:** You may need to update a response file to support a non-standard platform. Response files come into play when it is time to start managing passwords on these systems.



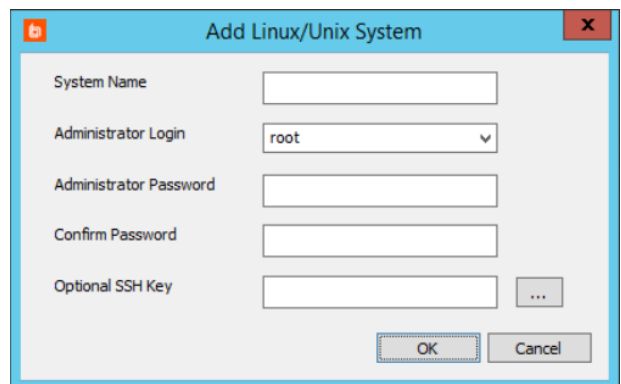
For more information on response files, see *"About Response Files"* on page 246.

## Enroll AIX

AIX does not have its own unique node. Use the **Linux/Unix Systems** node.

### To Enroll an AIX System

- Via API:
  - With PowerShell: **New-LSSystemInManagementSetLinux**.
  - Web service URI: **ManagementSetOps\_AddLinuxSystemToManagementSet**.
  - Web service REST: **ManagementSet/System/Linux** as a POST.
- Use any of the dynamic discovery options such as LDAP directories from the **Management Set Properties** dialog.




Please see *"Create Management Sets"* on page 64 for more information on adding Linux/Unix-based systems.

- Right-click on the **Linux/Unix System** node and click **Import Systems from Text File** to import from a line delimited file.
- Right-click on the **Linux/Unix Systems** node and select **Add Linux system**.
  - **System Name:** (Required) The name or IP address of the system.
  - **Administrator Login:** Select or enter the admin account.
  - **Administrator Password:** Enter the password for the designated admin account.
  - **Optional SSH Key:** If SSH keys have already been added, select which to use.

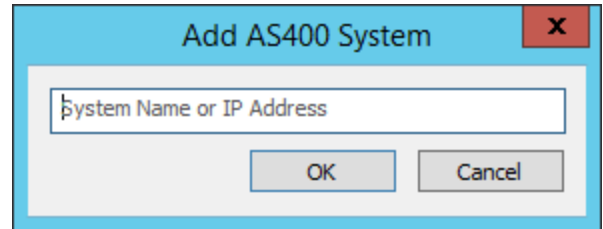
Now that the device is added, it can be managed.

## Enroll AS400

Use the **AS400 Systems** node to add AS400 systems.

### To Enroll an AS400 System

- Via API:
  - With PowerShell: **New-LSSystemInManagementSetAS400**.
  - Web service SOAP: **ManagementSetOps\_AddAS400SystemToManagementSet**.
  - Web service REST: **ManagementSet/System/AS400** as a POST.
- Right-click on the **AS400 Systems** node and import from a text file. Each system should be on its own line.
- Right-click on the **AS400 Systems** node and select **Add AS400** system. Enter the name or IP address of the system.



Now that the device is added, it can be managed.

Alternatively, AS400 systems can be added via any of the dynamic discovery options and then have their system type set after being added. To change a system type, right-click and select **Set System Type**. Choose the appropriate value from the list.



#### IMPORTANT!

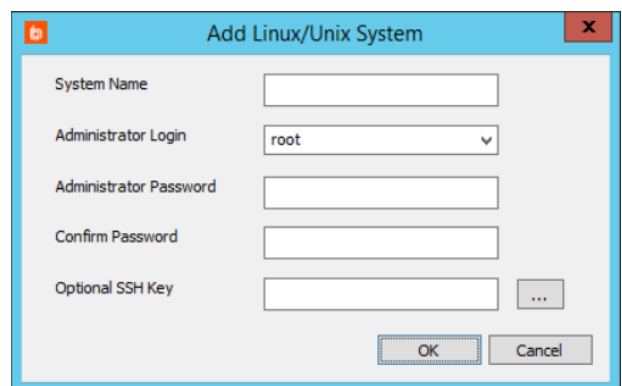
Managing systems using a 3270 or 5250 terminal type also requires separate purchase and installation of Quick3270 or Quick3270 Secure, available from DN-Computing at [www.dn-computing.com](http://www.dn-computing.com).

## Enroll Linux, Unix and Solaris

Use the **Linux/Unix Systems** node to add any Linux or Unix like operating system.

### To Enroll a Linux, Unix, or Solaris System

- Via API:
  - With PowerShell: **New-LSSystemInManagementSetLinux**.
  - Web service URI: **ManagementSetOps\_AddLinuxSystemToManagementSet**.
  - Web service REST: **ManagementSet/System/Linux** as a POST.
- Use any of the dynamic discovery options such as LDAP directories from the **Management Set Properties** dialog.




**i** Please see "*Create Management Sets*" on page 64 for more information on adding Linux/Unix-based systems.

- Right-click on the **Linux/Unix System** node and click **Import Systems from Text File** to import from a line delimited file.
- Right-click on the **Linux/Unix Systems** node and select **Add Linux system**.
  - **System Name:** (Required) The name or IP address of the system.
  - **Administrator Login:** Select or enter the admin account.
  - **Administrator Password:** Enter the password for the designated admin account.
  - **Optional SSH Key:** If SSH keys have already been added, select which to use.

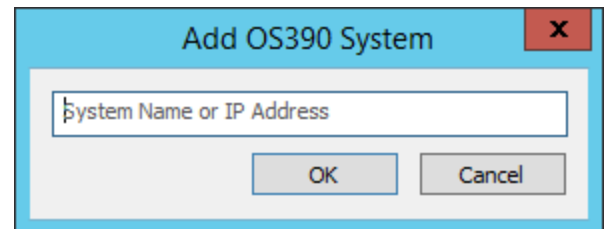
Now that the device is added, it can be managed.

## Enroll OpenVMS

 **Note:** OpenVMS does not have its own unique node. Instead, create a custom account store, using the **OS/390 Mainframes** response file as the template to create a custom OpenVMS response file. The following section describes adding the OpenVMS system as an OS/390 system.

### To Enroll an OpenVMS System as an OS/390 System

- Via API:
  - With PowerShell: **New-LSSystemInManagementSetOS390**.
  - Web service SOAP: **ManagementSetOps\_AddOS390SystemToManagementSet**.
  - Web service REST: **ManagementSet/System/OS390** as a POST.
- Right-click on the **OS/390 Mainframes** node and import from a line delimited text file.
- Right-click on the **OS/390 Mainframes** node and select **Add OS/390 Mainframe**. Enter the name or IP address of the system.



The screenshot shows a dialog box titled "Add OS390 System" with a close button (X) in the top right corner. Inside the dialog, there is a text input field labeled "System Name or IP Address". Below the input field are two buttons: "OK" and "Cancel".

Now that the device is added, it can be managed.

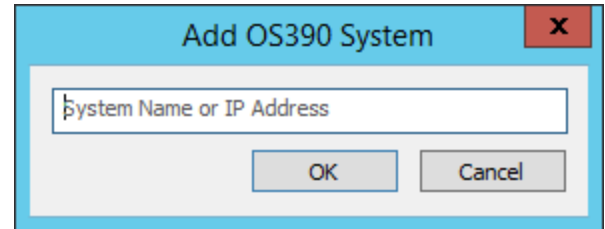
Alternatively, OS/390 systems can be added via any of the dynamic discovery options and then have their system type set after being added. To change a system type, right-click on the system(s) and select **Set System Type**. Choose the appropriate value from the list.

## Enroll OS/390

Use the **OS/390 Mainframes** node to add an OS/390 or z/OS system.

## To Enroll an OpenVMS System as an OS/390 System

- Via API:
  - With PowerShell: **New-LSSystemInManagementSetOS390**.
  - Web service SOAP: **ManagementSetOps\_AddOS390SystemToManagementSet**.
  - Web service REST: **ManagementSet/System/OS390** as a POST.
- Right-click on the **OS/390 Mainframes** node and import from a line delimited text file.
- Right-click on the **OS/390 Mainframes** node and select **Add OS/390 Mainframe**. Enter the name or IP address of the system.



Now that the device is added, it can be managed.

Alternatively, OS/390 systems can be added via any of the dynamic discovery options and then have their system type set after being added. To change a system type, right-click on the system(s) and select **Set System Type**. Choose the appropriate value from the list.

### IMPORTANT!

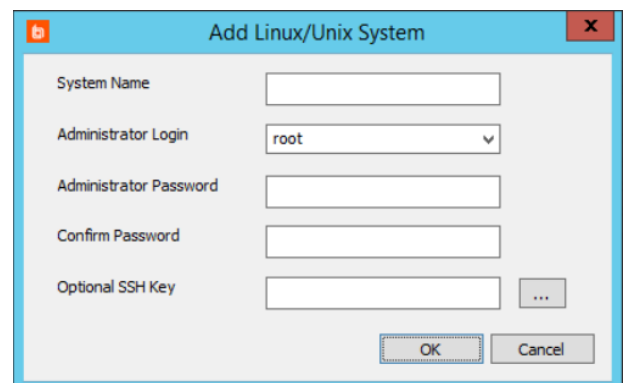
Managing systems using a 3270 or 5250 terminal type also requires separate purchase and installation of Quick3270 or Quick3270 Secure, available from DN-Computing at [www.dn-computing.com](http://www.dn-computing.com).

## Enroll OSX

Use the **Linux/Unix Systems** node to add an OSX system.

## To Enroll an OSX System

- Via API:
  - With PowerShell: **New-LSSystemInManagementSetLinux**.
  - Web service URI: **ManagementSetOps\_AddLinuxSystemToManagementSet**.
  - Web service REST: **ManagementSet/System/Linux** as a POST.
- Use any of the dynamic discovery options such as LDAP directories from the **Management Set Properties** dialog.



 Please see *"Create Management Sets"* on page 64 for more information on adding Linux/Unix-based systems.



- Right-click on the **Linux/Unix System** node and click **Import Systems from Text File** to import from a line delimited file.
- Right-click on the **Linux/Unix Systems** node and select **Add Linux system**.
  - **System Name:** (Required) The name or IP address of the system.
  - **Administrator Login:** Select or enter the admin account.
  - **Administrator Password:** Enter the password for the designated admin account.
  - **Optional SSH Key:** If SSH keys have already been added, select which to use.

Now that the device is added, it can be managed.

## Enroll TN3270 or TN5250 Terminal-Based Systems



**Note:** Historically, any system that required the use of a 3270 or 5250 terminal would be placed under the **TN3270** node. Since version 5.0.1, the functions supporting the use of these terminal types and their related options (such as code page and SSL) have been added and are fully supported under the other proper node types, such as OS/390 or AS400. BeyondTrust recommends using these nodes to add these systems, instead.

If you want to use the **TN3270** node, use the **3270Response.xml** file in the **AnswerFiles** directory. If the logon procedure has been customized or the default settings do not meet your needs, you must modify the answer file. The **TN3270** node is hidden by default; show it using the display options.



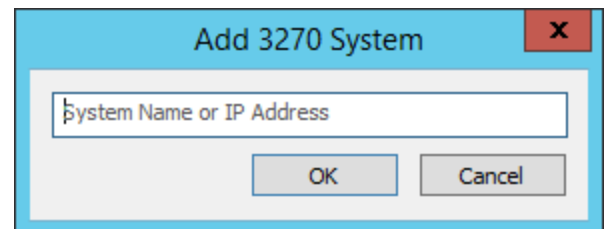
For more information on modifying answer files, see "[About Response Files](#)" on page 246.

### To Enroll a TN3270 System

- Right-click on the **TN3270 Systems** node and import from a text file. Each system should be on its own line.
- Right-click on the **TN3270 Systems** node and select **Add TN3270 system**. Enter the name or IP address of the system.

Now that the device is added, it can be managed.

Alternatively, TN3270 systems can be added via any of the dynamic discovery options and then have their system type set after being added. To change a system type, right-click and select **Set System Type**. Choose the appropriate value from the list.




### IMPORTANT!

Managing systems using a 3270 or 5250 terminal type also requires separate purchase and installation of Quick3270 or Quick3270 Secure, available from DN-Computing at [www.dn-computing.com](http://www.dn-computing.com).

## Enroll Tandem Realtime Systems

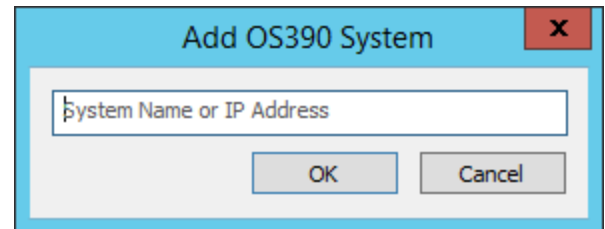


**Note:** Tandem Realtime Systems does not have its own unique node. Instead, create a custom account store, using the **OS/390 Mainframes** response file as the template to create a custom Tandem Realtime Systems response file. The following section describes adding the Tandem system as an OS/390 system.

Privileged Identity can manage Tandem systems via VT100, TN3270, TN5250, or native SSH with no emulation requirement. Currently, Privileged Identity does not support the Tandem 6530 terminal type.

### To Enroll a Tandem Realtime System as an OS/390 System

- Via API:
  - With PowerShell: **New-LSSystemInManagementSetOS390**.
  - Web service SOAP: **ManagementSetOps\_AddOS390SystemToManagementSet**.
  - Web service REST: **ManagementSet/System/OS390** as a POST.
- Right-click on the **OS/390 Mainframes** node and import from a line delimited text file.
- Right-click on the **OS/390 Mainframes** node and select **Add OS/390 Mainframe**. Enter the name or IP address of the system.



Now that the device is added, it can be managed.

Alternatively, OS/390 systems can be added via any of the dynamic discovery options and then have their system type set after being added. To change a system type, right-click on the system(s) and select **Set System Type**. Choose the appropriate value from the list.

## Enroll VMware ESX

There are two ways to remotely administer a VMware ESX host when using Privileged Identity:

- Native APIs
- SSH

Native APIs are the preferred mechanism to use when managing an ESX host as it requires no additional host configuration. When using SSH connections to manage an ESX host, certain functionality may not be available such as account discovery, VM guest discovery and more.

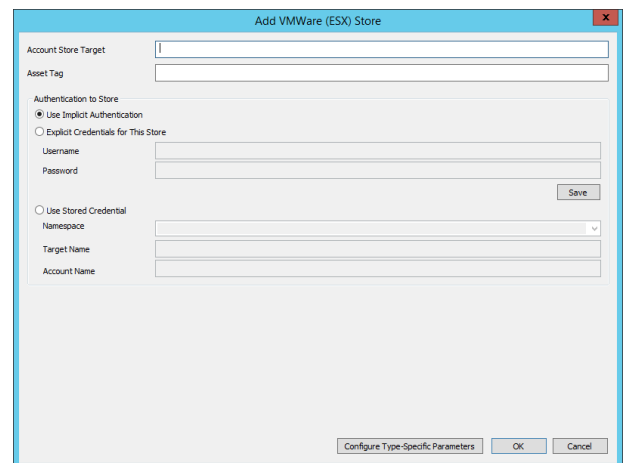
### To Enroll an ESX Host for Management Using VMware APIs

- With PowerShell: **New-LSSystemInManagementSetCustom**.
- Web service SOAP: **ManagementSetOps\_AddCustomInstanceToManagementSet**.
- Web service REST: **ManagementSet/System/Custom** as POST.
- Right-click on the **VMWare (ESX)** node and import from a comma-delimited text file. Each system should be on its own line.
- Right-click on the **VMWare (ESX)** node and select **Add VMWare ESX system**. The required elements are:
  - **Name or IP address** - The name or IP address of the system

Supply the following information:

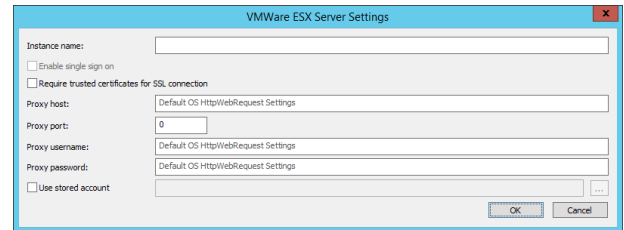
- **Account store target** - this is the name or IP of the host as can be resolved from the Privileged Identity host.
- **Asset tag** - (Optional) Add an asset tag for the system.
- **Authentication to Store** - identifies the credentials that Privileged Identity will use when attempting to authenticate to the target:
  - **Use Implicit Authentication** - use this option if the target will trust the deferred processor service account login. This requires a fully trusted domain scenario.
  - **Explicit Credentials for This Store** - supply a username and password. This information will be added to the secure password store and associated with this system. The next time the dialog re-opens, the Use Stored Credential fields will be populated with this information instead. Credentials added this way will be explicitly associated with the target server.
  - **Use Stored Credential** - supply the target account name and host system that has already been stored/managed in Privileged Identity. This option is useful when the local admin credential has already been stored/managed or when it is possible to use directory-based account for authentication.

When adding through the console, additional parameters may be specified when adding or editing the system. Click **Configure Type-Specific Parameters** to show the additional settings.



The following options are all optional:

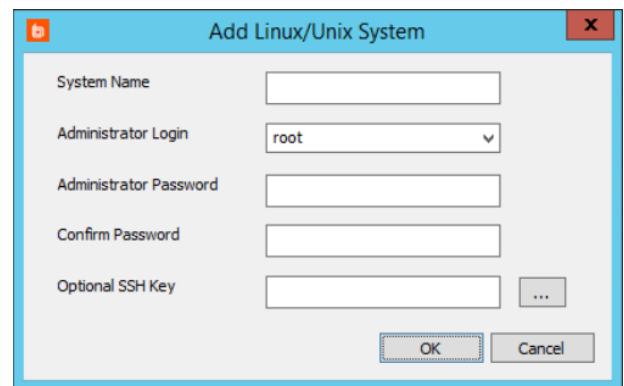
- **Enable single sign on** - not available for ESX.
- **Require trusted certificates for SSL connection** - when enabled, the Privileged Identity host performing the activity must trust the certificate presented by the target ESX host. If the certificate is considered invalid, management functions will fail.
- **Proxy host** - if a proxy server must be access prior to managing the target system, enter that proxy server information (IP/Name, port, username and password) as required.
- **Use stored account** - if the proxy server requires credentials, you may configure a static username and password by filling out the proxy username and password field or, if a valid proxy account is already being managed, use those stored managed credentials to authenticate to the proxy.



## To Enroll an ESX Host for Management Using SSH

VMware ESX, when managed using SSH, does not have its own unique node. Use the **Linux/Unix Systems** node.

- Via API:
  - With PowerShell: **New-LSSystemInManagementSetLinux**.
  - Web service URI: **ManagementSetOps\_AddLinuxSystemToManagementSet**.
  - Web service REST: **ManagementSet/System/Linux** as a POST.
- Use any of the dynamic discovery options such as LDAP directories from the **Management Set Properties** dialog.



**i** Please see *"Create Management Sets"* on page 64 for more information on adding Linux/Unix-based systems.

- Right-click on the **Linux/Unix System** node and click **Import Systems from Text File** to import from a line delimited file.
- Right-click on the **Linux/Unix Systems** node and select **Add Linux system**.
  - **System Name:** (Required) The name or IP address of the system.
  - **Administrator Login:** Select or enter the admin account.
  - **Administrator Password:** Enter the password for the designated admin account.
  - **Optional SSH Key:** If SSH keys have already been added, select which to use.

Now that the device is added, it can be managed.

**Note:** Do not enable Lockdown mode. Doing so blocks all connections and will prevent Privileged Identity from managing the host. Only the vCenter Server can manage the host when Lockdown mode is enabled.

## Enroll Windows Systems

Privileged Identity can manage Windows systems from Windows NT4 through the latest versions of Windows Server and Windows Workstation.



### IMPORTANT!

*If managing domain accounts, create a separate management set that includes the "domain." Specifically, if the DNS name of the domain is "example.int", then create a new management set for just this domain object using this name. Using this method, the Privileged Identity host will use DNS to locate the nearest domain controller to perform a password change job against. As such, managed domain accounts will always be associated with the domain rather than a domain controller. This approach is recommended because domain controller systems may come and go, but the domain will persist and the managed accounts will always be available for recovery. If this approach is not used and you target a specific domain controller, when that domain controller is retired and removed, the accounts will be removed with it and the jobs targeting the domain controller will simply fail.*

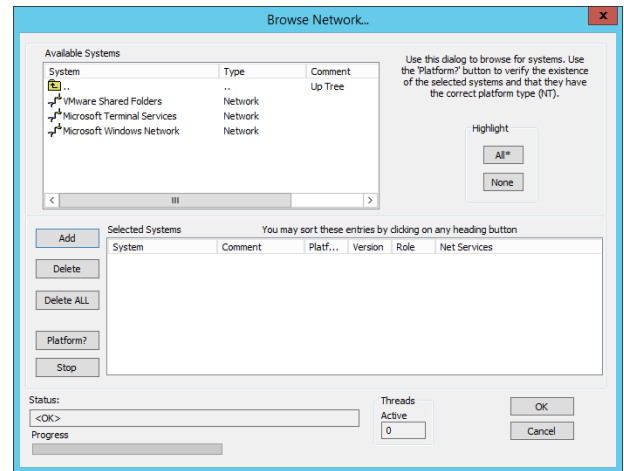
## To Enroll a Windows System

- PowerShell: **New-LSSystemInManagementSetWindows**.
- Web service SOAP: **ManagementSetOps\_AddWindowsSystemToManagementSet**.
- Web service REST: **ManagementSet/System/Windows** as POST.
- Use any of the dynamic discovery options such as Active Directory Paths from the management set properties dialog. There are many options for adding Windows systems dynamically and manually. Please see [Creating a Management Set](#) for more information.
- Right-click on the **Windows Systems** node and import from a comma-delimited text file. Each system should be on its own line.
- Right-click on the **Windows Systems** node and select **Add Windows system**. When choosing this method, there will be four sub-options:
  - Add from Network Browse
  - Add from Domain List
  - Add from Active Directory
  - Add from Manual Entry

## Manual Network Browse

If the Computer Browsers service is running on the host system, this dialog will be able to find other machines visible via the browser service. Any system where the Computer Browser service is not running, will not show in this list.

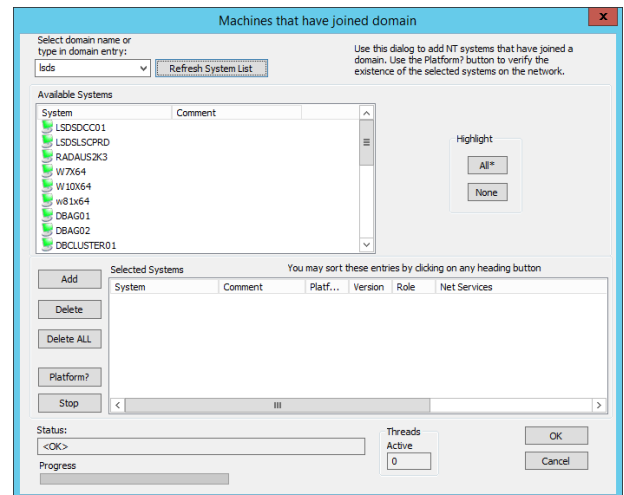
1. Browse through the folder structure to find systems.
2. Select the systems you wish to add and click **Add**.
3. Use the **Platform?** button to attempt a quick basic connection to the target systems as the current interactive user.
4. Once the desired systems have been added, click **OK** and the systems will be added to the Windows Systems node in the current management set.



## Manual Domain List

The domain list uses the Windows NT style picker and APIs to find systems that have joined the target trusted domain. There is no filter for this list as the underlying Windows API does not support it; beware of large domains when using this feature.

1. Select the systems you wish to add and click **Add**.
2. Use the **Platform?** button to attempt a quick basic connection to the target systems as the current interactive user.
3. Once the desired systems have been added, click **OK** and the systems will be added to the Windows Systems node in the current management set.



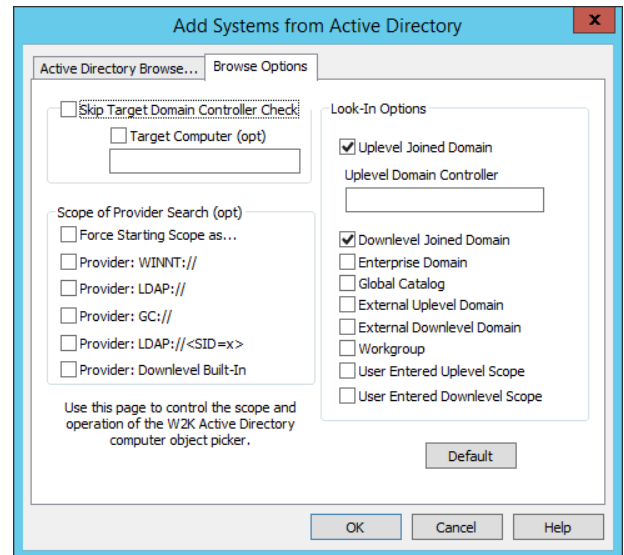
## Manual Active Directory

The manual options to browse Active Directory provide a lot of functionality and do not require full domain privileges as the Domain Browse feature does. Typically, the default options need not be changed.

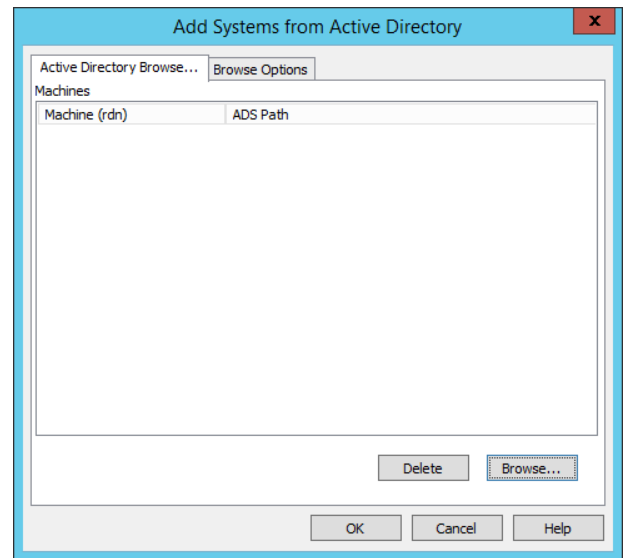
1. To use the manual Active Directory picker option start on the Browse Options tab.

2. Configure any options needed/desired to help perform the query:

- **Target Computer:** These options allow controlling where searches are to be performed. Normally these options should be ignored. Use these options to extract machine lists from foreign/non-Active Directory domains.
  - **Skip Target Domain Controller Check** - Set this flag if the computer is not a domain controller, to save time. However, if the machine is a domain controller, this flag would not typically be set. It is usually best to select domain objects from the domain scope rather than from the domain controller itself.
  - **Target Computer** - (Optional) Allows specifying where to execute the search via the text entry field below the check box. Set the check box and set the field to a non-Active Directory domain controller to see a list of machines that have joined that domain (The "Skip Target Domain Controller Check" should be unchecked in this scenario). If the "Target Computer" entry field is blank, the current machine is the target computer.
- **Scope of Provider Search:** These options allow controlling which data source is to be used for the machine search. Generally, leave all of these options unchecked.
  - **Force Starting Scope as...** - Sets the first entry in the "Look in" dropdown to the option selection. Normally the dropdown will default to its own choice.
  - **Provider...** - These options are different data sources for searches.
- **Look-In Options:** These options define the scope of the search.
  - **Up level Joined Domain** - Search the up level domain to which the target computer is joined. If this flag is set, use the "Up level Domain Controller" entry field to specify the name of a domain controller in the joined domain.
  - **Up level Domain Controller Field** - This field can be blank even if the "Up level Joined Domain" is checked, in which case, the dialog box looks up the domain controller. This entry field enables specifying a domain controller in a multi-master domain. For example, an administrative application might make changes on a domain controller in a multi-master domain, and then open the object picker dialog box before the changes have been replicated on the other domain controllers.
  - **Down level Joined Domain** - Search the down level domain to which the UMP host computer is joined.
  - **Enterprise Domain** - Search all Active Directory domains in the enterprise to which the target computer belongs. If the Up level Joined Domain check box is set, then the results represent all Active Directory domains in the enterprise except the joined domain.
  - **External Up level Domain** - Search all up level domains external to the enterprise but trusted by the domain to which the target computer is joined.
  - **External Down level Domain** - Search all down level domains external to the enterprise but trusted by the domain to which the target computer is joined.
  - **Workgroup** - Search the workgroup to which the target computer is joined. Applies only if the target computer is not joined to a domain.
  - **User Entered Up level Scope** - Enables entry of an up level scope. If neither of the "USER ENTERED..." types is specified, the dialog box restricts the query to the scopes in the "Look in" dropdown list.
  - **User Entered Down level Scope** - Enables entering a down level scope.



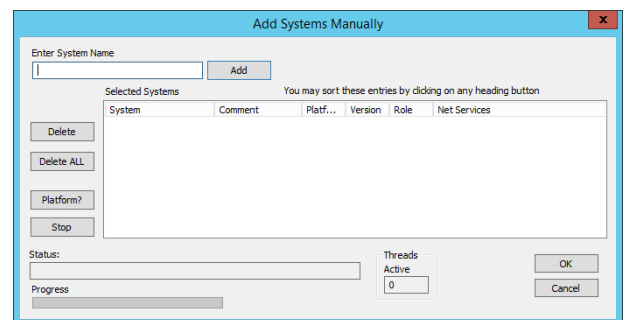
3. Next select the Active Directory Browse tab.
4. Click the **Browse** button in the lower right corner to bring up the active directory search dialog. Type in the [partial] name(s) of the computers to add or use the advanced find to locate system in Active Directory. This is the same dialog as is presented by Windows when searching Active Directory.
5. Select the systems you wish to add and click **Add**. The systems will be added to the Active Directory Browse tab and their RDN and path in Active Directory will be filled out.
6. Once done, click **OK** to add the systems on this tab to the current management set.



## Manual Entry

Manual entries are useful when it is only needed to add one system that may not be joined to AD or must be added in a certain way that cannot be directly resolved through other dynamic processes. For example, it is recommended to add your target Active Directory (when managing domain accounts) as the domain's FQDN rather than by adding a specific domain controller.

1. Type in the name of the system to add and click **Add**.
2. Use the **Platform?** button to attempt a quick basic connection to the target systems as the current interactive user.
3. Once the desired systems have been added, click **OK** and the systems will be added to the Windows Systems node in the current management set.





## Enroll Databases

Privileged Identity supports changing passwords for the explicit accounts hosted by a database such as SA in Microsoft SQL or System in an Oracle database. This section explains how to enroll various database types to discover and manage the database local accounts.

### Enroll IBM DB2

If the IBM DB2 node is not displayed in the account store view, it can be added by selecting **Custom Account Store Types** from the **Systems List** menu. On the **Custom Account Store** dialog, select **Reset to Default** from the bottom menu bar.

If you have not done so, download the Microsoft Provided IBM DB2 OLE DB provider and install it. Please see the Privileged Identity install guide for more information. Without this provider installed, no management or discovery functionality will be possible. You can add a DB2 database for account enumeration in the following ways:

- With PowerShell: **New-LSSystemInManagementSetCustom**.
- Web service SOAP: **ManagementSetOps\_AddCustomInstanceToManagementSet**.
- Web service REST: **ManagementSet/System/Custom** as a POST.
- Right-click on the **DB2 Databases** node and import from a comma-delimited text file. Each system should be on its own line. This is a comma-separated line-delimited file where the expected format is:

`systemname\initialCatalogName,username,password,namespace,assetTag`. For example:

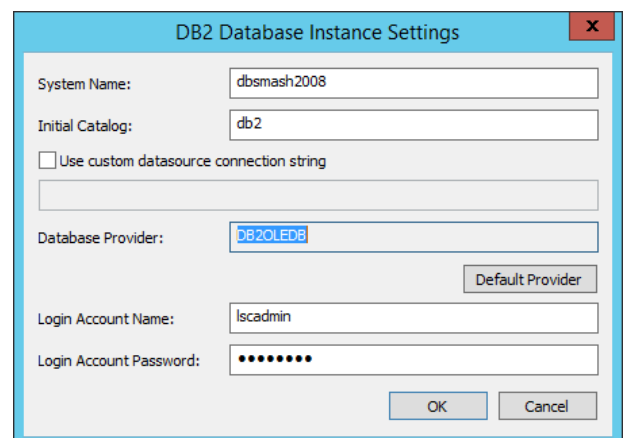
```
db2host\db2,db2admin,P@ssw0rd,,db2-db-host
db2host\db2,db2admin,P@ssw0rd,,
```

Notice in the above example, the second line has no namespace or asset tag but the commas are still present as place holders for those columns.



For more information on available namespaces, please see ["Namespace Values" on page 589](#).

- Right-click on the DB2 Databases node, and then select **Add DB2 Databases**, then click **OK**. The required elements are:
  - **System Name**: The system hosting the DB2 instance.
  - **Initial Catalog**: The initial database used for connection enumeration.
  - **Login Account Name**: The account to connect to the database with.
  - **Login Account Password**: The login account's password.



If the DB2 database is listening on a custom port, a custom data source connection string will also be required. As Privileged Identity makes use of the Microsoft OLE DB provider for DB2, the syntax may be slightly different than the IBM version of the connection string:

```
Provider=DB2OLEDB;Network Transport
Library=TCP;Network Address=192.168.99.117;Network Port=55555;Initial Catalog=db2db;Package
Collection=db2db;User ID=db2admin;Password=password;
```

All parameters included above are required parameters. Even when a custom connection string is used, the same (redundant) information must still be supplied to the dialog fields (e.g. username, password, etc.).

## Enroll MySQL and MariaDB

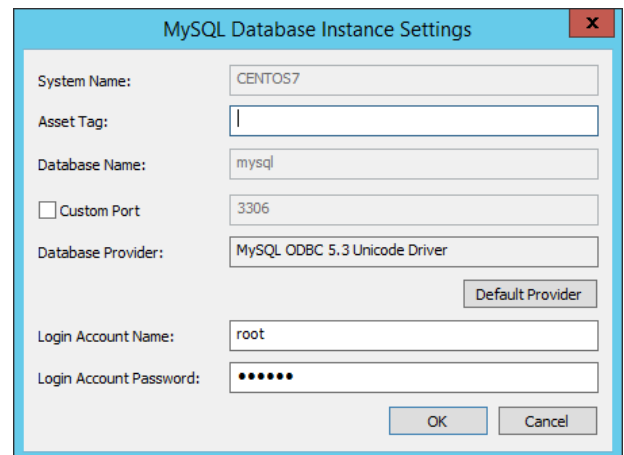
If you have not yet done so, download the OLE DB provider (connector) for MySQL from the [mysql.com](https://mysql.com) web site. See the installation guide for more information. Without this provider installed, no management or discovery functionality will be possible. You can add a MySQL or MariaDB database for account enumeration and management in the following ways:

- With PowerShell: **New-LSSystemInManagementSetMySQLInstance**.
- Web service SOAP: **ManagementSetOps\_AddMySQLInstanceToManagementSet**.
- Web service REST: **ManagementSet/System/MySQL** as a POST.
- Right-click the **MySQL Databases** node and select **Import MySQL Databases from a Text File**. Each system should be on its own line. This is a comma-separated line-delimited file where the expected format is:  
`system,instance,username,password,assetTag,port` where `instance` is the default database name. For example:

```
centdb7,mysql,root,P@ssw0rd,cos7-mysql,33006
centdb6,mysql,root,P@ssw0rd,,
```

Notice in the above example, the second line has no asset tag or custom port, but the commas are still present as place holders for those columns.

- Right-click the **MySQL Databases** node and select **Add MySQL database instance**. The required elements are:
  - **System Name**: The system hosting the MySQL database
  - **Asset Tag**: The asset tag for the database instance
  - **Database Name**: The MySQL database to connect to on the target system; typically MySQL
  - **Login Account Name**: The account to connect to the database with
  - **Login Account Password**: The login account's password



## Enroll Oracle

If you have not yet done so, download the 32bit OLE DB provider for Oracle from the Oracle web site. Please see the Privileged Identity install guide for more information. Without this provider installed, no management or discovery functionality is possible. You can add an Oracle database for account enumeration and management in the following ways:

- With PowerShell: **New-LSSystemInManagementSetOracleInstance**.
- Web service SOAP: **ManagementSetOps\_AddOracleDatabaseToManagementSet**.
- Web service REST: **ManagementSet/System/Oracle** as a POST.
- Right-click the **Oracle Databases** node and select **Import Oracle Database Instances from a Text File**. Each system should be on its own line. This is a comma-separated, line-delimited file where the expected format is:
 

```
system,ServiceName,username,password,CustomConnectionString
```

  - Use a value of 0 in the custom connection string field to indicate no custom connection string is to be defined. If Oracle is not listening on the default communication port, 1521, then a custom connection string might need to be used. See below for more information. A proper connection string can be obtained from the `listener.ora` or `tnsnames.ora` file on the Oracle server.
  - Service name is the service name as defined in the `listeners.ora` file and not the SID name.

For example:

```
centdb7,orcl11g.lsd.s.int,dbmgr,P@ssw0rd,c0s7-orcl,0
centdb6,orcl12c.lsd.s.int,root,P@ssw0rd,,(DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)
(HOST=sql.demo.org)(PORT=6969)(SERVICE_NAME=orcl)))
```

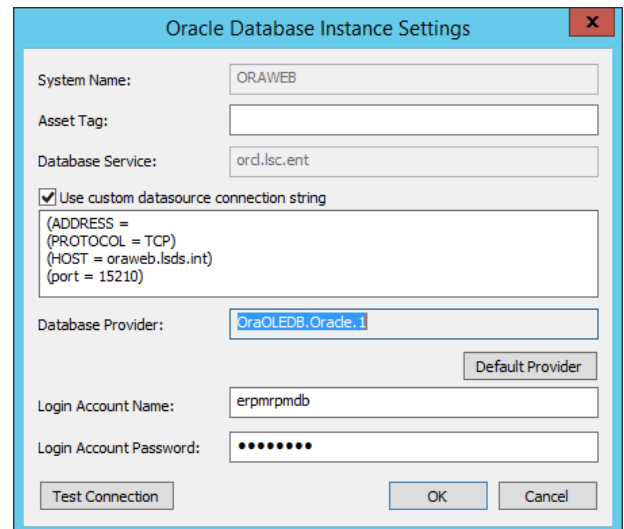
Notice in the above example, the second line has no asset tag, but the commas are still present as place holders for those columns.

- Right-click on the **Oracle Databases** node and select **Add Oracle database instance**. The required elements are:
  - **System Name**: The system hosting the Oracle database.
  - **Asset Tag**: The asset tag for the database instance.
  - **Database Service**: The Oracle service to connect to on the target system - typically: ORCL. This is not necessarily the same as the SID.
  - **Login Account Name**: The account to connect to the database with.
  - **Login Account Password**: The login account's password.

If Oracle is not listening on port 1521, then a custom connection string must be used. A proper connection string can be obtained from the `listener.ora` or `tnsnames.ora` file on the Oracle server.

The connections string would look similar to this:

```
(DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)
(HOST=sql.demo.org)(PORT=6969)(SERVICE_NAME=orcl)))
```



As an alternative to using a custom connection string, if a properly formatted `tnsnames.ora` file is loaded on the Privileged Identity host in the `$ORACLE_HOME$/NETWORK/ADMIN` directory, then the Oracle provider will resolve the name, port, and other relevant information from this file. In this case, simply supply the system name and service name as listed in the `tnsnames.ora` file and omit the custom connection string.

Use the **Test Connection** button to verify your settings for this database instance. Click **OK** to add the instance.

## Enroll PostgreSQL

If you have not yet done so, download the OLE DB provider (connector) for PostgreSQL from the PostgreSQL web site. Please see the Privileged Identity install guide for more information. Without this provider installed, no management or discovery functionality is possible.

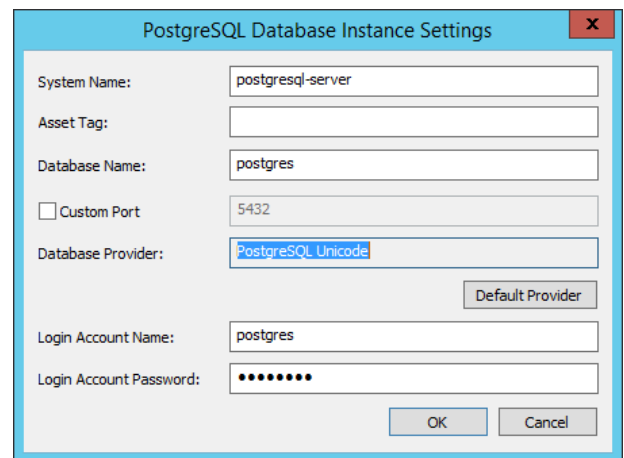
### Adding a PostgreSQL Database for Account Enumeration and Management

- With PowerShell: **New-LSSystemInManagementSetPostgreSQLInstance**.
- Web service SOAP: **ManagementSetOps\_AddPostgreSQLInstanceToManagementSet**.
- Web service REST: **ManagementSet/System/PostgreSQL** as a POST.
- Right-click the **PostgreSQL Databases** node and select **Import PostgreSQL Databases from a Text File**. Each system should be on its own line. This is a comma-separated line-delimited file where the expected format is:  
`system,database,username,password,assetTag,port` where `instance` is the default database name. For example:

```
centdb7,postgres,postgres,P@ssw0rd,cos7-postgres,54320
centdb6,postgres,postgres,P@ssw0rd,,0
```

Notice in the above example, the second line has no asset tag but the commas are still present as place holders for those columns. Also note the zero in the port column. Use a "0" to denote there is no custom port.

- Right-click the **MySQL Databases** node and select **Add MySQL database instance**. The required elements are:
  - **System Name:** The system hosting the PostgreSQL database
  - **Asset Tag:** The asset tag for the database instance
  - **Database Name:** The PostgreSQL database to connect to on the target system; typically postgres
  - **Login Account Name:** The account to connect to the database with
  - **Login Account Password:** The login account's password



## Enroll Microsoft SQL Server

If you are managing a SQL Server instance configured to require TLS 1.2, you need to download and install the latest SQL Native Client provider on all components that will connect to the SQL Server instance or management will fail.

**i** For more information, please see <https://support.microsoft.com/en-us/help/3135244/tls-1.2-support-for-microsoft-sql-server>.

You can add a Microsoft SQL database for account enumeration and management in the following ways:

- With PowerShell: **Remove-LSSystemFromManagementSetSQLInstance**.
- Web service SOAP: **ManagementSetOps\_AddSQLInstanceToManagementSet**.
- Web service REST: **ManagementSet/System/SQLServer** as a POST.
- Right-click on Windows systems in the same management set under the **Windows Systems** node and select **Search for SQL Server Instances**.
- Use any of the dynamic discovery options such as Active Directory Paths from the management set properties dialog to define a list of systems that will be searched for SQL Server instances using the Scan for Target Type scan option. There are many options for adding systems dynamically and manually. Please see Creating a Management Set for more information.
- Right-click the **SQL Server Instances** node and select to **Import SQL Instances from a Text File**. Each system should be on its own line. This is a comma-separated, line-delimited file where the expected format is:

```
system,instance,port,username,password,asset tag,provider,encryption.
```

- For default instances of SQL, specify the instance name as **DEFAULT**.
- For the port, specify either the assigned port or 0 if using a dynamic port assignment (default on named instances and SQL Server 2005 and later installations).
- If using explicit SQL authentication, input the user name and password. For integrated SQL authentication, specify **INTEGRATED** for both the user name and password field.
- For provider, choose from the following values:
  - **2** - Default OLE DB Provider
  - **4** - v9 SQL Native Client OLE DB
  - **5** - v10 SQL Native Client OLE DB
  - **6** - v11 SQL Native Client OLE DB
  - **7** - v9 SQL Native Client ODBC
  - **8** - v10 SQL Native Client ODBC
  - **9** - v11 SQL Native Client ODBC

The best choices are 2 or 9. Use 2 for non-encrypted connections or connections secured with SSL or TLS 1.0. Use 9 when the connection is encrypted and protected with TLS 1.2. The latest version of the SQL Native Client must be installed to use TLS 1.2 when communicating with SQL Server. The SQL Native Client must also be installed on any system that will connect to and/or manage that particular database instance.

- For encryption, set to 1 to use encryption (which will be negotiated with the server), or leave empty to specify no encryption.

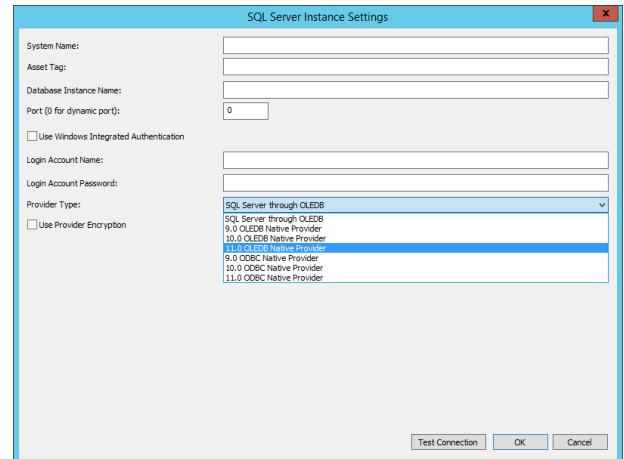
- For example:

```
msdb1, redim, 49298, INTEGRATED, INTEGRATED, , 9, 1
msdb2, DEFAULT, 0, sa, P@ssw0rd, DMZ-SQL-SRV, 2, 0
```

Notice in the above example, the first line has no asset tag but the commas are still present as place holders for that columns.

- Right-click on the **SQL Server Instances** node and select **Add SQL Server instance**. The required elements are:

- **System Name:** The system hosting the SQL Server instance.
- **Asset Tag:** The asset tag for the database instance.
- **Database Instance Name:** The SQL database instance to connect to on the target system; typically left blank for default instances.
- **Port:** The port the SQL Server instance is listening on (default is 1433). If SQL Server is set to use dynamic ports (typical for named or multiple instances), set the port value to equal zero, 0, which will direct Privileged Identity to use dynamic ports.
- **Use Integrated Windows Authentication:** When enabled specifies Privileged Identity will use Integrated Windows Authentication to connect to the target database instance. When running an interactive job, this will be the credentials of the interactive user. When running a scheduled job, this will be the credentials of the deferred or zone processor service account.
- **Login Account Name:** If Integrated Windows Authentication will not be used, then specify the name of a SQL Local account, for example: sa.
- **Login Account Password:** The login account's password if an explicit SQL login account name is used.
- **Provider Type:** SQL Server through OLE DB is the default selection and sufficient for almost all SQL Server installations. If managing server where TLS 1.2 is required, install and use the 11.0 ODBC Native Provider (see note above).
- **Use Provider Encryption:** Select this option to negotiate the use of SSL or TLS with the target server. If the target server requires SSL or TLS, this option must be selected.



- Use the **Test Connection** button to verify your settings for this database instance. Click **OK** to add the instance.



## Enroll Sybase ASE

If you have not yet done so, download the OLE DB provider (connector) for Sybase ASE from the Sybase web site. Please see the Privileged Identity install guide for more information. Without this provider installed, no management or discovery functionality is possible.

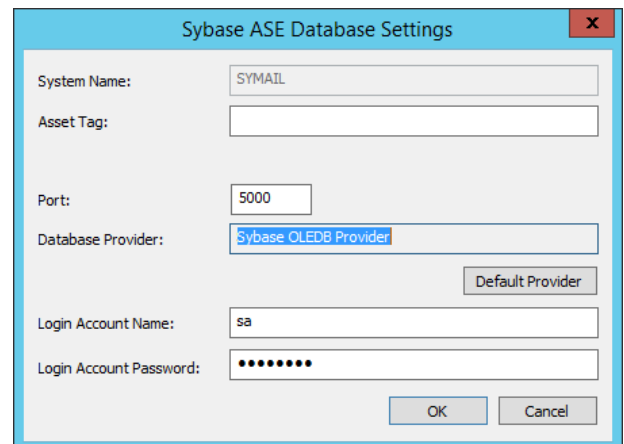
### Add a Sybase Database for Account Enumeration and Management

- With PowerShell: **New-LSSystemInManagementSetSybaseInstance**.
- Web service SOAP: **ManagementSetOps\_AddSybaseInstanceToManagementSet**.
- Web service REST: **ManagementSet/System/Sybase** as a POST.
- Right-click on the **Sybase ASE Databases** node and select **Import Sybase Databases from a Text File**. Each system should be on its own line. This is a comma-separated, line-delimited file where the expected format is:  
`system,instance,port,username,password,asset tag`. For example:

```
centdb7,,5000,sybadm,P@ssw0rd,cos7-sybase
centdb6,,55555,sybadm,P@ssw0rd,cos6-transrv
```

Notice the instance names are both blank, but the commas are still present to hold the place values.

- Right-click the **Sybase ASE Databases** node and select **Add Sybase ASE database**. The required elements are:
  - **System Name**: The system hosting the Sybase ASE database.
  - **Asset Tag**: The asset tag for the database instance.
  - **Database Instance Name**: The Sybase ASE database to connect to on the target system; typically left blank.
  - **Login Account Name**: The account to connect to the database with.
  - **Login Account Password**: The login account's password.



## Enroll Teradata

If you have not yet done so, download the OLE DB provider (connector) for Teradata from the Teradata web site. Please see the Privileged Identity install guide for more information. Without this provider installed, no management or discovery functionality is possible.

### Add a Teradata Database for Account Enumeration and Management

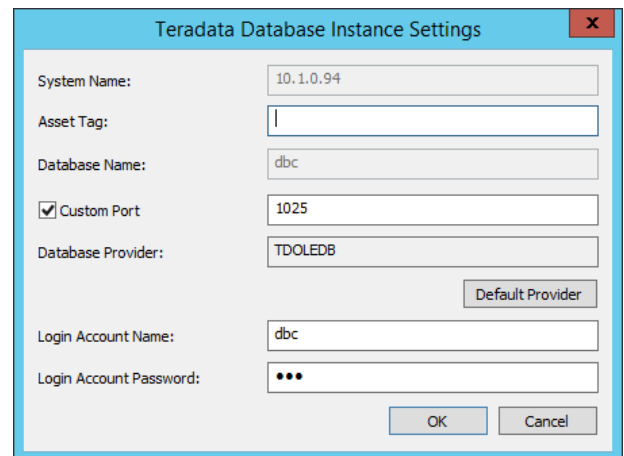
There are four ways to register a Teradata database with Privileged Identity to manage the database's local accounts.

- With PowerShell: **New-LSSystemInManagementSetTeradataInstance**.
- Web service URI: **ManagementSetOps\_AddTeradataInstanceToManagementSet**.
- Web service REST: **ManagementSet/System/Teradata** as a POST.
- Right-click on the **Teradata Databases Instances** node and select **Import Teradata Databases from a Text File**. Each system should be on its own line. This is a comma-separated, line-delimited file where the expected format is:  
`system,database,username,password,asset tag,port`. For example:

```
centdb7,dbc,tdadm,P@ssw0rd,cos7-td1,1025
centdb6,dbc,tddbcadm,P@ssw0rd,,1025
```

Notice the asset tag is not supplied for the second instance, but the commas are still present to hold the place values.

- Right-click the **Teradata Databases Instances** node and select **Add Teradata Database Instance**. The required elements are:
  - **System Name**: The system hosting the Teradata database.
  - **Asset Tag**: The asset tag for the database instance.
  - **Database Name**: The Teradata database to connect to on the target system; typically dbc.
  - **Custom Port**: define a custom port if configured to listen on a port other than 1025
  - **Login Account Name**: The account to connect to the database with.
  - **Login Account Password**: The login account's password.



## Enroll LDAP Directories

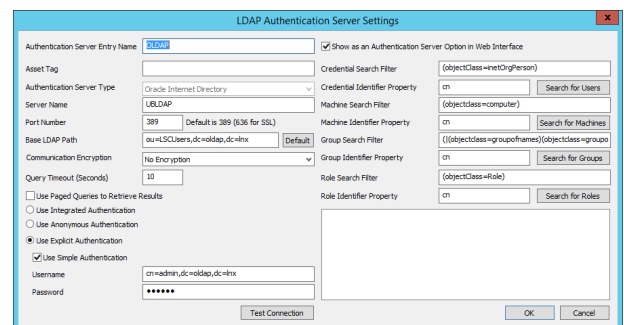
Privileged Identity can manage passwords in any LDAP compliant directory. From the **Account Store View** there are four directory node types available:

- Oracle Internet Directories
- Novell eDirectory Databases
- IBM Tivoli Directories
- ViewDS Directories

The purpose of these nodes is to identify default search and authentication settings appropriate to the directory types. It is possible to add any directory type that is LDAP compatible to any of the nodes. You can enroll an LDAP directory in the following ways:

- With PowerShell: **New-LSSystemInManagementSetLDAPServer**.
- Web service SOAP: **ManagementSetOps\_AddLDAPServerToManagementSet**.
- Web service REST: **ManagementSet/System/LDAP** as a POST.
- If an Authentication Server entry already exists for the LDAP directory, simply right-click on the appropriate/desired node and select **Add Existing Entry from Authenticator List** to the appropriate node.
- Choose the appropriate node (OID, eDirectory, etc.), right-click and select **Add <...> Instance**. then supply the appropriate instance information. This method will also add an Authentication Server entry. Authentication Servers permit users from these directories to be available for delegations when recovering passwords or performing other actions in the web site.

- **Authentication Server Entry Name:** the name friendly directory name. This value will be appended to the server name when the directory is added to the list. When also used as an authentication server entry, this name will appear in the authenticators list on the web site's login page.
- **Asset Tag:** (optional) add an asset tag for this directory.
- **Authentication Server Type:** This entry will be filled out depending on what node you are adding your directory service to. The sole purpose of this [pre-made] selection is to define the default search filters for the right side of the dialog.
- **Server Name:** The name of the server Privileged Identity should bind to. This can be a short name, FQDN, or IP address.
- **Port Number:** The port your directory is listening on. The default port is the default non-secured LDAP port, 389. If you know your directory requires SSL/TLS, then set the port to 636. Otherwise, supply an alternate port if the directory is not listening on the default port.
- **Base LDAP Path:** The path from which all object searches will start. When used as an authentication server, this value will be appended to the user's name automatically when logging into the web site.
- **Communication Encryption:** Choose from No Encryption (default), Use Start TLS, or Use SSL. If using TLS or SSL, the Privileged Identity host making the connection must trust the cert and there can be no certificate errors.
- **Query Timeout (Seconds):** The time that any query can take before the call times out. The default is 10 seconds. Set this to a higher value if the target LDAP server is slow to respond.
- **Use Pages Queries to Retrieve Results:** The use of this option is directory specific. If the login is successful and search filters are valid, enable this option if queries fail. Not all directories support paged queries.



- **Authentication:**
  - **Use Integrated Authentication:** this option is set by default for Windows Domains and will use the credentials of the calling user during web site/service logins when a username and password is provided or will use the COM application identity credentials when Integrated Windows Authentication is performed if no username and password is provided. For management operations, the interactive user or deferred/zone processor account will be used to perform lookups and management. This option is typically not supported by anything other than Windows Active Directory domains.
  - **Use Anonymous authentication:** If the directory supports lookups using anonymous authentication, this option may be used. Management operations (password resets) will typically fail with this configuration.
  - **Use Explicit Authentication:** Lookup and password reset functions will use the account name and password specified in the username and password field. Enable **Use Simple Authentication** to pass the exact name shown in the username field.
- **Show as an Authentication Server Option in the Web Interface:** enable this option to make this LDAP server available to the web site for user logins. The Authentication Server Entry Name will be shown in the Authenticators list.
- **{Option} Search Filter:** The filter that will be used to locate user, computer or groups from the Base LDAP Path.
- **{Option} Identifier Property:** The attribute that, when present, will be used to identify the object. For web site logins, this value will be pre-pended to the username. Use the appropriate search button to test the search filter and identifier property query. Results will be shown in the results pane in the lower right corner.

# Enroll Middleware, Application Servers, and Enterprise Software

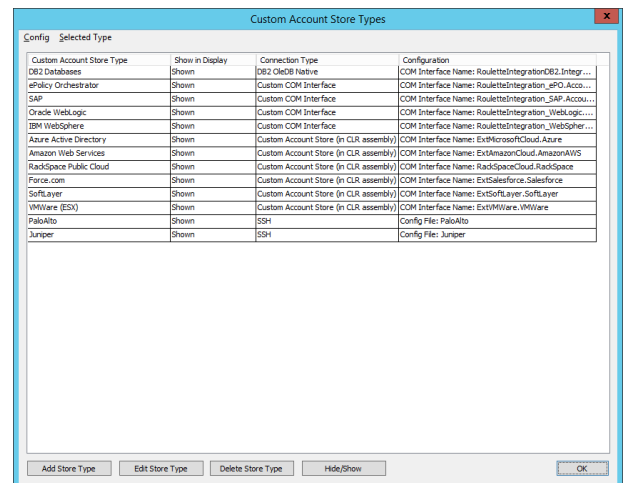
This section describes how to enroll supported middleware, application servers (IBM WebSphere and Oracle WebLogic), and other enterprise software.

## Enroll McAfee ePO

McAfee ePolicy Orchestrator can have its privileged identities managed by Privileged Identity. This is done by manipulating information directly in the ePO database. To enroll an EPO system, the name and connection information of the EPO SQL database will be required.

McAfee ePolicy Orchestrator is not a default type within Privileged Identity. To make the node visible in the **Account Store** view, open the **SystemsList** menu then open **Custom Account Store Types**. If ePolicy Orchestrator is not listed as a custom account store type, click **Register Built-in Types** from the **Config** menu. The dialog may not respond for a short period of time (up to 60 seconds) while the necessary COM registrations are created in Windows and then registered in Privileged Identity. Once the registration is complete, ePolicy Orchestrator will be listed as a custom account store type.

Click **OK**.



## Enroll an EPO database

- With PowerShell: **New-LSSystemInManagementSetCustom**.
- Web service SOAP: **ManagementSetOps\_AddCustomInstanceToManagementSet**.
- Web service REST: **ManagementSet/System/Custom** as a POST.
- Right-click on the **ePolicy Orchestrator** node and import from a comma-delimited text file. Each system should be on its own line. This is a comma-separated line-delimited file where the expected format is:

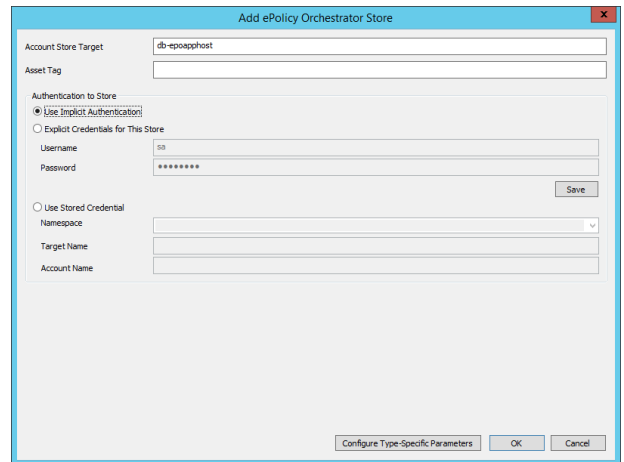
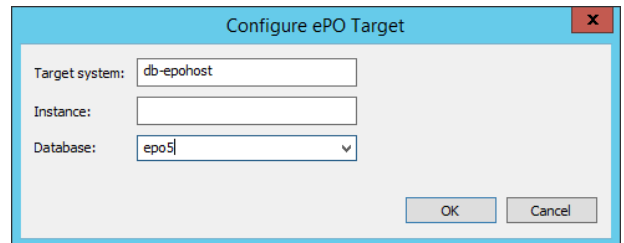
`TargetName,username,password,namespace,assetTag` where `TargetName` is the DB-Host. For example:

```
msdb1,sa,P@ssw0rd,,ent-app-dbhost-1
msdb2,sa,,LSC,ent-app-dbhost-2
```

Notice in the above example, the first line uses explicit authentication and thus leaves the namespace attribute blank. The second line uses a stored managed DB account and thus leaves the password field blank and then specifies the namespace of the target account. The commas representing the required values are still present as place holders for those columns. See **Namespace Values** for more information on available namespaces.

You must still edit the account store once the import is complete and click **Configure Type Specific Parameters** and specify the Instance name if required and the target EPO database.

- Right-click on the ePolicy Orchestrator node and select **Add ePolicy Orchestrator**. The required elements are:
  - **Account Store Target:** The name of the EPO database host.
  - **Asset Tag:** The supply an asset tag for this entry.
  - **Authentication to Store:** Select how to authenticate to the EPO database using one of the following options.
    - **Use Implicit Authentication:** Select this option to use the credentials of the interactive user for interactive operations or the deferred/zone processor credentials for scheduled operations. This option is valid for database hosts that are domain joined and trust the account used to make the connection.
    - **Explicit Credentials for this Store:** Use this option to use SQL Server explicit login accounts, such as SA, to connect to the database host. These credentials will be saved for future connections and added to the secure password store which will also make them available for retrieval via the web or web service interface.
    - **Use Stored Credential:** Use a credential that has already been stored (managed) from the secure password store.
- Click **Configure Type-Specific Parameters**. This is the configuration dialog for connecting to the EPO database. The target system name will be filled out. If the ePO database is running on a named instance of SQL, supply the instance name, otherwise, leave blank.
- Select the EPO database from the Database list, and click **OK**.

## Enrolling SAP

Privileged Identity can directly manage at least SAP v7.01 (see SAP note 1287410) using direct API calls or using the SAP Netweaver gateway. The SAP NetWeaver Gateway needs to be at least version 7.02 SP10. Using the SAP NetWeaver Gateway requires further configuration on the SAP NetWeaver Gateway instance. Please see "[Configure SAP Gateway for Enrollment](#)" on page 125 for details.

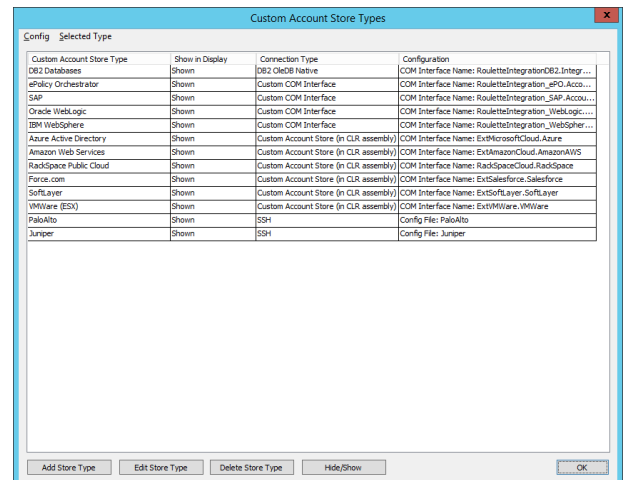
Also, before Privileged Identity can manage an SAP instance, copy **librfc32.dll** from the SAP installation binaries to the **%systemroot%\SysWoW64** directory of the Privileged Identity host system. Then run the **regsvr32 librfc32.dll** command from an administrative command prompt.

Within SAP, for password changes to be made:

- The logon session during which a productive password is set must be secured using Secure Network Communications (SNC).
- The user (for example, the communication user for an RFC destination) needs an additional authorization to set a productive password (authorization object: S\_USER\_GRP, activity: 'PP' - Set Productive)

Contact your SAP administrator for more information.

SAP is not a default type within Privileged Identity. To make the node visible in the **Account Store** view, open the **SystemsList** menu then open **Custom Account Store Types**. If SAP is not listed as a custom account store type, click **Register Built-in Types** from the **Config** menu. The dialog may not respond for a short period of time (up to 60 seconds) while the necessary COM registrations are created in Windows and then registered in Privileged Identity. Once the registration is complete, SAP will be listed as a custom account store type.



## To Enroll an SAP Instance

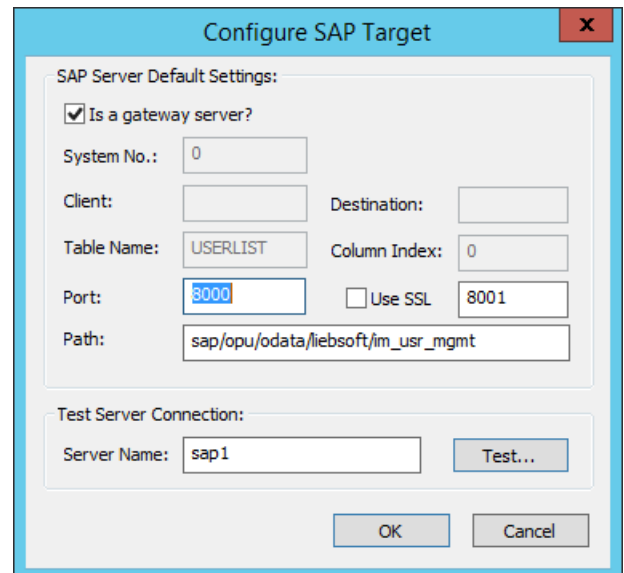
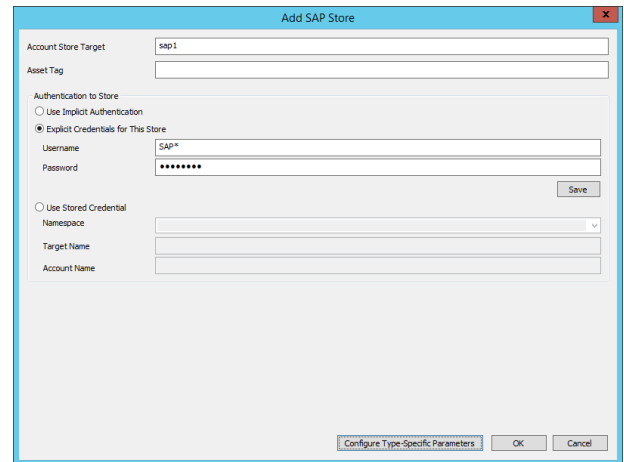
- With PowerShell: **New-LSSystemInManagementSetCustom**.
- Web service URI: **ManagementSetOps\_AddCustomInstanceToManagementSet**.
- Web service REST: **ManagementSet/System/Custom** as a POST.
- Right-click on the SAP node and import from a comma-delimited text file. Each system should be on its own line. This is a comma-separated line-delimited file where the expected format is: `TargetName, username, password, namespace, assetTag` where `TargetName` is the target SAP server. For example:

```
sap1, SAP*, P@ssw0rd, , sap-host-1
sap2, sap-adm, , LSC, sap-host-2
```

Notice in the above example, the first line uses explicit authentication and thus leaves the namespace attribute blank. The second line uses a stored managed account and thus leaves the password field blank and then specifies the namespace of the target account. The commas representing the required values are still present as place holders for those columns. See "[Namespace Values](#)" on page 589 for more information on available namespaces.

You must still edit the account store once the import is complete and click **Configure Type Specific Parameters** and specify the gateway and/or client settings.

- Right-click on the SAP node and select **Add SAP**. The required elements are:
  - **Account Store Target** - the name of the SAP host server.
  - **Asset Tag - optional** - supply an asset tag for this entry.
  - **Authentication to Store** - select how to authenticate to the SAP Instance using one of the following options.
    - **Use Implicit Authentication** - Selecting this option will use the credentials of the interactive user for interactive operations or the deferred/zone processor credentials for scheduled operations. This option is valid for SAP instances hosts that are domain joined and trust the account used to make the connection. This is not typical for managing SAP installations.
    - **Explicit Credentials for this Store** - Use this option to use SAP explicit login accounts, such as SAP\*, to connect to the SAP instance. These credentials will be saved for future connections and added to the secure password store which will also make them available for retrieval via the web or web service interface.
    - **Use Stored Credential** - use a credential that has already been stored (managed) from the secure password store.
- Then click **Configure Type-Specific Parameters**. This is the configuration dialog for connecting to the SAP instance. The target system name will be filled out. Fill out the requisite information regarding system number, client number, etc. Contact your SAP administrator for valid connection and path information. Using the SAP NetWeaver Gateway requires further configuration on the SAP NetWeaver Gateway instance. Please see "[Configure SAP Gateway for Enrollment](#)" on page 125 for details.





## Configure SAP Gateway for Enrollment

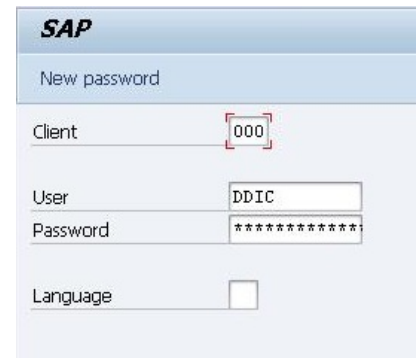
When managing SAP using the SAP NetWeaver Gateway there are additional configurations to be made in the SAP instance and **the SAP NetWeaver Gateway needs to be at least version 7.02 SP10 and the minimum Add-On installation tool's version is 7.02/0045.**

- From the Gateway host, copy the `LIEBSOFT0ERP` `0010000.PAT` file from the SupplementalInstallers folder of the product installation directory to the EPS inbox folder on the SAP server. The typical location for the EPS inbox folder is `\usr\sap\trans\EPS\in`.

The password for the package is: **8B80CF5DCC**

GW_CORE	200	0003 SAP GW CORE 200
WEBCUIF	701	0004 SAP Web UI Framework
IW_CBS	200	0003 SAP IW CBS 200
IW_CNT	200	0003 SAP IW CNT 200
IW_FND	250	0003 SAP IW FND 250
SAP_BW	702	0010 SAP Business Warehouse
PI_BASIS	702	0010 Basis Plug-In
SAP_ABA	702	0010 Cross-Application Component
SAP_BASIS	702	0010 SAP Basis Component

- Start the SAP GUI, log in from client 000, and go to transaction SAINT.



The screenshot shows the SAP login interface. At the top, it says "SAP" and "New password". Below that, there are input fields for "Client" (with "000" entered), "User" (with "DDIC" entered), "Password" (with "\*\*\*\*\*" entered), and "Language" (with a dropdown menu).

- Go to **Installation Package > Load Packages From Application Server**. Verify the file was uploaded successfully.



The screenshot shows the SAINT transaction window titled "SAINT: Uploading Packages from the File System". It contains a table with the following data:

OCS File Name	Package	Result	RC	Message Text
NSP0020610633_0000002.PAT	SAPK-170COINLIEBSOFT	00	0000	Uploaded successfully

- Go to transaction SAINT and verify all the prerequisite components have been installed. Click the **Start** button to continue.

**Add-On Installation Tool - Version 7.02/0045**

Add-On Installation Tool : Installed Add-ons

Add-on/PCS	Release	Level	Description	Import ..
GW_CORE	200	0003	SAP GW CORE 200	OC
IW_BEP	200	0003	Backend Event Provider	OC
IW_CBS	200	0003	SAP IW CBS 200	OC
IW_CNT	200	0003	SAP IW CNT 200	OC
IW_FND	250	0003	SAP IW FND 250	OC
IW_SCS	200	0003	Screen Scraping	OC
PI_BASIS	702	0010	Basis Plug-In	OC
SAP_ABA	702	0010	Cross-Application Component	OC
SAP_BASIS	702	0010	SAP Basis Component	OC

**Status/Remarks**

**i** The overview shows you the installed Add-ons and Preconfigured systems  
 - Choose [START] to start an installation or an upgrade

Back Start Cancel

**Add-On Installation Tool - Version 7.02/0045**

Add-On Installation Tool : Installed Add-ons

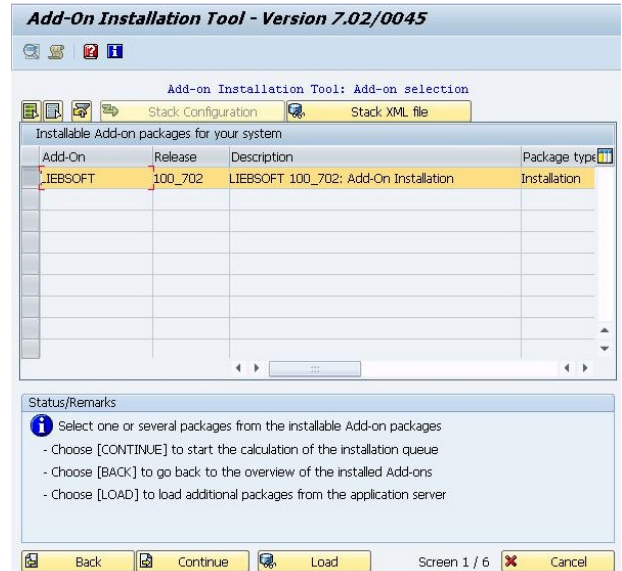
Add-on/PCS	Release	Level	Description	Import ..
SAP_BW	702	0010	SAP Business Warehouse	OC
WEBCUIF	701	0004	SAP Web UI Framework	OC

**Status/Remarks**

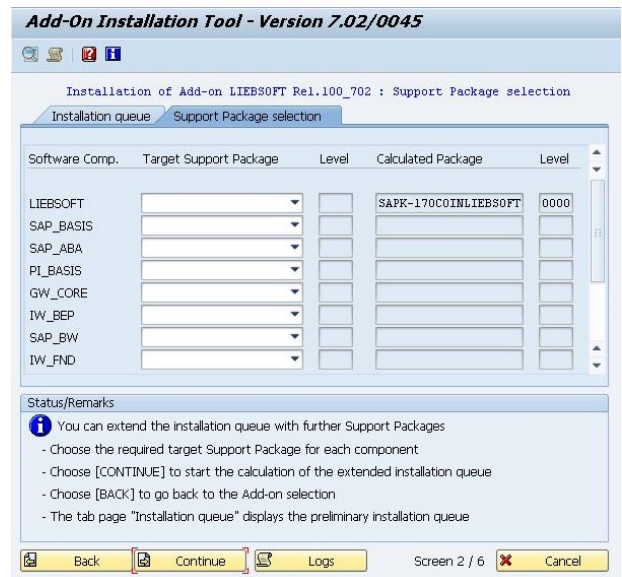
**i** The overview shows you the installed Add-ons and Preconfigured systems  
 - Choose [START] to start an installation or an upgrade

Back Start Cancel

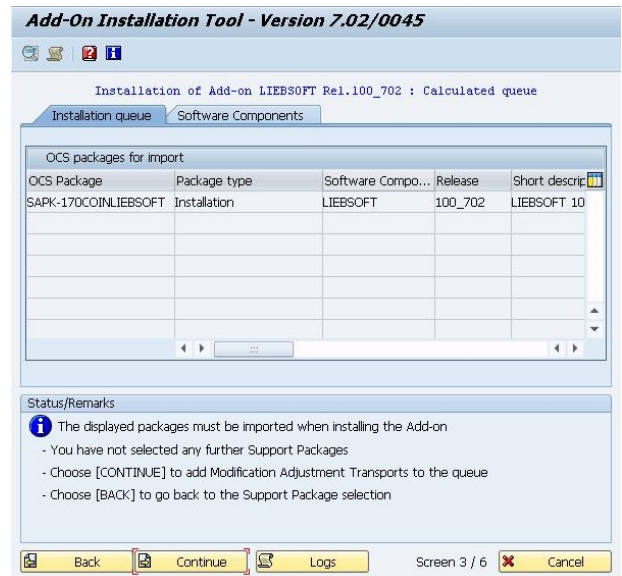
5. Verify and select the Add-On to be installed, then click **Continue**.



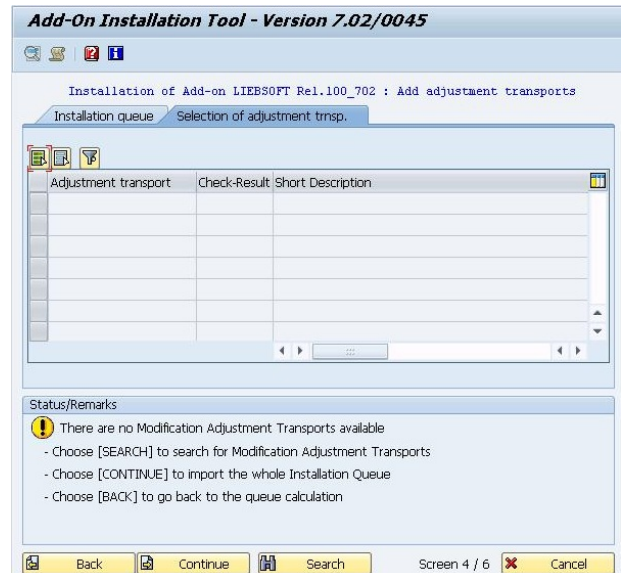
6. Click **Continue** on the next screen.



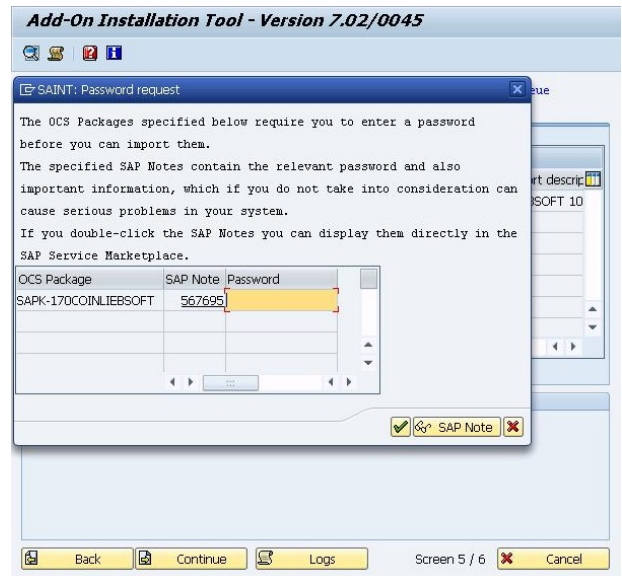
- Click **Continue**. Choose **Yes** when prompted whether you want to add Modification Adjustment Transports to the queue.



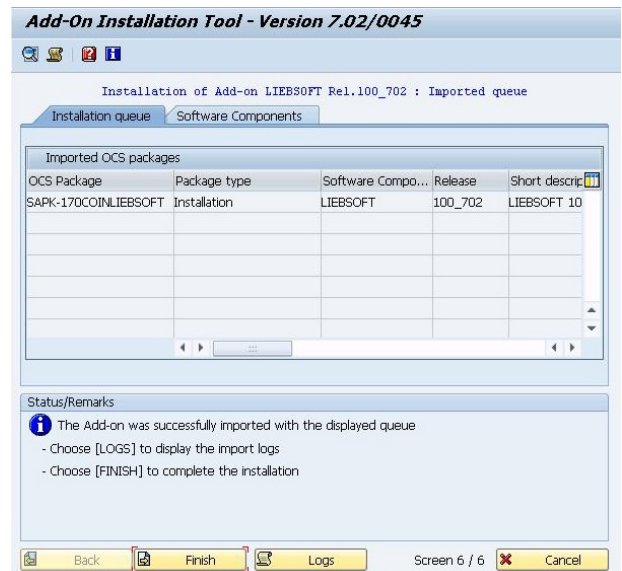
- Click **Continue** on the next screen.



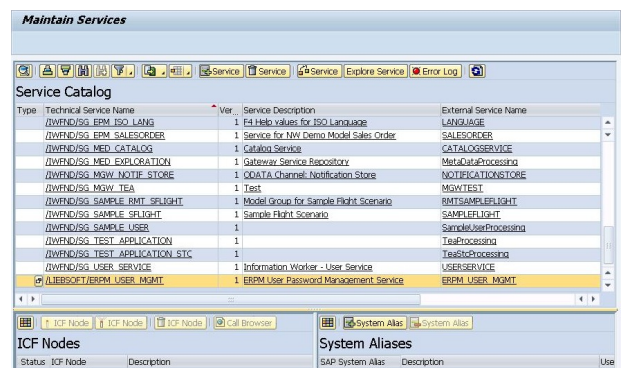
- Enter the password for the Add-On component you are installing. The password for the package is **8B80CF5DCC**. Then click **Continue** and then **Import**. The import process will then start.



- Once the installation indicated the add-on was successfully imported, click **Finish**.

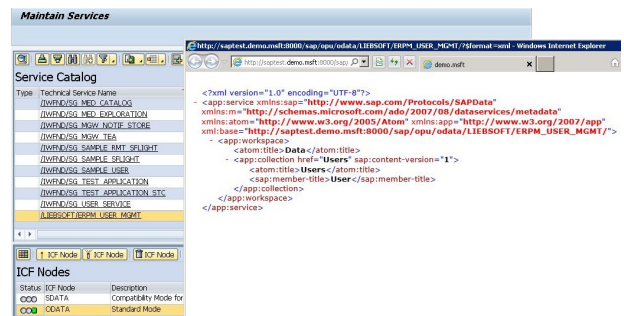
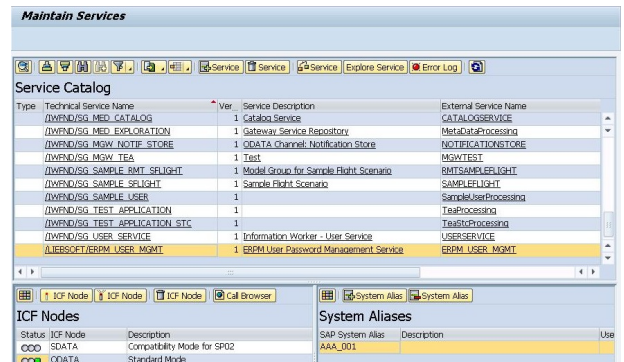
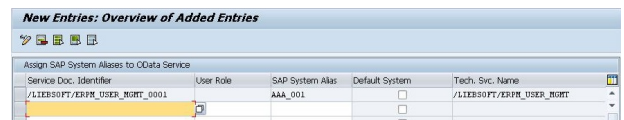
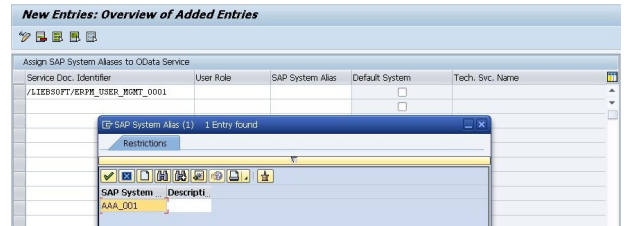
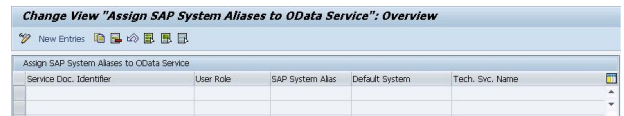


- Log on to the Gateway system as a client exception 000. Run transaction `/iwfnd/maint_service` and locate `/LIEBSOFT/ERPM_USER_MGMT`.





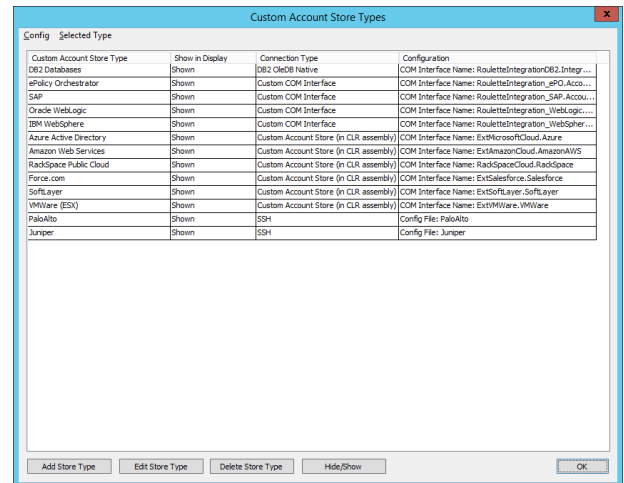
12. In the **System Aliases** panel, click on **System Alias** to add the system alias related to this service.
13. Click **New Entry**. Select **/LIEBSOFT/ERPM\_USER\_MGMT\_0001** as the Service Doc. Identifier. Select the system alias of the ABAP system that contains component **LIEBSOFT/ERPM\_USER\_MGMT** and press enter.
14. The following image depicts an added system alias. Save the change and go back to the maintain service screen.
15. Go back and double click **/LIEBSOFT/ERPM\_USER\_MGMT**. Its corresponding ODATA and SDATA information will be shown. Highlight the **ODATA** nodes and click **ICF Node** to activate it. The status icon will change to green.
16. Click on **Call Browser** to test the service.
17. After providing the credential, you should see the metadata information for this service.
18. The installation this Add-On is now complete.



## Enroll Oracle

### Enroll Oracle WebLogic

Oracle WebLogic is not a default type within Privileged Identity. To make the node visible in the **Account Store** view, open the **SystemsList** menu then open **Custom Account Store Types**. If Oracle WebLogic is not listed as a custom account store type, click **Register Built-in Types** from the **Config** menu. The dialog may not respond for a short period of time (up to 60 seconds) while the necessary COM registrations are created in Windows and then registered in Privileged Identity. Once the registration is complete, Oracle WebLogic will be listed as a custom account store type.



### To Enroll an Oracle WebLogic Instance

#### ! IMPORTANT!

Before management of a WebLogic instance can occur, additional steps must be taken in the target WebLogic instance. See ["Configure Oracle WebLogic for Enrollment"](#) on page 132 for more information.

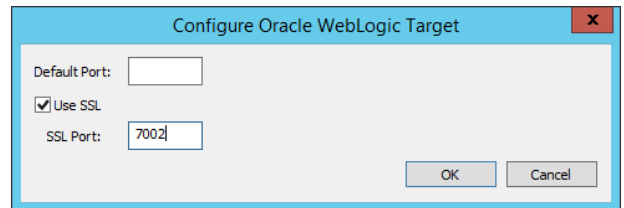
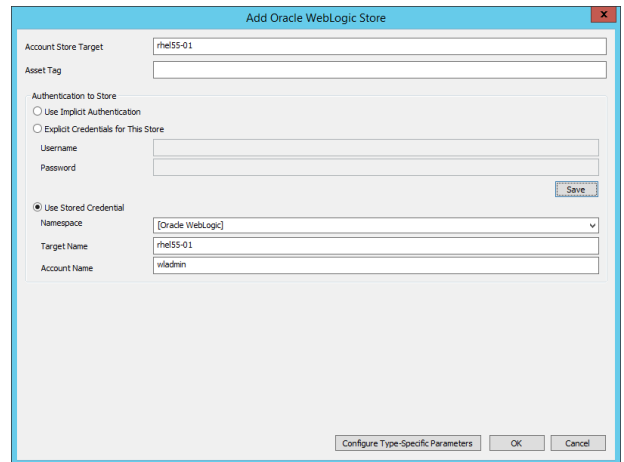
- With PowerShell: **New-LSSystemInManagementSetCustom**.
- Web service URI: **ManagementSetOps\_AddCustomInstanceToManagementSet**.
- Web service REST: **ManagementSet/System/Custom** as a POST.
- Right-click on the **Oracle WebLogic** node and import from a comma-delimited text file. Each system should be on its own line. This is a comma-separated line-delimited file where the expected format is:  
`TargetName,username,password,namespace,assetTag` where `TargetName` is the target WebLogic server. For example:

```
wlweb-1,wladmin,IHeartOracleWL,,wl-host-1
wlweb-2,wl-dir-adm,,[LDAP],wl-host-2
```

Notice in the above example, the first line uses explicit authentication and thus leaves the namespace attribute blank. The second line uses a stored managed account and thus leaves the password field blank and then specifies the namespace of the target account. The commas representing the required values are still present as place holders for those columns. See ["Namespace Values"](#) on page 589 for more information on available namespaces.

- You must still edit the account store once the import is complete and click **Configure Type Specific Parameters** and specify the connection port and SSL configurations.

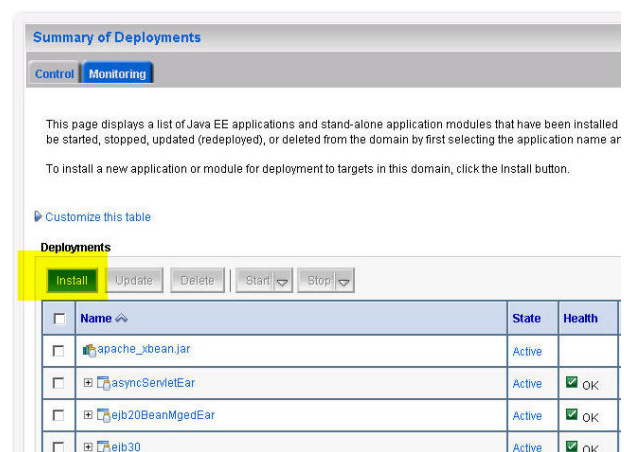
- Right-click on the Oracle WebLogic node and select **Add Oracle WebLogic**. The required elements are:
  - **Account Store Target** - the name of the Oracle Web Logic host server.
  - **Asset Tag** - optional - supply an asset tag for this entry.
  - **Authentication to Store** - select how to authenticate to the Oracle WebLogic Instance using one of the following options.
    - **Use Implicit Authentication** - Selecting this option will use the credentials of the interactive user for interactive operations or the deferred/zone processor credentials for scheduled operations. This is not typical for managing Oracle WebLogic installations.
    - **Explicit Credentials for this Store** - Use this option to use WebLogic explicit login accounts to connect to the WebLogic instance. These credentials will be saved for future connections and added to the secure password store which will also make them available for retrieval via the web or web service interface.
    - **Use Stored Credential** - use a credential that has already been stored (managed) from the secure password store.
- Then click **Configure Type-Specific Parameters**. This is the configuration dialog for connecting to the WebLogic instance.
- Identify the management port. If SSL is to be used, select the Use SSL option and supply the correct SSL port.



## Configure Oracle WebLogic for Enrollment

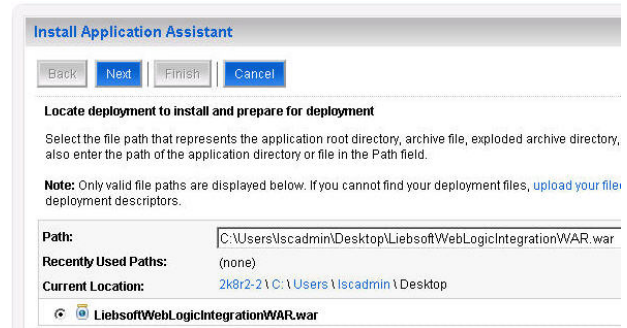
In order for WebLogic propagation to work, install and configure **LiebssoftWebLogicIntegrationEAR.ear**. This binary is found in the **SupplementalInstallers** sub-folder of the Privileged Identity installation. LiebssoftWebLogicIntegrationEAR, is a Java Servlet running as a WebLogic Web Application.

1. Log into WebLogic as a user which can deploy a Web Application.
2. From the **Home Page > Domain Configurations** section, open **Deployments**.
3. On the **Summary of Deployments** page, click **Install**.





- From the **Install Web Application Assistant** page, supply the path to the **LiebssoftWebLogicIntegrationEAR.ear** file, select it and click **Next**.



**Install Application Assistant**

Back Next Finish Cancel

**Locate deployment to install and prepare for deployment**

Select the file path that represents the application root directory, archive file, exploded archive directory, also enter the path of the application directory or file in the Path field.

**Note:** Only valid file paths are displayed below. If you cannot find your deployment files, [upload your file](#) deployment descriptors.

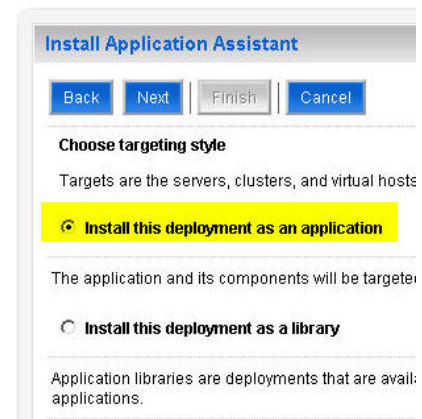
**Path:** C:\Users\iscadmin\Desktop\LiebssoftWebLogicIntegrationWAR.war

**Recently Used Paths:** (none)

**Current Location:** 2kbR2-2 | C:\Users\iscadmin\Desktop

LiebssoftWebLogicIntegrationWAR.war

- Select to **Install this deployment as an application** then click **Next**.



**Install Application Assistant**

Back Next Finish Cancel

**Choose targeting style**

Targets are the servers, clusters, and virtual hosts

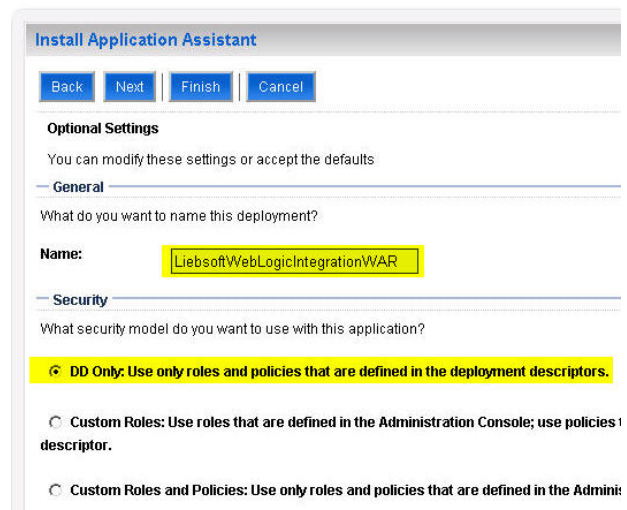
**Install this deployment as an application**

The application and its components will be targeted to the selected targets.

**Install this deployment as a library**

Application libraries are deployments that are available to all applications.

- Supply the name of the application as **LiebssoftWebLogicIntegrationEAR** and set the **Security Model** to **DD Only: Use only roles and policies that are defined in the deployment descriptors**, then click **Next**.



**Install Application Assistant**

Back Next Finish Cancel

**Optional Settings**

You can modify these settings or accept the defaults

**General**

What do you want to name this deployment?

**Name:** LiebssoftWebLogicIntegrationWAR

**Security**

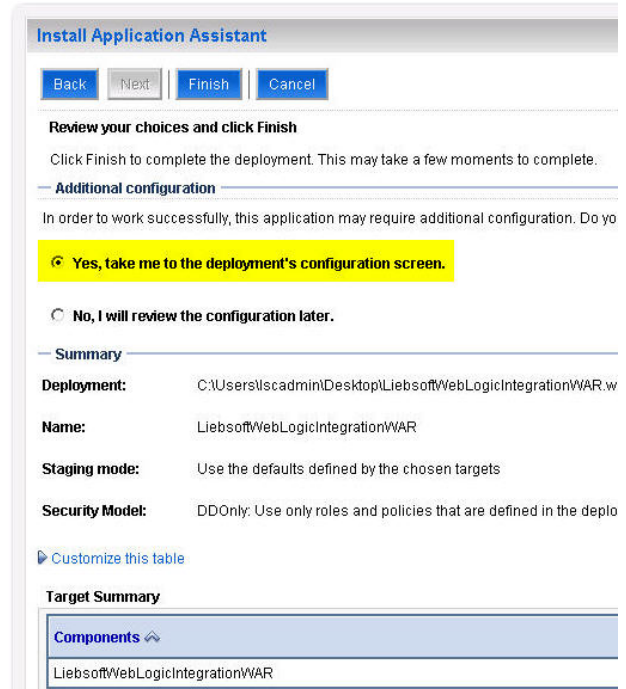
What security model do you want to use with this application?

**DD Only: Use only roles and policies that are defined in the deployment descriptors.**

**Custom Roles: Use roles that are defined in the Administration Console; use policies in the deployment descriptor.**

**Custom Roles and Policies: Use only roles and policies that are defined in the Administration Console.**

7. Select **Yes, take me to the deployment's configuration screen** then click **Finish**.



**Install Application Assistant**

Back Next Finish Cancel

**Review your choices and click Finish**

Click Finish to complete the deployment. This may take a few moments to complete.

**Additional configuration**

In order to work successfully, this application may require additional configuration. Do you

**Yes, take me to the deployment's configuration screen.**

No, I will review the configuration later.

**Summary**

**Deployment:** C:\Users\iscadmin\Desktop\Liebsoft\WebLogicIntegration\WAR.w

**Name:** LiebsoftWebLogicIntegrationWAR

**Staging mode:** Use the defaults defined by the chosen targets

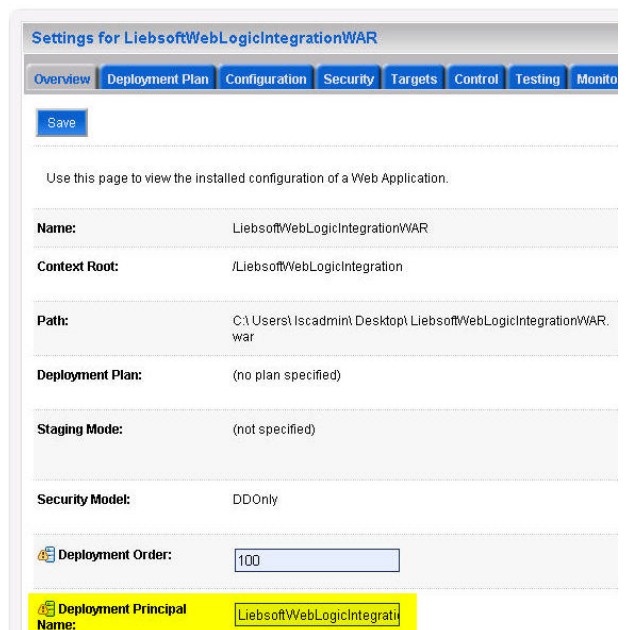
**Security Model:** DDOnly: Use only roles and policies that are defined in the depl

[Customize this table](#)

**Target Summary**

Components
LiebsoftWebLogicIntegrationWAR

8. Supply the **Deployment Principal Name** as **LiebsoftWebLogicIntegrationEAR** then click **Save**.



**Settings for LiebsoftWebLogicIntegrationWAR**

Overview Deployment Plan Configuration Security Targets Control Testing Monitor

Save

Use this page to view the installed configuration of a Web Application.

**Name:** LiebsoftWebLogicIntegrationWAR

**Context Root:** /LiebsoftWebLogicIntegration

**Path:** C:\Users\iscadmin\Desktop\LiebsoftWebLogicIntegration\WAR.war

**Deployment Plan:** (no plan specified)

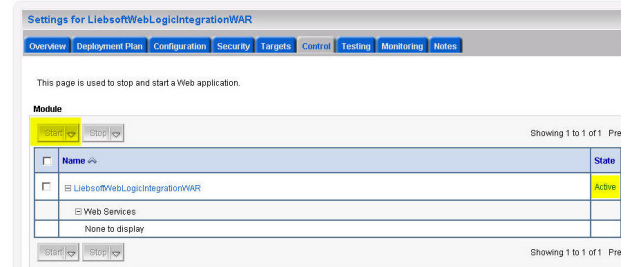
**Staging Mode:** (not specified)

**Security Model:** DDOnly

**Deployment Order:** 100

**Deployment Principal Name:** LiebsoftWebLogicIntegrati

- From the **Control** tab, elect to **start** the application now.

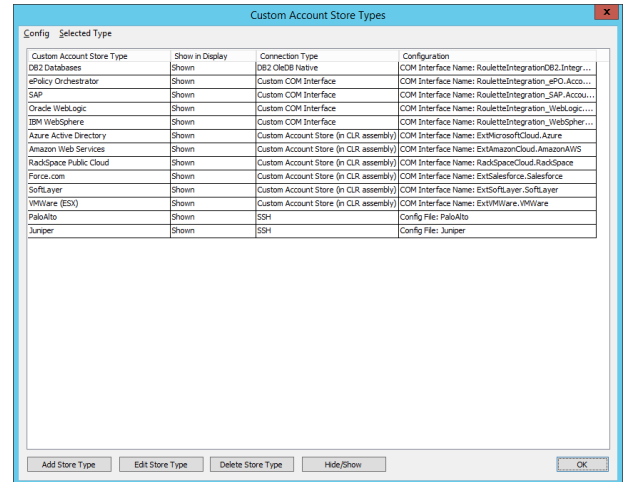


## Enroll Oracle PeopleSoft

Privileged Identity can manage certain versions of Oracle PeopleSoft. PeopleSoft is configured using the custom account stores and CLR configurations.

## Enroll IBM WebSphere

IBM WebSphere is not a default type within Privileged Identity. To make the node visible in the **Account Store** view, open the **SystemsList** menu then open **Custom Account Store Types**. If IBM WebSphere is not listed as a custom account store type, click **Register Built-in Types** from the **Config** menu. The dialog may not respond for a short period of time (up to 60 seconds) while the necessary COM registrations are created in Windows and then registered in Privileged Identity. Once the registration is complete, IBM WebSphere will be listed as a custom account store type.



## To Enroll an IBM Web Logic Instance

### ! IMPORTANT!

Before management of a WebSphere instance can occur, additional steps must be taken in the target WebLogic instance. See "[Configure IBM WebSphere for Enrollment](#)" on page 137 for more information.

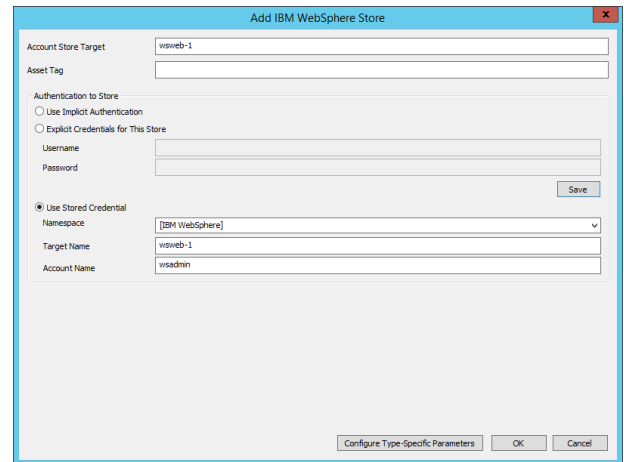
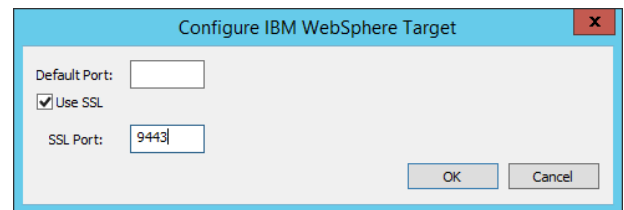
- With PowerShell: **New-LSSystemInManagementSetCustom**.
- Web service URI: **ManagementSetOps\_AddCustomInstanceToManagementSet**.
- Web service REST: **ManagementSet/System/Custom** as a POST.
- Right-click on the **IBM WebSphere** node and import from a comma-delimited text file. Each system should be on its own line. This is a comma-separated line-delimited file where the expected format is: `TargetName,username,password,namespace,assetTag` where `TargetName` is the target WebSphere server. For example:

```
wweb-1,wsadmin,IHeartIBMWs,,ws-host-1
wweb-2,ws-dir-adm,,[LDAP],ws-host-2
```

Notice in the above example, the first line uses explicit authentication and thus leaves the namespace attribute blank. The second line uses a stored managed account and thus leaves the password field blank and then specifies the namespace of the target account. The commas representing the required values are still present as place holders for those columns. See "[Namespace Values](#)" on page 589 for more information on available namespaces.

You must still edit the account store once the import is complete and click **Configure Type Specific Parameters** and specify the connection port and SSL configurations.

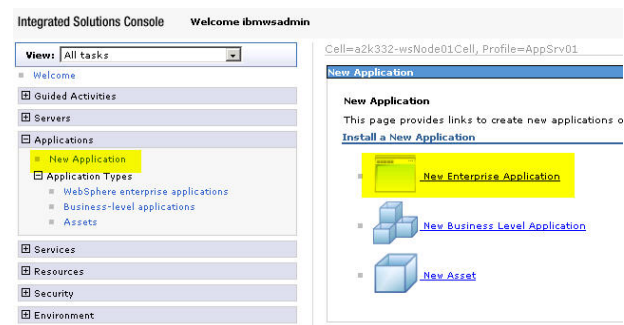
- Right-click on the IBM WebSphere node and select **Add IBM WebSphere**. The required elements are:
  - **Account Store Target** - the name of the IBM Web Logic host server.
  - **Asset Tag** - optional - supply an asset tag for this entry.
  - **Authentication to Store** - select how to authenticate to the IBM WebSphere Instance using one of the following options.
    - **Use Implicit Authentication** - Selecting this option will use the credentials of the interactive user for interactive operations or the deferred/zone processor credentials for scheduled operations. This is not typical for managing IBM WebSphere installations.
    - **Explicit Credentials for this Store** - Use this option to use WebSphere explicit login accounts to connect to the WebSphere instance. These credentials will be saved for future connections and added to the secure password store which will also make them available for retrieval via the web or web service interface.
    - **Use Stored Credential** - use a credential that has already been stored (managed) from the secure password store.
- Then click **Configure Type-Specific Parameters**. This is the configuration dialog for connecting to the WebSphere instance.
- Identify the management port. If SSL is to be used, select the Use SSL option and supply the correct SSL port.

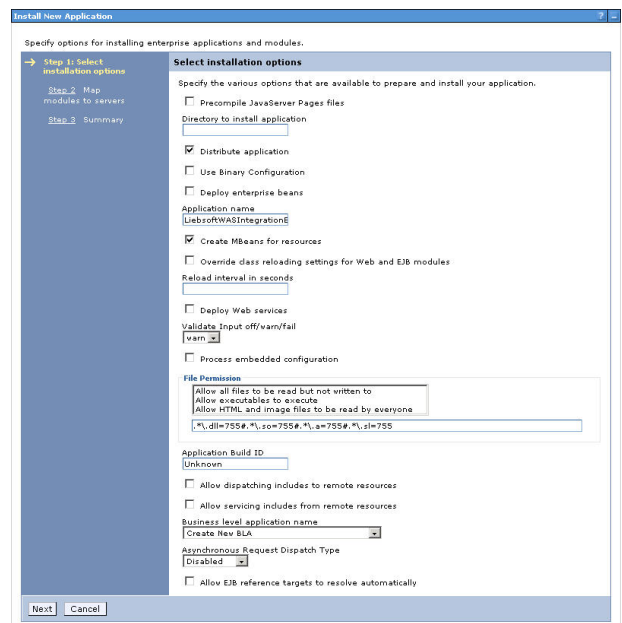
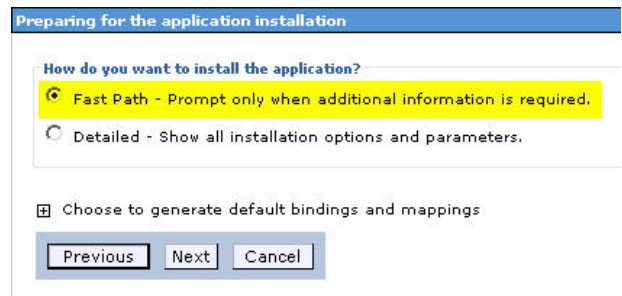
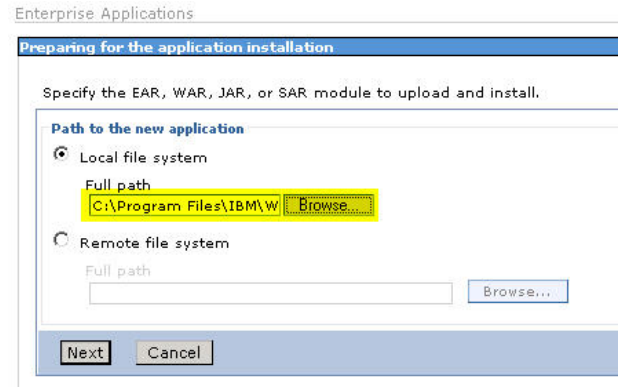
## Configure IBM WebSphere for Enrollment

This integration is supported on WebSphere v 7.x and later and Java JDK 1.6 and later.

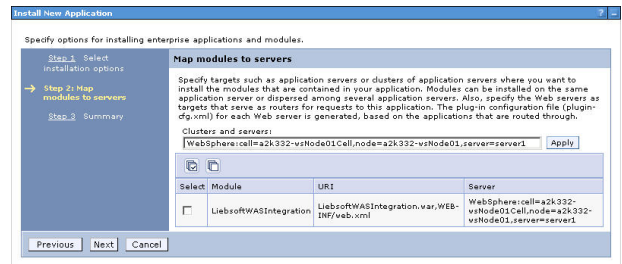
1. Log into WebSphere as a user which can install Enterprise Applications, users, and assign administrative roles.
2. From the **Actions** pane, select **Applications > New Application** then click **New Enterprise Application**.



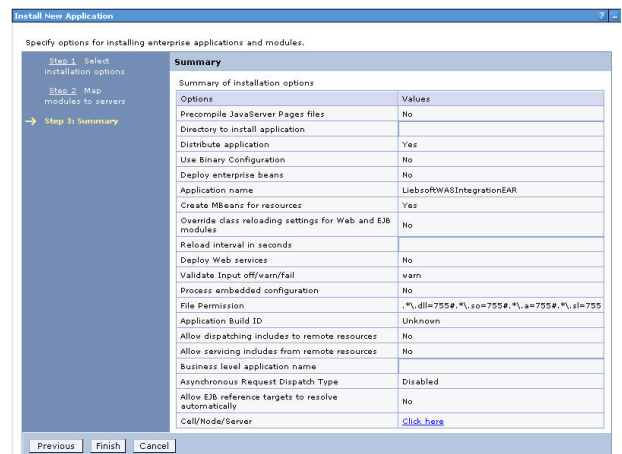
3. Copy **LiebssoftWASIntegrationEAR.ear** from the **SupplementalInstallers** directory of the Privileged Identity installation directory to a local path on the WebSphere Server. Enter the full path to the **LiebssoftWASIntegrationEAR.ear** file then click **Next**.
  
4. Select **Fast Path - Prompt only when additional information is required** then click **Next**.
  
5. On the **Select Installation Options** page, no changes are necessary, click **Next**.



6. On the **Map modules to servers**, no changes are necessary, click **Next**.



7. On the **Summary** page, click **Finish**.



8. On the installation summary page, towards the end of the installation log, note that message "ADMA50131" Application LiebsoftWASIntegrationEAR installed successfully" is received. Then click **Save**.

ADMA5011: The cleanup of the temp directory for application LiebsoftWASIntegrationEAR is complete.

**ADMA50131: Application LiebsoftWASIntegrationEAR installed successfully**

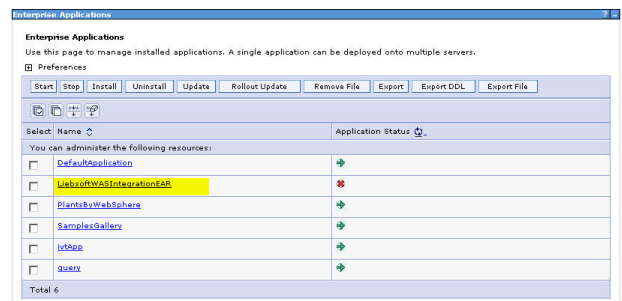
Application LiebsoftWASIntegrationEAR installed successfully.

To start the application, first save changes to the master configuration.

Changes have been made to your local configuration. You can:

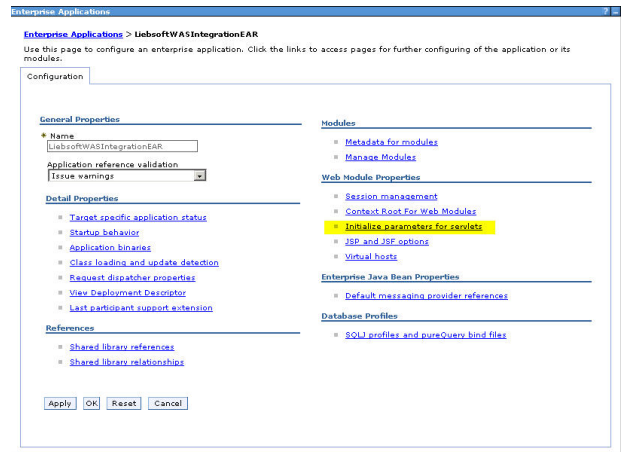
- **Save directly to the master configuration**,
- [Review](#) changes before saving or discarding.

9. From the **Actions** pane, click **Applications > Application Types > Websphere enterprise applications** then click **LiebsoftWASIntegrationEAR** to edit it.





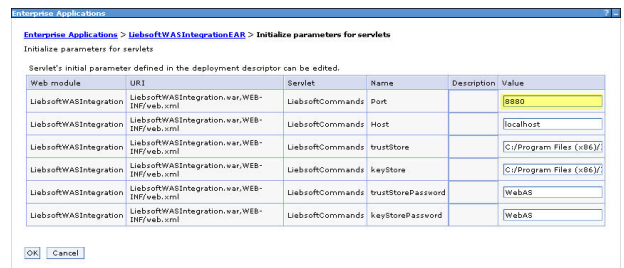
- On the **Configuration** page under **Web Module Properties**, click **Initialize parameters for servlets**.



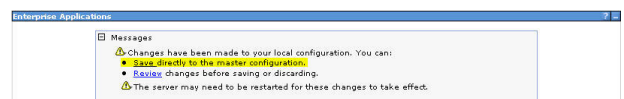
- The following information must be modified:

- Port** = Set this value to be the same as the WebSphere **SOAP\_CONNECTOR\_PORT**, typically 8880 of the WebSphere application server.
- trustStore** = Set TrustStore to location of **DummyClientTrustFile.jks** for the WS server hosting the app. In this installation example, the path was "C:\Program Files\IBM\WebSphere\AppServer\profiles\AppSrv01\etc\DummyClientTrustFile.jks"
- keyStore** = Set KeyStore to location of **DummyClientKeyFile.jks** for the WS server hosting the app. In this installation example, the path was "C:\Program Files\IBM\WebSphere\AppServer\profiles\AppSrv01\etc\DummyClientKeyFile.jks"

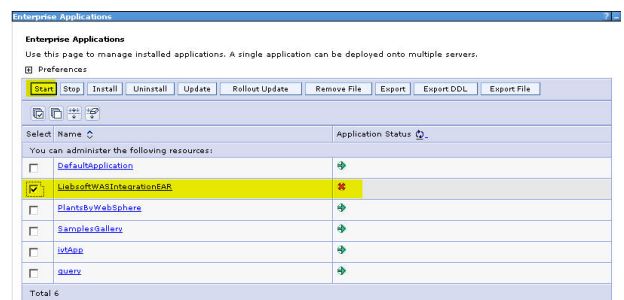
- Click **OK**.



- Click **Save** to save the changes.



- From the list of **Enterprise Applications**, select **LielsortWASIntegrationEAR** then click **Start** to start the application.





## Enroll Network Devices

This section documents how to enroll various network devices. Some of the devices mentioned have their own node type in the Account Store View, and you can add other mentioned types to these nodes or other nodes based on your functional requirements.

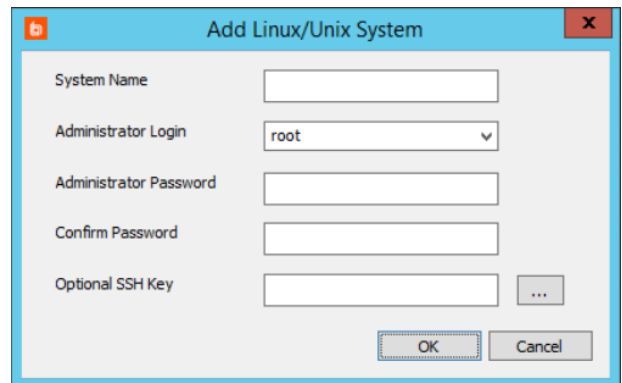
While the nodes do have names indicating what they could be used for, the nodes are not type-definitive. If you have a device or OS type that does not have a dedicated node type, you can add it to any node that supports an SSH or Telnet option or create a custom account store, however, you may need to modify the response file to support the platform.

Account discovery is not supported for most network devices at this time.

## Enroll CheckPoint Devices

CheckPoint does not have its own unique node. Privileged Identity ships with a response file configured for the Linux/Unix node. To use the supplied answer file for management, use the Linux/Unix node or create a custom account store. Create a custom account store if it is desired to have a CheckPoint node. This page describes adding the CheckPoint device as a Linux/Unix system.

- Via API:
  - With PowerShell: **New-LSSystemInManagementSetLinux**.
  - Web service URI: **ManagementSetOps\_AddLinuxSystemToManagementSet**.
  - Web service REST: **ManagementSet/System/Linux** as a POST.
- Use any of the dynamic discovery options such as LDAP directories from the **Management Set Properties** dialog.



**i** Please see *"Create Management Sets"* on page 64 for more information on adding Linux/Unix-based systems.

- Right-click on the **Linux/Unix System** node and click **Import Systems from Text File** to import from a line delimited file.
- Right-click on the **Linux/Unix Systems** node and select **Add Linux system**.
  - **System Name:** (Required) The name or IP address of the system.
  - **Administrator Login:** Select or enter the admin account.
  - **Administrator Password:** Enter the password for the designated admin account.
  - **Optional SSH Key:** If SSH keys have already been added, select which to use.

Now that the device is added, it can be managed.

## Enroll Cisco Devices

To add a device that follows the Cisco general format, use the Cisco node.

### To Enroll a Cisco Device With API

- With PowerShell: **New-LSSystemInManagementSetCisco**.
- Web service URI: **ManagementSetOps\_AddCiscoSystemToManagementSet**.
- Web service REST: **ManagementSet/System/Cisco** as a POST.

### To Enroll a Cisco Device from a Text File

Right-click on the **Cisco** node and select **Import Cisco Devices from Text File**. Browse to and select the text file in which you have configured your device information. Each system should be on its own line. This is a comma-separated line-delimited file where the expected format is:

`SystemName,AssetTag,AdministratorLogin,SSHKeyLabel,Password,AAEnabled.EnablePassword,SSHEnabled`. For example:

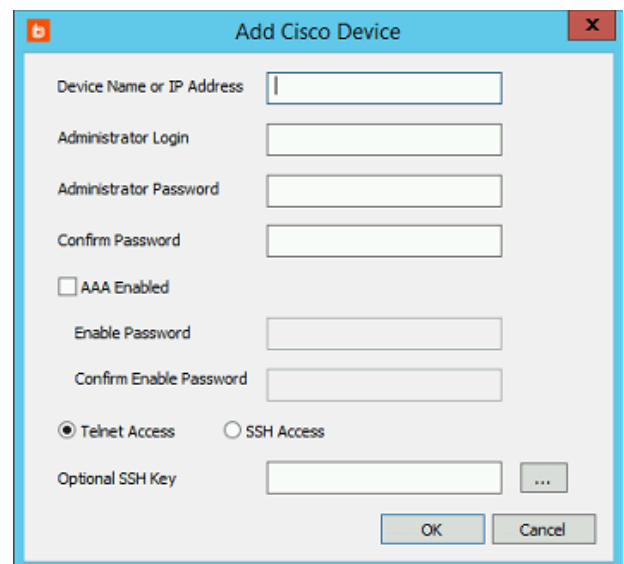
```
cios-01,c7200-abcdefg
casa-02,c7200-ghi,hprotagonist,,password,false,,false
```

Notice in the above example, the second line has no SSH Key Label or Enable Password values, but the commas are still present as placeholders for those columns.

You will be presented an Import **Preview/Config** dialog window. Click OK to accept the column mappings.

### To Enroll a Cisco Device Using the Dialog

- Right-click on the **Cisco** node and select **Add Cisco**. The following fields are available:
  - **Device Name or IP address** - The name or IP address of the system.
  - **Administrator Login** - The administrator's login id.
  - **Administrator Password** - The administrator's login password.
  - **Confirm Password** - Enter the administrator's login password again.
  - **AAA Enabled** checkbox - If the AAA security mechanism is enabled on the Cisco device, check this box.
    - **Enable Password** - The AAA enable password on the Cisco device. This option becomes available when the **AAA Enabled** option is checked.
    - **Confirm Enable Password** - Enter the AAA enable password again.



- **Telnet Access** and **SSH Access** radio buttons - Specify whether the connection will be made via Telnet or SSH.
- **Optional SSH Key** - Enter the SSH private key label, or click the ... button and select the private key from a dialog.

Account and SSH key discovery is supported for Cisco devices.

## Enrolling Dell Remote Access Control (DRAC) Devices

The DRAC node uses SSH to manage DRAC devices. This node type is limited in overall functionality. It is recommended to use the IPMI to locate and manage DRAC devices rather than the DRAC node. When added to the DRAC node, account management will be done using the RACADM commands.

Account discovery is not supported for DRAC when managed via SSH. To support account discovery, add the DRAC devices under the IPMI node.

### To Enroll a Dell DRAC device

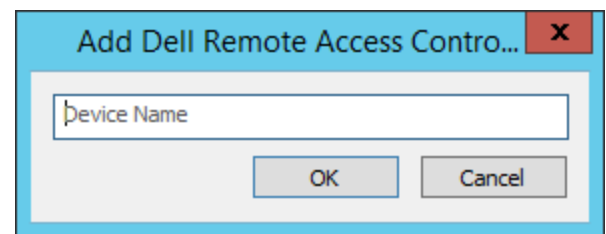
- With PowerShell: **New-LSSystemInManagementSetDRAC**.
- Web service URI: **ManagementSetOps\_AddDRACSystemToManagementSet**.
- Web service REST: **ManagementSet/System/DRAC** as a POST.
- Right-click on the **DRAC Nodes** node and import from a text file. Each system should be on its own line. This is a comma-separated line-delimited file where the expected format is:  
`systemname,assetTag`. For example:

```
drac-01,blade-chassis-01
drac-02,
```

Notice in the above example, the second line has no asset tag but the commas are still present as place holders for those columns.

- Right-click on the **DRAC Nodes** node and select **Add Dell Remote Access Control Node**. The required elements are:
  - **Name or IP address** - The name or IP address of the system

Now that the device is added, it can be managed.

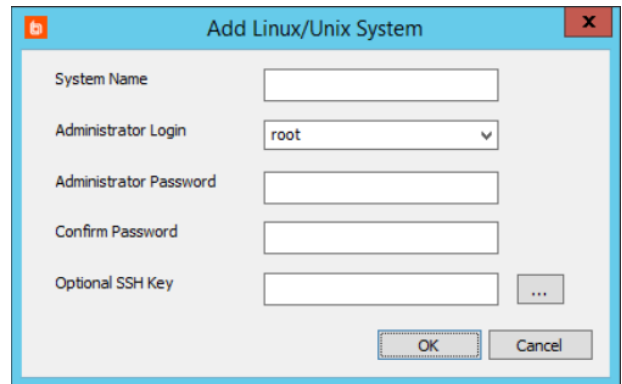


## Enroll F5 Devices

F5 does not have its own unique node. Privileged Identity ships with a response file configured for the Linux/Unix node. To use the supplied answer file for management, use the Linux/Unix node or create a custom account store. Create a custom account store if it is desired to have an F5 node. This page describes adding the F5 device as a Linux/Unix system.

### To Enroll an F5 device

- Via API:
  - With PowerShell: **New-LSSystemInManagementSetLinux**.
  - Web service URI: **ManagementSetOps\_AddLinuxSystemToManagementSet**.
  - Web service REST: **ManagementSet/System/Linux** as a POST.
- Use any of the dynamic discovery options such as LDAP directories from the **Management Set Properties** dialog.



**i** Please see *"Create Management Sets"* on page 64 for more information on adding Linux/Unix-based systems.

- Right-click on the **Linux/Unix System** node and click **Import Systems from Text File** to import from a line delimited file.
- Right-click on the **Linux/Unix Systems** node and select **Add Linux system**.
  - **System Name:** (Required) The name or IP address of the system.
  - **Administrator Login:** Select or enter the admin account.
  - **Administrator Password:** Enter the password for the designated admin account.
  - **Optional SSH Key:** If SSH keys have already been added, select which to use.

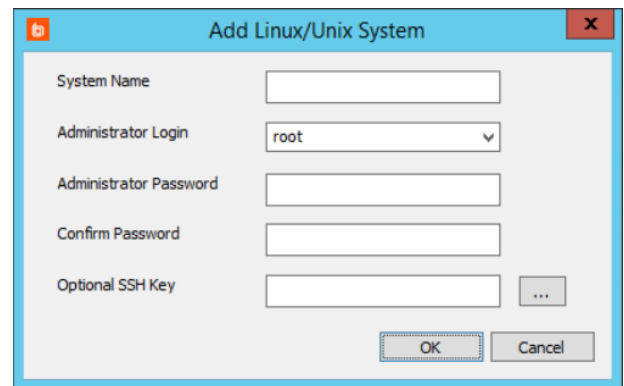
Now that the device is added, it can be managed.

## Enroll Fortigate Devices

Fortigate does not have its own unique node. Privileged Identity ships with a response file configured for the Linux/Unix node. To use the supplied answer file for management, use the Linux/Unix node or create a custom account store. Create a custom account store if it is desired to have an Fortigate node. This page describes adding the Fortigate device as a Linux/Unix system.

### To Enroll a Fortigate device

- Via API:
  - With PowerShell: **New-LSSystemInManagementSetLinux**.
  - Web service URI: **ManagementSetOps\_AddLinuxSystemToManagementSet**.
  - Web service REST: **ManagementSet/System/Linux** as a POST.
- Use any of the dynamic discovery options such as LDAP directories from the **Management Set Properties** dialog.



**i** Please see *"Create Management Sets"* on page 64 for more information on adding Linux/Unix-based systems.

- Right-click on the **Linux/Unix System** node and click **Import Systems from Text File** to import from a line delimited file.
- Right-click on the **Linux/Unix Systems** node and select **Add Linux system**.
  - **System Name:** (Required) The name or IP address of the system.
  - **Administrator Login:** Select or enter the admin account.
  - **Administrator Password:** Enter the password for the designated admin account.
  - **Optional SSH Key:** If SSH keys have already been added, select which to use.

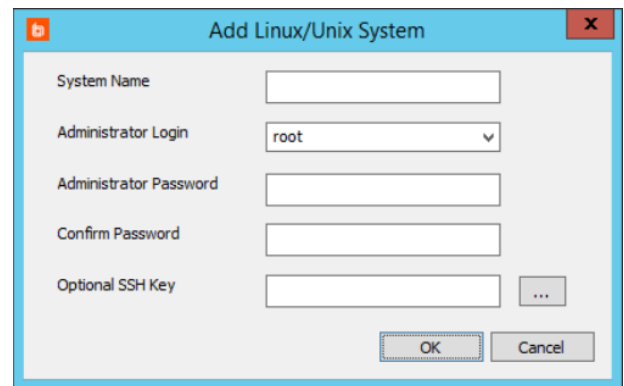
Now that the device is added, it can be managed.

## Enroll Foundry Devices

Foundry does not have its own unique node. Privileged Identity ships with a response file configured for the Linux/Unix node. To use the supplied answer file for management, use the Linux/Unix node or create a custom account store. Create a custom account store if it is desired to have an Foundry node. This page describes adding the Foundry device as a Linux/Unix system.

### To Enroll a Foundry device

- Via API:
  - With PowerShell: **New-LSSystemInManagementSetLinux**.
  - Web service URI: **ManagementSetOps\_AddLinuxSystemToManagementSet**.
  - Web service REST: **ManagementSet/System/Linux** as a POST.
- Use any of the dynamic discovery options such as LDAP directories from the **Management Set Properties** dialog.



**i** Please see *"Create Management Sets"* on page 64 for more information on adding Linux/Unix-based systems.

- Right-click on the **Linux/Unix System** node and click **Import Systems from Text File** to import from a line delimited file.
- Right-click on the **Linux/Unix Systems** node and select **Add Linux system**.
  - **System Name:** (Required) The name or IP address of the system.
  - **Administrator Login:** Select or enter the admin account.
  - **Administrator Password:** Enter the password for the designated admin account.
  - **Optional SSH Key:** If SSH keys have already been added, select which to use.

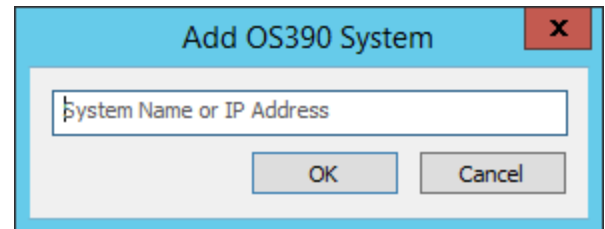
Now that the device is added, it can be managed.

## Enroll HP Procurve Devices

HP Procurve does not have its own unique node. Privileged Identity ships with a response file configured for the OS390 node. To use the supplied answer file for management, use the **OS/390 Mainframes**. Create a custom account store if it is desired to have an HP Procurve node. Taking this action will require you to create a custom response file. This page describes adding the HP Procurve device as an OS/390 system.

### To Enroll an HP Procurve System as an OS/390 System

- Via API:
  - With PowerShell: **New-LSSystemInManagementSetOS390**.
  - Web service SOAP: **ManagementSetOps\_AddOS390SystemToManagementSet**.
  - Web service REST: **ManagementSet/System/OS390** as a POST.
- Right-click on the **OS/390 Mainframes** node and import from a line delimited text file.
- Right-click on the **OS/390 Mainframes** node and select **Add OS/390 Mainframe**. Enter the name or IP address of the system.



Now that the device is added, it can be managed.

Alternatively, OS/390 systems can be added via any of the dynamic discovery options and then have their system type set after being added. To change a system type, right-click on the system(s) and select **Set System Type**. Choose the appropriate value from the list.



## Enroll IPMI Devices

IPMI is an open protocol (running on UDP port 623) that is supported by many lights out (out of band, BMC, lights out) devices such as those supplied by Dell, HP, Sun, SuperMicro, etc. Using the IPMI node to manage these devices allows for many additional features not permitted by SSH/Telnet-based management such as retrieval of host OS and chassis information, listing of local user credentials, and power management of the chassis.

It is recommended to use the IPMI node for all IPMI compatible devices.

There are certain configurations to be cognizant of:

- IPMI over LAN must be enabled on the device. It is not always enabled by default.
- HP iLO 2 require BIOS version 2.05 to be compliant with the IPMI spec.
- Use of IPMI Utilities (free on the internet and originally supplied by Intel) is a very helpful suite of utilities for troubleshooting IPMI connectivity.

### To Enroll an IPMI device

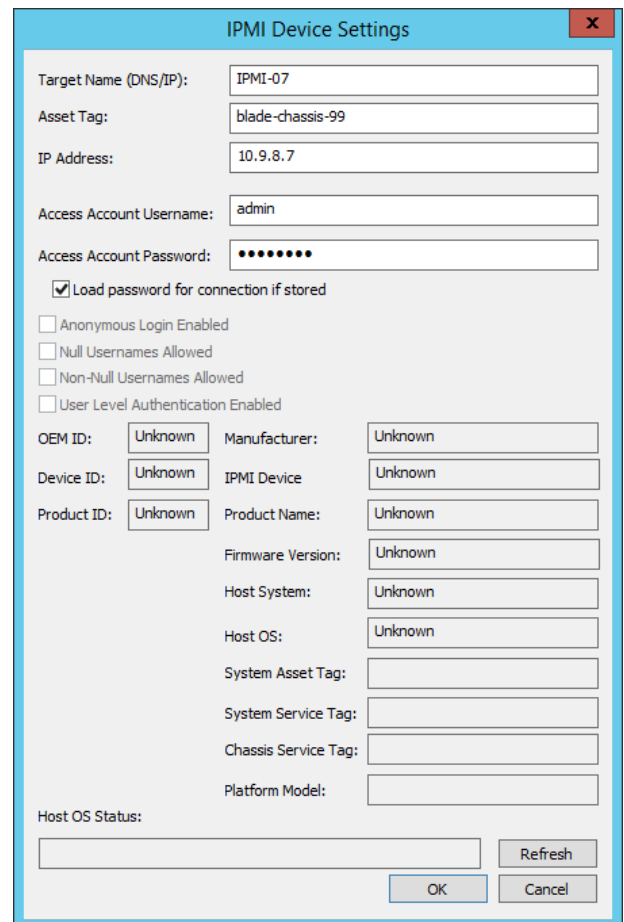
- With PowerShell: **New-LSSystemInManagementSetIPMI**.
- Web service URI: **ManagementSetOps\_AddIPMIDeviceToManagementSet**.
- Web service REST: **ManagementSet/System/IPMI** as a POST.
- Right-click on the **IPMI Devices** node and select **Import IPMI Devices from a text file**. Each system should be on its own line. This is a comma-separated line-delimited file where the expected format is: `systemname,username,password,assetTag`. For example:

```
drac-01,root,password,blade-chassis-01
```


```
hpilo-01,admin,password,
```

Notice in the above example, the second line has no asset tag but the commas are still present as place holders for those columns.

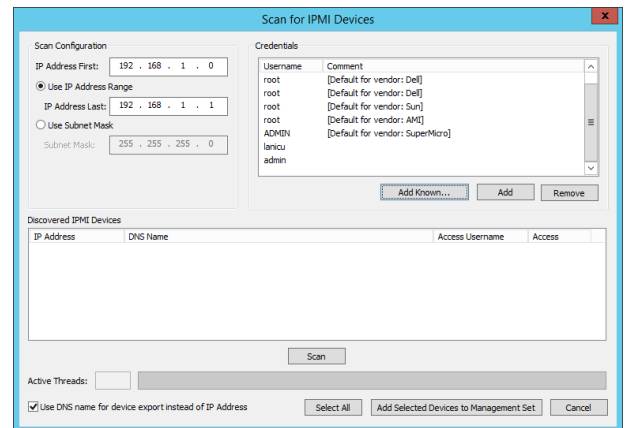
- Right-click on the **IPMI Devices** node and select **Add IPMI Device by Name**. The elements for **Add IPMI Device by Name** are:
- **Required - Target Name (DNS/IP)** - The name or IP address of the device
- **Optional - IP Address**
- **Optional - Access Account Username**
- **Optional - Access Account Password**
- **Optional - Anonymous Login Enabled**
- **Optional - Null Usernames Allowed**
- **Optional - Non-Null Usernames Allowed**



- **Optional - User Level Authentication Enabled**

 **Note:** *The other elements which cannot be edited such as OEM ID, etc., will be read from the device upon device refresh once successfully added.*

- Right-click on the IPMI Devices node and select **Scan for IPMI Devices**. An address range to scan must be provided. The product will also attempt to connect to the system using the well known credentials in the top-right quadrant of the dialog. If the credentials do not work or none are provided, the scan will proceed and will list all devices found responding to IPMI, but will be unable to enumerate any information until the device is edited to include the proper connection credentials. To begin the scan, click the **Scan** button in the lower-right corner.
- Once the scan is complete and the devices are listed, select the devices to add and click **Add Selected Devices to System Set**. The devices along with any relevant connection information will be added to the IPMI Devices node of the current system set.



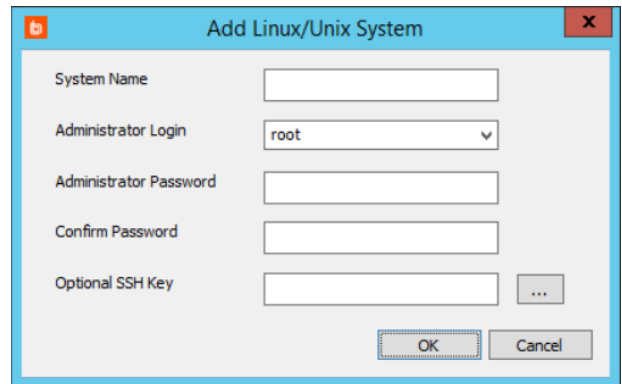
Now that the device is added, it can be refreshed and managed. To refresh the IPMI device, right-click on the device(s) and select **Refresh device information** or hit **F5**. To view the properties once discovered, right-click on the device and select **View/Change IPMI Device Settings**.

## Enroll Juniper Devices

Juniper does not have its own unique node. Privileged Identity ships with a response file configured for the Linux/Unix node. To use the supplied answer file for management, use the Linux/Unix node or create a custom account store. Create a custom account store if it is desired to have an Juniper node. This page describes adding the Juniper device as a Linux/Unix system.

### To Enroll a Juniper device

- Via API:
  - With PowerShell: **New-LSSystemInManagementSetLinux**.
  - Web service URI: **ManagementSetOps\_AddLinuxSystemToManagementSet**.
  - Web service REST: **ManagementSet/System/Linux** as a POST.
- Use any of the dynamic discovery options such as LDAP directories from the **Management Set Properties** dialog.



**i** Please see *"Create Management Sets"* on page 64 for more information on adding Linux/Unix-based systems.

- Right-click on the **Linux/Unix System** node and click **Import Systems from Text File** to import from a line delimited file.
- Right-click on the **Linux/Unix Systems** node and select **Add Linux system**.
  - **System Name:** (Required) The name or IP address of the system.
  - **Administrator Login:** Select or enter the admin account.
  - **Administrator Password:** Enter the password for the designated admin account.
  - **Optional SSH Key:** If SSH keys have already been added, select which to use.

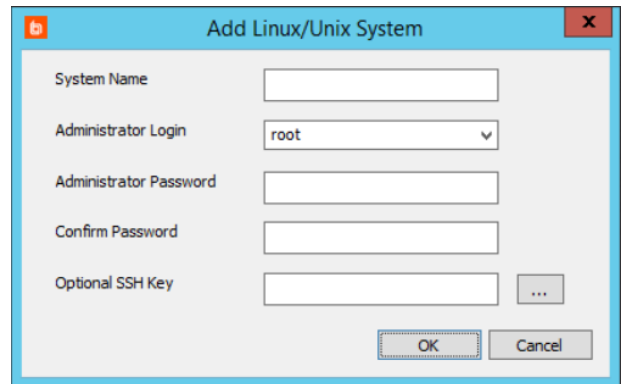
Now that the device is added, it can be managed.

## Enroll NetApp Devices

NetApp does not have its own unique node. Privileged Identity ships with a response file configured for the Linux/Unix node. To use the supplied answer file for management, use the Linux/Unix node or create a custom account store. Create a custom account store if it is desired to have a NetApp node. This page describes adding the NetApp device as a Linux/Unix system.

### To Enroll a NetApp device

- Via API:
  - With PowerShell: **New-LSSystemInManagementSetLinux**.
  - Web service URI: **ManagementSetOps\_AddLinuxSystemToManagementSet**.
  - Web service REST: **ManagementSet/System/Linux** as a POST.
- Use any of the dynamic discovery options such as LDAP directories from the **Management Set Properties** dialog.



**i** Please see *"Create Management Sets"* on page 64 for more information on adding Linux/Unix-based systems.

- Right-click on the **Linux/Unix System** node and click **Import Systems from Text File** to import from a line delimited file.
- Right-click on the **Linux/Unix Systems** node and select **Add Linux system**.
  - **System Name:** (Required) The name or IP address of the system.
  - **Administrator Login:** Select or enter the admin account.
  - **Administrator Password:** Enter the password for the designated admin account.
  - **Optional SSH Key:** If SSH keys have already been added, select which to use.

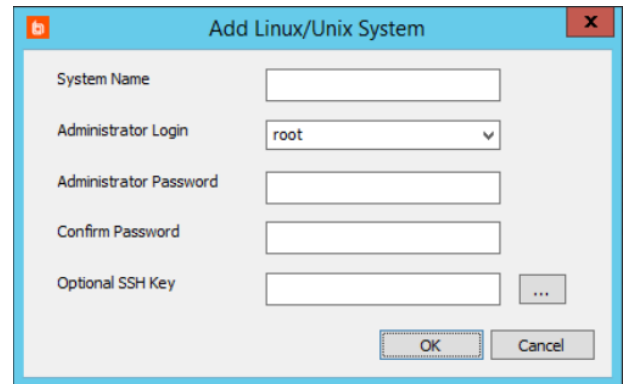
Now that the device is added, it can be managed.

## Enroll Palo Alto Devices

Palo Alto does not have its own unique node. Privileged Identity ships with a response file configured for the Linux/Unix node. To use the supplied answer file for management, use the Linux/Unix node or create a custom account store. Create a custom account store if it is desired to have an Palo Alto node. This page describes adding the Palo Alto device as a Linux/Unix system.

### To Enroll a Palo Alto device

- Via API:
  - With PowerShell: **New-LSSystemInManagementSetLinux**.
  - Web service URI: **ManagementSetOps\_AddLinuxSystemToManagementSet**.
  - Web service REST: **ManagementSet/System/Linux** as a POST.
- Use any of the dynamic discovery options such as LDAP directories from the **Management Set Properties** dialog.



**i** Please see *"Create Management Sets"* on page 64 for more information on adding Linux/Unix-based systems.

- Right-click on the **Linux/Unix System** node and click **Import Systems from Text File** to import from a line delimited file.
- Right-click on the **Linux/Unix Systems** node and select **Add Linux system**.
  - **System Name:** (Required) The name or IP address of the system.
  - **Administrator Login:** Select or enter the admin account.
  - **Administrator Password:** Enter the password for the designated admin account.
  - **Optional SSH Key:** If SSH keys have already been added, select which to use.

Now that the device is added, it can be managed.

## Enroll Xerox Phaser Printers

This topic documents the various ways to enroll Xerox 6700 printers and variations so that Privileged Identity can manage the administrator account name and password.

A Xerox Phaser Printer can be added in the following ways

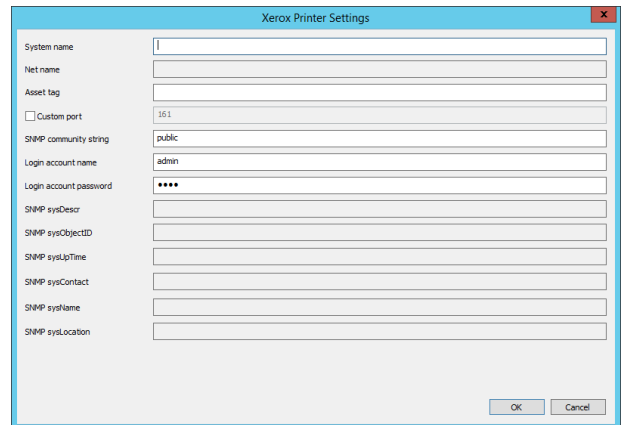
- With PowerShell: **New-LSSystemInManagementSetXeroxPhaserInstance**.
- Web service URI: **ManagementSetOps\_AddXeroxPhaserInstanceToManagementSet**.
- Web service REST: **ManagementSet/System/XeroxPhaser** as a POST.
- Right-click on the **Xerox Phaser Printers** node and import from a text file. Each system should be on its own line. This is a comma-separated line-delimited file where the expected format is:  
`DeviceName,CommunityString,Username,Password,assetTag,port` where `DeviceName` is the target Xerox Phaser printer. For example:

```
xptr-01,public,administrator,IHeartRedIM,,161
xptr-02,public,administrator,P@ssw0rd,xp12345,161
```

Notice in the above example, the first line does not present an asset tag. The commas representing the required values are still present as place holders for those columns.

## To use the Management Console to add a Xerox Printer

1. Open the console and switch to Account Store View.
2. Right-click **Xerox Phaser Printers** and choose **Add Xerox Phaser Printer**.
3. Supply the following information:
  - a. **System Name** - The name of the printer.
  - b. **Net Name** - The IP address or DNS name of the printer.
  - c. **Asset tag** - The asset tag for the printer instance.
  - d. **Custom port** - Select to configure this value for your network. The default SNMP port value is 161.
  - e. **SNMP community string** - Set to public.
  - f. **Login account name** - Enter the device logon name for the administrator account. The default value is **admin**.
  - g. **Login account password** - Enter the password for the administrator account.
  - h. **SNMP sysDescr** - The SNMP system description response. This is a hard-coded response that indicates what type of device this is. You can filter the listings to only show you systems that match the wildcard in the SNMP Match Filter field.
  - i. **SNMP sysObjectID** - The SNMP response that provides a unique, device-provided identifier for this type/version of device.
  - j. **SNMP sysUpTime** - The SNMP calculated response that reports the last time the device was rebooted.
  - k. **SNMP sysContact** - The SNMP per-device configuration that reports who to contact about this device. This value is set by either the user or the IT department.
  - l. **SNMP sysName** - The SNMP per-device configuration that reports the name of this device. This value is set by either the



user or the IT department.

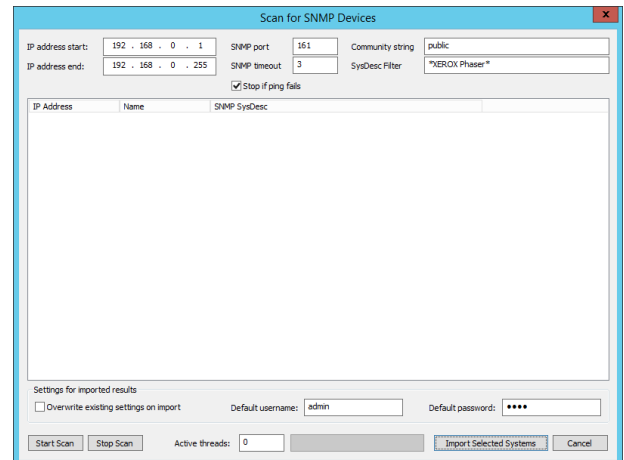
- m. **SNMP sysLocation** - The SNMP per-device configuration that reports where this device is located. This value is set by either the user or the IT department.

4. Click **OK**.

## To Scan for Xerox Printers to add

Xerox printers can also be added by running an SNMP scan of the network.

1. Open the console and switch to Account Store View.
2. Right-click **Xerox Phaser Printers** and choose **Scan for SNMP Devices**.
3. Supply the following scanner parameters:
  - a. Define an IP address range by setting the appropriate starting and stopping addresses in the **IP address start** and **IP address end** fields.
  - b. Change the default SNMP port if required.
  - c. Set an appropriate time-out value in the **SNMP timeout** field. (For example, use a value greater than two seconds if you have systems on a wide-area network that have response times that exceed two seconds.)
  - d. Change the default **Community string** if required.
4. Click **Start Scan**.
5. Once the scan is complete, select one or more printers in the results table and click **Import Selected Systems** to enroll the printers.



# Enroll Cloud Service Providers

This section describes how to enroll cloud service providers such as Microsoft Azure, Amazon Web Services (AWS), RackSpace, Salesforce, and SoftLayer.

## Enroll Amazon Web Services (AWS)

Only native Amazon Web Services (AWS) user accounts stored in AWS Identity and Access Management (IAM) are managed by the Privileged Identity cloud service account store. Federated accounts or imported accounts that may be visible in AWS IAM are not managed.

To enroll Amazon Web Services with Privileged Identity, first add the cloud service providers to the display tree, then add one or more Amazon Web Services directory instances. Finally, automatically populate the account store with system instances. Use the following steps.

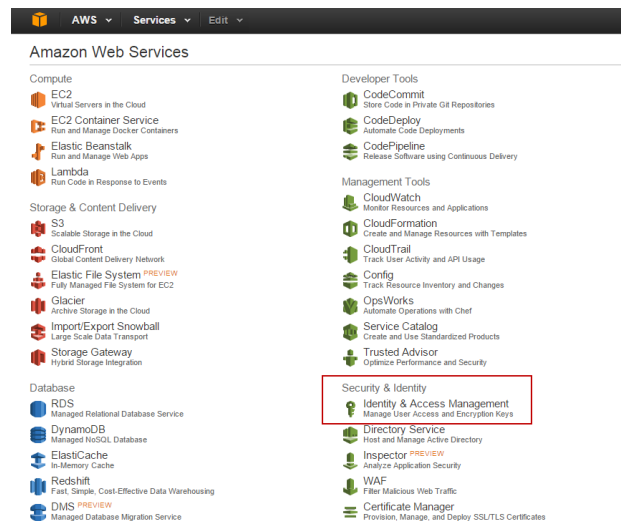
### Add the Cloud Service Provider to the Display Tree

The cloud service providers are added automatically during installation of the solution. This means the cloud providers should already have a management node visible in the Account Store View. However, if there were any issues during installation or upgrade and the cloud provider nodes are not visible in the Account Store View (and you already checked your **Display Options**), please see "Add a Cloud Service Provider Account Store Manually" on page 167.

### Configure Amazon Web Services to Work With Privileged Identity

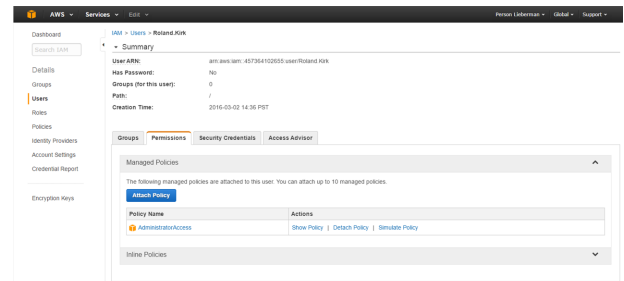
These steps are provided to help you prepare an AWS user account that will be used by Privileged Identity to manage AWS user account passwords.

1. Log into the Amazon Web Services management console, and then click **Identity & Access Management**.





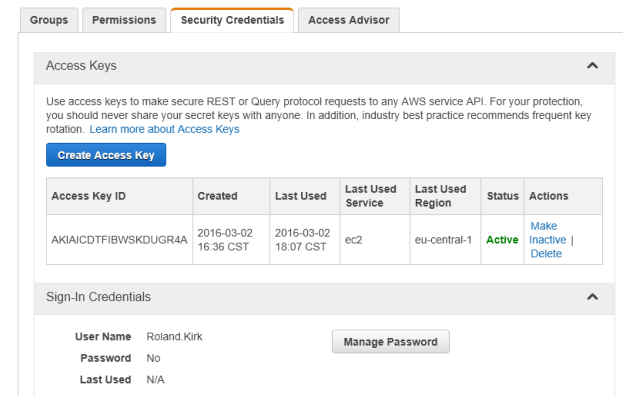
2. Click **Users**.
3. Create the user account that Privileged Identity will use to manage AWS user passwords, or select an existing account.
4. Select the **Permissions** tab. If this is an existing account, verify that either the **AdministratorAccess** policy, or a similar admin-level policy is attached to the user. Otherwise, click **Attach Policy** and attach the admin-level policy to the user.



5. Select the **Security Credentials** tab.

The **Security Credentials** tab lists the access key IDs associated with the user account. In Privileged Identity, when you enter the user credentials that will be used to authenticate to AWS, you will need to enter a valid **Access Key ID** in the **Username** field and the **Secret Access Key** in the **Password** field.

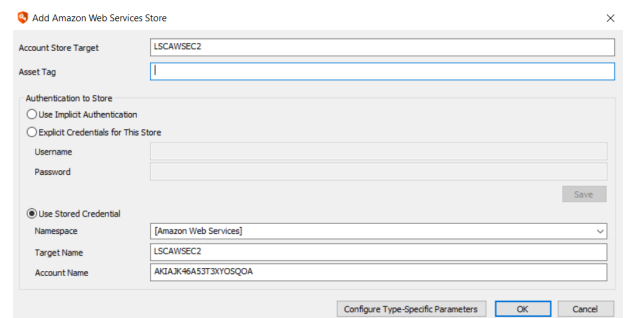
Skip this step if you already downloaded user security credentials for the user account. Otherwise, click **Create Access Key** to create an access key and download the credentials, which you will use in the next section.



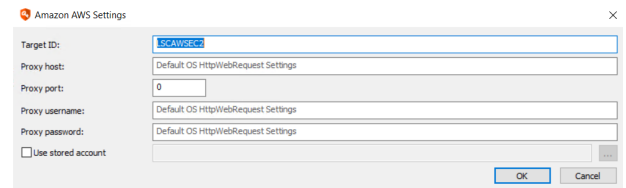
## Add an Amazon Web Services Directory Instance

With the AWS user established, admin policy configured, and the Access Key created, you are ready to add your AWS instance to Privileged Identity.

1. Open the management console, change to the desired management set or create a new one and choose **View > Account Store View**.
2. Right-click the Amazon Web Services account store and choose **Add Amazon Web Services**.
3. Complete the form as follows:
  - **Account Store Target:** Enter a name for the services instance. This name does not need to be anything specific but should be easy to distinguish from other AWS instances when managing multiple instances. Please limit your account store target name to the following characters: a-z, A-Z, 0-9, dot (.), hyphen (-) and underscore (\_).
  - **Asset Tag:** (Optional) If applicable, enter the asset tag for the services instance.
4. Complete the **Authentication to Store** section. Use this section to specify how to authenticate to the instance.
  - **Use Implicit Authentication:** Select only if your environment supports integrated authentication with Amazon Web Services, which is not typical.
  - **Explicit Credentials for This Store:** Enter the account credentials you use to administer this Amazon Web Services instance. For **Username**, enter an active **Access Key ID**, and for **Password**, enter the **Secret Access Key**, and then click **Save**.



- **Use Stored Credential:** Select to use a stored (managed) password that is valid for this Amazon Web Services instance.
    - **Namespace:** Select the namespace that has the stored password.
    - **Target Name:** Enter the name of the target system associated with the stored account.
    - **Account Name:** Enter the name of the stored account to use.
5. If using a proxy connection to connect to the outside world, click **Configure Type-Specific Parameters**, otherwise skip to step 7.
  6. Complete the form.
    - **Target ID:** If not previously entered in step 3, enter the name of the instance. Do not enter a name containing spaces or reserved characters.
    - **Proxy host:** (Optional) If a proxy must be used to connect to the AWS instance, enter the proxy server host name or IP address.
    - **Proxy port:** (Optional) Enter the proxy server port.
    - **Proxy username:** (Optional) Enter a user name if your proxy server requires it. If the credentials required by your proxy server are saved in a stored account, select **Use stored account** instead.
    - **Proxy password:** (Optional) Enter a password if your proxy server requires it.
    - **Use stored account:** (Optional) Select if the credentials required by your proxy server are saved in a stored account. Click the ellipsis (...) to select the account.
  7. Click **OK** to save your changes and close the dialog.




For more information on available namespaces, please see "[Namespace Values](#)" on page 589.

## Enroll Microsoft Azure

Only native Microsoft Azure accounts are managed by the Privileged Identity cloud service account store.

To enroll Microsoft Azure Services with Privileged Identity, first add the cloud service provider to the display tree, and then add one or more Amazon Web Services directory instances. Finally, automatically populate the account store with system instances. Use the following steps.

### Add the Cloud Service Provider to the Display Tree

The cloud service providers are added automatically during installation of the solution. This means the cloud providers should already have a management node visible in the Account Store View. However, if there were any issues during installation or upgrade and the cloud provider nodes are not visible in the Account Store View (and you already checked your **Display Options**), please see "[Add a Cloud Service Provider Account Store Manually](#)" on page 167.

### Configure Microsoft Azure to Work With Privileged Identity

These steps are provided to help you prepare an Azure user account that will be used by Privileged Identity to manage Azure user account passwords.

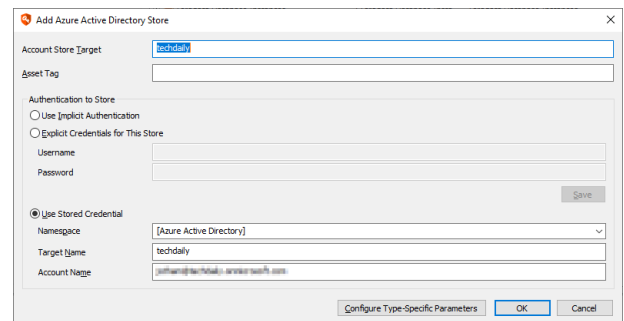
To connect to an Azure AD instance to retrieve a list of Azure Accounts and manage their passwords requires only a valid administrative level Azure AD account configured as a service administrator or co-administrator.

To connect to Azure AD to obtain a list of machines present in the Azure instance (optional functionality) requires a management certificate. Please refer to your current Microsoft Azure documentation for configuring management certificates.

### Add a Microsoft Azure Instance

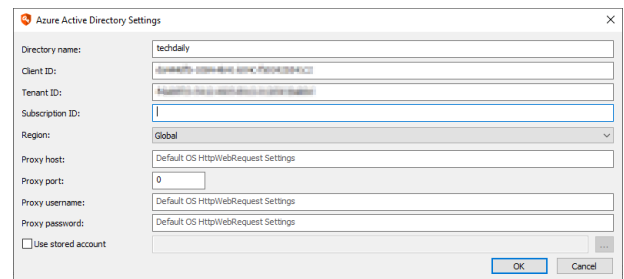
With the Azure user established and admin settings configured, you are ready to add your Azure instance to Privileged Identity.

1. Open the management console, change to the desired management set or create a new one, and then select **View > Account Store View**.
2. Right-click the Azure Active Directory node and choose **Add Azure Active Directory**.
3. Complete the form as follows:
  - **Account Store Target:** Enter a name for the services instance. This name does not need to be anything specific but should be easy to distinguish it from other Microsoft Azure instances when managing multiple instances. Please limit your account store target name to the following characters: a-z, A-Z, 0-9, dot (.), hyphen (-) and underscore (\_).
  - **Asset Tag:** (Optional) If applicable, enter the asset tag for the services instance.
4. Complete the **Authentication to Store** section. Use this section to specify how to authenticate to the instance.
  - **Use Implicit Authentication:** Select only if your environment supports Integrated Windows Authentication from the machine hosting Privileged Identity, not typical.



- **Explicit Credentials for This Store:** Enter the account credentials you use to administer this Azure Active Directory instance. Enter the username and password, then click **Save**.
  - **Use Stored Credential:** Select to use a stored (managed) password that is valid for this Azure Active Directory instance.
    - **Namespace:** Choose the namespace that has the stored password.namespaces.
    - **Target Name:** Type the name of the target system associated with the stored account.
    - **Account Name:** Type the name of the stored account to use.
5. Click **Configure Type-Specific Parameters**. To configure required elements and optional proxy settings.

6. Complete the form.



- **Directory Name:** If not previously entered in step 3, enter the name of the instance. Do not enter a name containing spaces or reserved characters in their names.
  - **Client ID:** Your client ID for the target instance of Azure Active Directory.
  - **Tenant ID:** Your tenant ID for the target instance of Azure Active Directory.
  - **Subscription ID:** Your subscription ID for the target instance of Azure Active Directory.
  - **Region:** By default, this is **Global**, but you can select **China** or **Germany** if required for your Azure instance.
  - **Proxy host:** (Optional) If a proxy must be used to connect to the Azure Active Directory instance, enter the proxy server host name or IP address.
  - **Proxy port:** (Optional) Enter the proxy server port.
  - **Proxy username:** (Optional) Enter a user name if your proxy server requires it. If the credentials required by your proxy server are saved in a stored account, select **Use stored account** instead.
  - **Proxy password:** (Optional) Enter a password if your proxy server requires it.
  - **Use stored account:** (Optional) Select if the credentials required by your proxy server are saved in a stored account. Click the ellipsis (...) to select the account.
7. Click **OK** to save your changes and close the dialog.



For more information on available namespaces, please see "[Namespace Values](#)" on page 589.

## Enroll RackSpace

Only native RackSpace cloud user accounts are managed by the Privileged Identity cloud service account store. Federated accounts or imported accounts that may otherwise be visible in the native RackSpace dashboards are not managed.

To enroll RackSpace with Privileged Identity, first add the cloud service provider to the display tree, and then add one or more RackSpace instances. Finally, automatically populate the account store with system instances. Use the following steps.

### Add the Cloud Service Provider to the Display Tree

The cloud service providers are added automatically during installation of the solution. This means the cloud providers should already have a management node visible in the Account Store View. However, if there were any issues during installation or upgrade and the cloud provider nodes are not visible in the Account Store View (and you already checked your **Display Options**), please see "[Add a Cloud Service Provider Account Store Manually](#)" on page 167.

### Configure RackSpace to Work With Privileged Identity

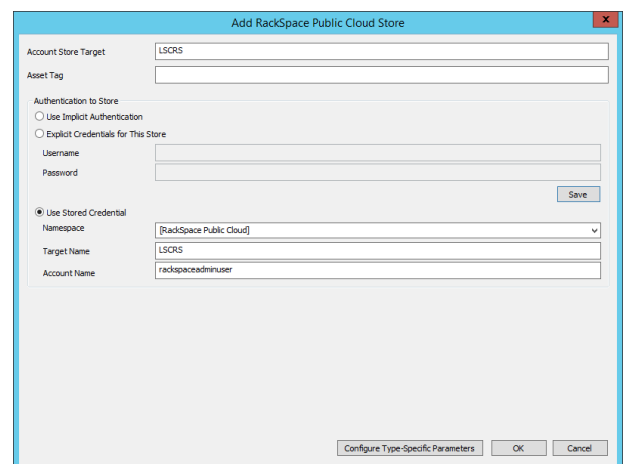
These steps are provided to help you prepare a RackSpace user account that will be used by Privileged Identity to manage other RackSpace user account passwords.

1. Create or designate a user credential with the proper administrative rights to reset user passwords in the RackSpace instance.
2. Note the username and password.

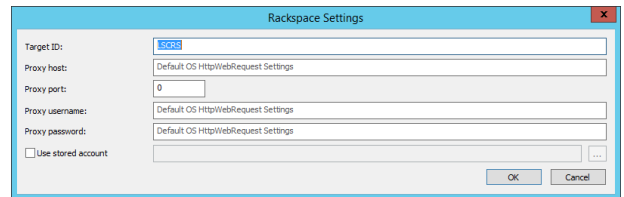
### Add a RackSpace Instance

With the RackSpace user established, and administrative rights established, you are ready to add your RackSpace instance to Privileged Identity.

1. Open the management console, change to the desired management set or create a new one and choose **View | Account Store View**.
2. Right-click the RackSpace Public Cloud node and choose **Add RackSpace Public Cloud**.
3. Complete the form as follows:
  - **Account Store Target:** Enter a name for the services instance. This name does not need to be anything specific but should be easy to distinguish it from other RackSpace instances when managing multiple instances. Please limit your account store target name to the following characters: a-z, A-Z, 0-9, dot (.), hyphen (-) and underscore (\_).
  - **Asset Tag:** (Optional) If applicable, enter the asset tag for the services instance.
4. Complete the **Authentication to Store** section. Use this section to specify how to authenticate to the instance.
  - **Use Implicit Authentication:** Select only if your environment supports integrated authentication with RackSpace, not typical.



- **Explicit Credentials for This Store:** Enter the account credentials you use to administer this RackSpace instance. Enter a **Username**, and **Password**, then click **Save**. The password will be the API key for the user account, not its simple password.
  - **Use Stored Credential:** Select to use a stored (managed) password that is valid for this RackSpace instance.
    - **Namespace:** Choose the namespace that has the stored password.
    - **Target Name:** Type the name of the target system associated with the stored account.
    - **Account Name:** Type the name of the stored account to use.
5. If using a proxy connection to connect to the outside world, Click **Configure Type-Specific Parameters**. Otherwise skip to step 7.
6. Complete the form.
- **Target ID:** If not previously entered in step 3, enter the name of the instance. Do not enter a name containing spaces or reserved characters in their names.
  - **Proxy host:** (Optional) If a proxy must be used to connect to the RackSpace instance, enter the proxy server host name or IP address.
  - **Proxy port:** (Optional) Enter the proxy server port.
  - **Proxy username:** (Optional) Enter a user name if your proxy server requires it. If the credentials required by your proxy server are saved in a stored account, select **Use stored account** instead.
  - **Proxy password:** (Optional) Enter a password if your proxy server requires it.
  - **Use stored account:** (Optional) Select if the credentials required by your proxy server are saved in a stored account. Click the ellipsis (...) to select the account.
7. Click **OK** to save your changes and close the dialog.



For more information on available namespaces, please see "[Namespace Values](#)" on page 589.

## Enroll Salesforce

Only native Salesforce (Force.com) user accounts are managed by the Privileged Identity.

To enroll Salesforce (Force.com) with Privileged Identity, complete the following tasks:

1. Create a Native Client Application in Salesforce.
2. Add the Salesforce directory instance.

Use the following steps.

### Create a Native Client Application in Salesforce

1. Go to the Force.com REST API Developer Guide and follow the steps in the *Defining Connected Apps* topic, located here: [https://developer.salesforce.com/docs/atlas.en-us.api\\_rest.meta/api\\_rest/intro\\_defining\\_remote\\_access\\_applications.htm](https://developer.salesforce.com/docs/atlas.en-us.api_rest.meta/api_rest/intro_defining_remote_access_applications.htm)

Note the following:

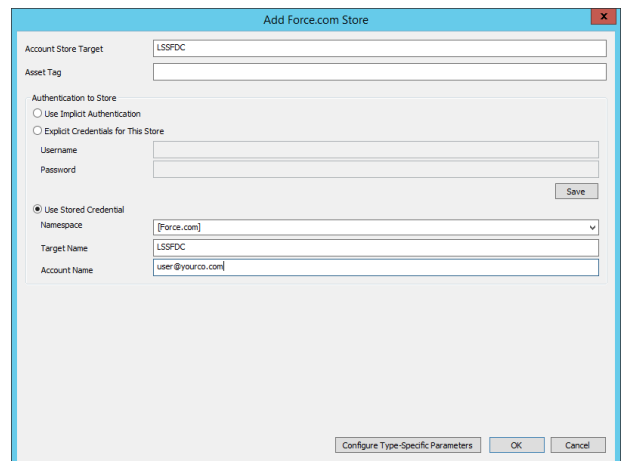
- When entering a name for the application, choose a name that indicates that Privileged Identity is the application that is authenticating to Salesforce
  - When entering the Callback URL, enter the web application URL and specify **PWCWeb/auth/oauth2** as the resource, for example, <https://serverURL/PWCWeb/auth/oauth2>.
2. Copy and save the **Consumer Key** and **Consumer Secret** values. You will need these values when you add the Salesforce directory instance to Privileged Identity.

### Add a Salesforce Directory Instance

Before you add a Salesforce directory instance, the native application must be created.

1. Open the management console, change to the desired management set or create a new one and choose **View > Account Store View**.
2. Right-click the Salesforce (Force.com) node and choose **Add Salesforce**.
3. Complete the form as follows:

- **Account Store Target:** Enter a name for the services instance. This name does not need to be anything specific but should be easy to distinguish it from other Salesforce instances when managing multiple instances. Please limit your account store target name to the following characters: a-z, A-Z, 0-9, dot (.), hyphen (-) and underscore (\_).
- **Asset Tag:** (Optional) If applicable, enter the asset tag for the services instance.



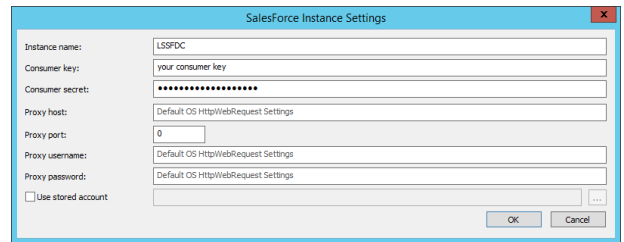
4. Complete the **Authentication to Store** section. Use this section to specify how to authenticate to the instance.
  - **Use Implicit Authentication:** Select only if your environment supports integrated authentication with Salesforce, not typical.

- **Explicit Credentials for This Store:** Enter the account credentials you use to administer this Salesforce instance. Enter a **Username**, and **Password**, then click **Save**.
- **Use Stored Credential:** Select to use a stored (managed) password that is valid for this Salesforce instance.
  - **Namespace:** Choose the namespace that has the stored password.
  - **Target Name:** Type the name of the target system associated with the stored account.
  - **Account Name:** Type the name of the stored account to use.

5. Click **Configure Type-Specific Parameters**.

6. Complete the form.

- **Instance name:** The name of your Salesforce instance. The system auto-populates this field with the **Account Store Target** value. Do not enter a name containing spaces or reserved characters in their names.
- **Consumer key:** The consumer key created by Salesforce when you added the native client application to Salesforce.
- **Consumer secret:** The secret that is paired with the consumer key. The consumer secret is separate from the account password.
- **Proxy host:** (Optional) If a proxy must be used to connect to the RackSpace instance, enter the proxy server host name or IP address.
- **Proxy port:** (Optional) Enter the proxy server port.
- **Proxy username:** (Optional) Enter a user name if your proxy server requires it. If the credentials required by your proxy server are saved in a stored account, select **Use stored account** instead.
- **Proxy password:** (Optional) Enter a password if your proxy server requires it.
- **Use stored account:** (Optional) Select if the credentials required by your proxy server are saved in a stored account. Click the ellipsis (...) to select the account.



7. Click **OK** to save your changes and close the dialog.



For more information on available namespaces, please see *"Namespace Values"* on page 589.



## Enroll Softlayer

Only native SoftLayer cloud user accounts are managed by the Privileged Identity cloud service account store. Federated accounts or imported accounts that may otherwise be visible in the native SoftLayer dashboards are not managed.

To enroll SoftLayer with Privileged Identity, first add the cloud service providers to the display tree, then add one or more SoftLayer instances. Finally, automatically populate the account store with system instances. Use the following steps.

### Add the Cloud Service Providers to the Display Tree

The cloud service providers are added automatically during installation of the solution. This means the cloud providers should already have a management node visible in the Account Store view. However, if there were any issues during installation or upgrade and the cloud provider nodes are not visible in the account store view (and you already checked your Display Options), please see "[Add a Cloud Service Provider Account Store Manually](#)" on page 167.

### Configure SoftLayer to Work With Privileged Identity

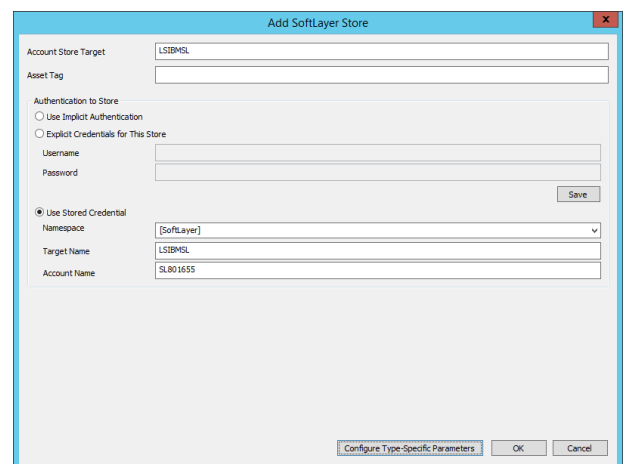
These steps are provided to help you prepare a SoftLayer user account that will be used by Privileged Identity to manage other SoftLayer user account passwords.

1. Create or designate a user credential with the proper administrative rights to reset user passwords in the SoftLayer instance.
2. Note the username and password.

### Add a SoftLayer Instance

With the SoftLayer user established, and administrative rights established, you are ready to add your SoftLayer instance to Privileged Identity.

1. Open the management console, change to the desired management set or create a new one and choose **View | Account Store View**.
2. Right-click the SoftLayer Public Cloud node and choose **Add SoftLayer Public Cloud**.
3. Complete the form as follows:
  - **Account Store Target:** Enter a name for the services instance. This name does not need to be anything specific but should be easy to distinguish it from other SoftLayer instances when managing multiple instances. Please limit your account store target name to the following characters: a-z, A-Z, 0-9, dot (.), hyphen (-) and underscore (\_).
  - **Asset Tag:** (Optional) If applicable, enter the asset tag for the services instance.
4. Complete the **Authentication to Store** section. Use this section to specify how to authenticate to the instance.
  - **Use Implicit Authentication:** Select only if your environment supports integrated authentication with SoftLayer, not typical.

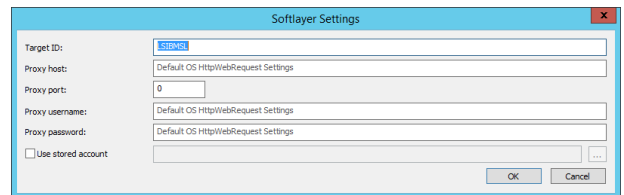


- **Explicit Credentials for This Store:** Enter the account credentials you use to administer this SoftLayer instance. Enter a **Username**, and **Password**, then click **Save**. The password will be the API key for the user account, not its simple password.
- **Use Stored Credential:** Select to use a stored (managed) password that is valid for this SoftLayer instance.
  - **Namespace:** Choose the namespace that has the stored password. See "[Namespace Values](#)" on page 589 for more information on available namespaces.
  - **Target Name:** Type the name of the target system associated with the stored account.
  - **Account Name:** Type the name of the stored account to use.

5. If using a proxy connection to connect to the outside world, Click **Configure Type-Specific Parameters**. Otherwise skip to step 7.

6. Complete the form.

- **Target ID:** If not previously entered in step 3, enter the name of the instance. Do not enter a name containing spaces or reserved characters in their names.
- **Proxy host:** (Optional) If a proxy must be used to connect to the SoftLayer instance, enter the proxy server host name or IP address.
- **Proxy port:** (Optional) Enter the proxy server port.
- **Proxy username:** (Optional) Enter a user name if your proxy server requires it. If the credentials required by your proxy server are saved in a stored account, select **Use stored account** instead.
- **Proxy password:** (Optional) Enter a password if your proxy server requires it.
- **Use stored account:** (Optional) Select if the credentials required by your proxy server are saved in a stored account. Click the ellipsis (...) to select the account.

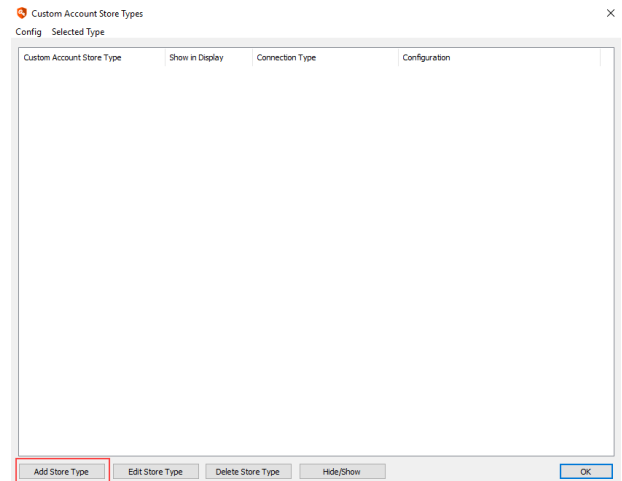


7. Click **OK** to save your changes and close the dialog.

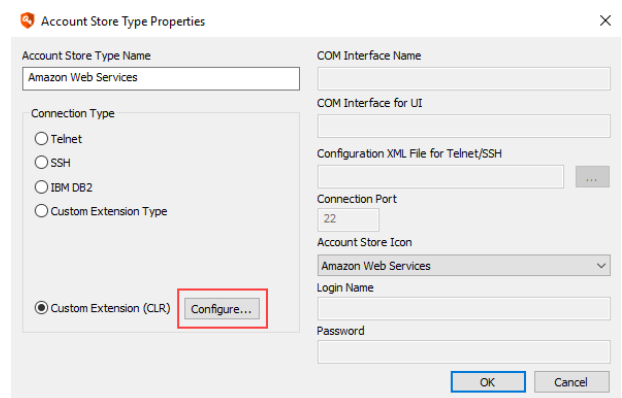
## Add a Cloud Service Provider Account Store Manually

This section describes how to manually add a cloud provider account store to the display tree. This alternate approach allows you to register the account store using a custom name.

1. Open the management console.
2. Select **Systems List > Custom Account Store Types**.
3. Click **Add Store Type**.

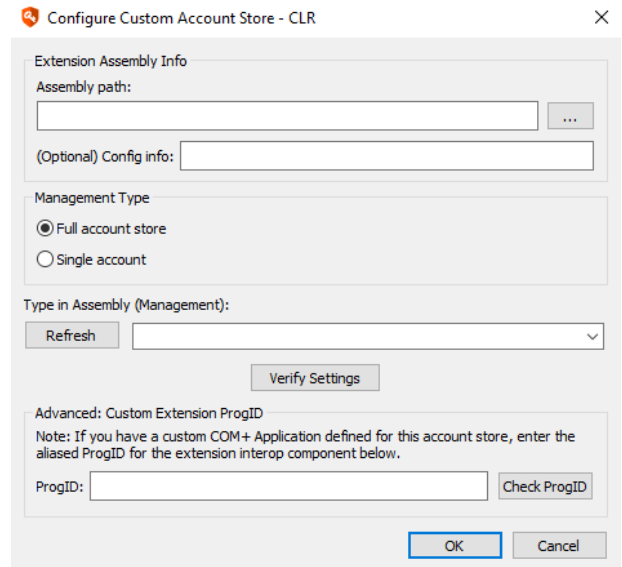


4. Complete the **Account Store Type Properties** dialog form:
  - **Account Store Type Name:** Enter a name for the cloud service provider you are adding, for example **Amazon Web Services**.
  - **Account Store Icon:** Select the appropriate icon for the service provider from the dropdown. This specifies the account store icon to display in the display tree. You cannot add custom icons. You may choose any icon from the list. This icon affects only what is displayed in the tree control.
    - For Amazon, select **Amazon Web Services**.
    - For Microsoft Azure, select **Windows system** or another option (Microsoft Azure does not have its own icon).
    - For RackSpace, select **RackSpace Public Cloud**.
    - For Salesforce, select **SalesForce**.
    - For IBM SoftLayer, select **SoftLayer**.
  - **Connection Type:** Select **Custom Extension (CLR)**, and then click **Configure**.



5. Complete the **Configure Custom Account Store - CLR** dialog form:

- **Assembly path:** Click the ellipses (...) to open the **Locate Assembly** dialog.
  - For Amazon Web Services, navigate to **Install Folder > AccountStoreSupport > AmazonWebServices > ExtAmazonCloud.dll**, and then click **Open** to set the assembly path to **ExtAmazonCloud.dll**.
  - For Microsoft Azure, navigate to **Install Folder > AccountStoreSupport > AzureActiveDirectory > ExtMicrosoftCloud.dll**, and then click **Open** to set the assembly path to **ExtMicrosoftCloud.dll**.
  - For RackSpace, navigate to **Install Folder > AccountStoreSupport > RackSpaceCloud > RackSpaceCloud.dll**, and then click **Open** to set the assembly path to **RackSpaceCloud.dll**.
  - For Salesforce, navigate to **Install Folder > AccountStoreSupport > ForceDotCOM > ExtSalesforce.dll**, and then click **Open** to set the assembly path to **ExtSalesforce.dll**.
  - For IBM Softlayer, navigate to **Install Folder > AccountStoreSupport > SoftLayer > ExtSoftLayer.dll**, and then click **Open** to set the assembly path to **ExtSoftLayer.dll**.
- **Management Type:** Select **Full account store**.
- **Type in Assembly (Management):** These are auto-populated as follows:
  - For Amazon: **ExtAmazonCloud.AmazonAWS**
  - For Microsoft Azure: **ExtMicrosoftCloud.Azure**
  - For RackSpace: **RackSpaceCloud.RackSpace**
  - For Salesforce: **ExtSalesforce.Salesforce**
  - For IBM Softlayer: **ExtSoftLayer.SoftLayer**



6. Leave **Advanced: Customer Extension ProgID** empty.

7. Click **Verify Settings** to ensure the extension is working properly. Typical problems at this point are generally related to bad Type in Assembly, bad ProgID (should be blank), or missing .Net 4.5.2 framework (or later from 4.x family) or missing Windows Management Framework 4 or later.

8. Click **OK** to close the **Configure Custom Account Store - CLR** dialog.

9. Click **OK** to close the **Custom Account Store Types** dialog.

## Enroll Custom Account Stores

Privileged Identity supports quite a few out of the box targets for discovery and management. However, there is not enough real estate to add everything that could possibly be managed by name. For example, you may wish to have a node for BSD and a different node for AIX and a different node for Solaris and a different node for HP-UX and so on.

Keeping in mind, many nodes host the same functionality (e.g. Linux/Unix, DRAC, AS400, OS390 or all the LDAP nodes) the question is what is required to connect to the endpoint. If the answer is SSH or Telnet or LDAP, outside of the included nodes, we can most likely connect to and manage the target.

If you then desire to have a node specific to your target system type, you can add a custom account store.

In the current iteration, Privileged Identity has many custom account stores already defined such as VMware ESX, Amazon Web Services and so on. BeyondTrust provides the extension modules to plug into these types of custom account stores. For SSH and Telnet connections, the SSH and Telnet functionality is provided by the core product. So when you have a target that uses SSH and Telnet and wish to have a custom node for it, you may simply add a node by following the steps in this section.

Adding a custom node affects the following:

A new node will be added to the tree control in the management console's account store view.

A custom namespace will be established and named after the account store you have added. We recommend keeping the name simple and limited to the following characters:

- a-z
- A-Z
- dot (.)
- Hyphen (-)

Spaces are permitted, but must be accounted for in all programmatic access by wrapping the namespace in double quotes ("").

The custom namespace will be present in search filters in the web interface automatically.

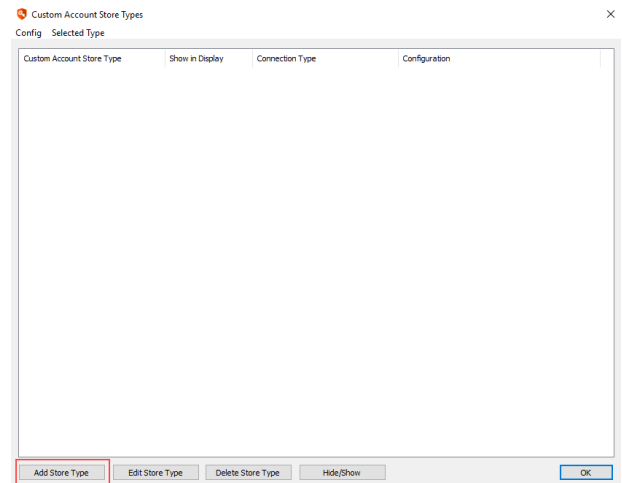


*For more information on available namespaces, please see "Namespace Values" on page 589.*

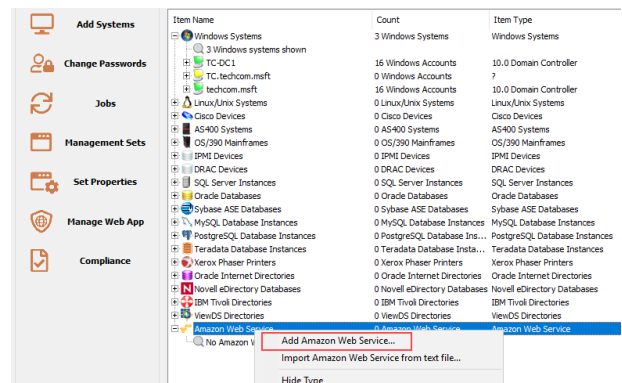
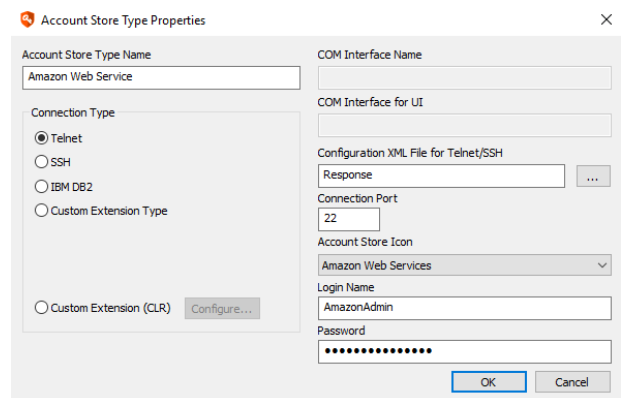
Accidentally deleting the custom account store will cause havoc as systems and devices associated with the custom account store will need to be manually re-added to the account store upon re-creation.

There are no move or copy operations on custom account stores. That means you must add or remove systems from any and all management sets you associate the custom store with.

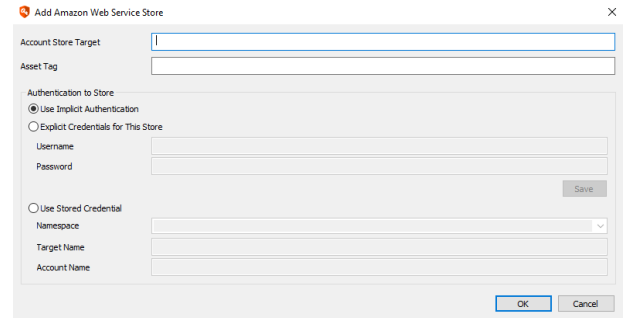
1. To add a custom account store, open the management console, and then select **Systems List > Custom Account Store Types** from the menu.
2. Click **Add Store Type**.



3. Select the connection type as either **Telnet** or **SSH**. This determines the default connection selection when performing password rotations against the target system.
4. Select the **Configuration XML File for Telnet/SSH**. This determines the default response file that will appear in the job configuration. You may change this default answer file at any time or during job creation.
5. Define the default connection port. This port is used for refresh jobs and can be overridden by the port configuration in the answer file during management operations.
6. Select the **Account Store Icon** from the list. It is not possible to define a custom icon outside of this list.
7. Define the **Login Name** and **Password** to be used as the default login when configuring a password change job. These fields may be filled in now or left blank and filled in later during job creation.
8. Click **OK**.
9. The newly added account store now appears as a node in the **Account Store View**.
10. To add an instance to the newly created account store, right-click the node, and then select **Add <account store name>**.



11. Enter the name for **Account Store Target**, and then click **Ok**. The name can be a simple name, fully qualified domain name (FQDN), or an IP address.



**Add Amazon Web Service Store**

Account Store Target

Asset Tag

Authentication to Store

Use Implicit Authentication

Explicit Credentials for This Store

Username

Password

Use Stored Credential

Namespace

Target Name

Account Name

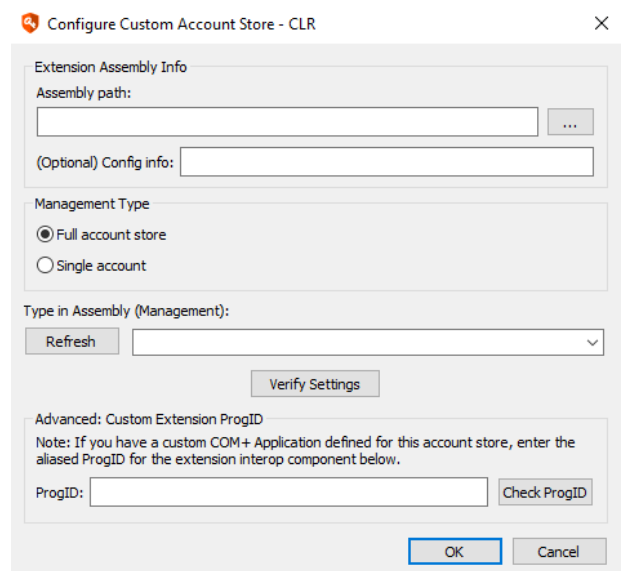
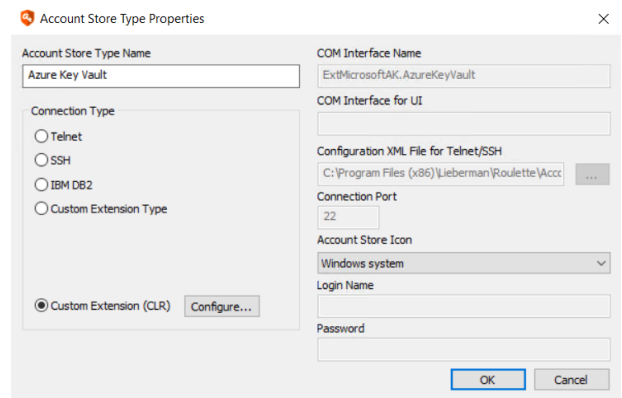
# Enroll Key Vault and Secrets Manager Providers

This section describes how to enroll the following key vault and secrets managers providers: Azure Key Vault, AWS Secrets Manager, and HashiCorp Vault Enterprise.

To enroll key vault providers with Privileged Identity, first add the key vault provider to the display tree, and then add one or more key vault directory instances. Finally, automatically populate the account store with system instances. Use the following steps.

## Add the Key Vault or Secrets Manager Provider to the Display Tree

1. Open the management console.
2. Select **Systems List > Custom Account Store Types**.
3. Click **Add Store Type**.
4. Complete the **Account Store Type Properties** dialog form:
  - **Account Store Type Name:** Enter a name for the key vault provider you are adding, for example **Azure Key Vault**.
  - **Account Store Icon:** Select the appropriate icon for the service provider from the dropdown. This specifies the account store icon to display in the display tree. You cannot add custom icons. You may choose any icon from the list. This icon affects only what is displayed in the tree control.
    - For Azure Key Vault, select **Windows system** or another option (Microsoft Azure does not have its own icon).
    - For AWS Secrets Manager, leave as **Uknown Type**.
    - For HashiCorp Vault Enterprise, leave as **Uknown Type**.
  - **Connection Type:** Select **Custom Extension (CLR)**, and then click **Configure**.
5. Complete the **Configure Custom Account Store - CLR** dialog form:
  - **Assembly path:** Click the ellipses (...) to open the **Locate Assembly** dialog.
    - For Microsoft Azure, navigate to **Install Folder > AccountStoreSupport > MicrosoftAK > ExtMicrosoftAK.dll**, and then click **Open** to set the assembly path to the **ExtMicrosoftAK.dll**.
    - For AWS Secrets Manager, navigate to **Install Folder > AccountStoreSupport > AmazonSecretManager > ExtAmazonSecretManager.dll**, and then click **Open** to set the assembly path to the **ExtAmazonSecretManager.dll**.
    - For HashiCorp Vault Enterprise, navigate to **Install**





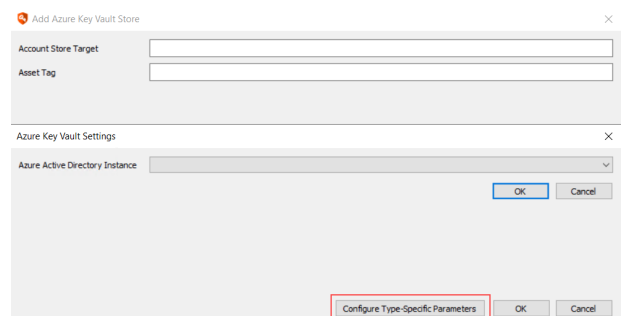
**Folder > AccountStoreSupport > HashiCorpVault> ExtHashiCorpVault.dll**, and then click **Open** to set the assembly path to **ExtHashiCorpVault.dll**.

- **Management Type:** Select **Full account store**.
  - **Type in Assembly (Management):** These are auto-populated as follows:
    - For Azure Key Vault: **ExtMicrosoftAK.AzureKeyVault**
    - For AWS Secrets Manager: **ExtAmazonSecretManager.SecretAPI**
    - For HashiCorp Vault: **ExtHashiCorpVault.VaultWebAPI**
6. Leave **Advanced: Customer Extension ProgID** empty.
  7. Click **Verify Settings** to ensure the extension is working properly. Typical problems at this point are generally related to bad Type in Assembly, bad ProgID (should be blank), or missing .Net 4.5.2 framework (or later from 4.x family) or missing Windows Management Framework 4 or later.
  8. Click **OK** to close the **Configure Custom Account Store - CLR** dialog.
  9. Click **OK** to close the **Custom Account Store Types** dialog.

## Add Directory Instances for Key Vaults and Secrets Managers

The process for adding key vault directory instances is similar to adding instances for any other account store type.

1. Open the management console, change to the desired management set or create a new one, and then select **View > Account Store View** from the menu.
2. Right-click the applicable key vault account store type (Azure Key Vault, Amazon Secrets, HashiCorp Vault), and then select **Add <vault name>**.
3. Complete the form as follows:
  - **Account Store Target:**
    - For Azure Key Vault, enter your Azure subscription name
    - For AWS Secrets Manager, enter AWS region name
    - For HashiCorp Vault, enter the https server address
  - **Asset Tag:** (Optional) If applicable, enter the asset tag for the services instance.
4. For Azure Key Vault, click **Configure Type-Specific Parameters**, select your Azure Active Directory Instance, and then click **OK**.



5. For AWS Secrets and HashiCorp Vault, complete the **Authentication to Store** section, and then click **Ok**. Use this section to specify how to authenticate to the instance.
  - **Use Implicit Authentication:** Select only if your environment supports integrated authentication with key vault provider, which is not typical.

- **Explicit Credentials for This Store:** Enter the account credentials you use to administer the key vault instance. For **Username**, enter an active **Access Key ID**, and for **Password**, enter the **Secret Access Key**, and then click **Save**.



**Note:** For HashiCorp Vault, Privileged Identity supports only tokens and username/password types of authentication.

- **Use Stored Credential:** Select to use a stored (managed) password that is valid for this instance.
  - **Namespace:** Select the namespace that has the stored password.
  - **Target Name:** Enter the name of the target system associated with the stored account.
  - **Account Name:** Enter the name of the stored account to use.



For more information on available namespaces, please see "[Namespace Values](#)" on page 589.

## Add Certificates, Keys, and Secrets to the Directory Tree

Once your Azure or HashiCorp key vault or AWS Secrets Manager has been added to the directory tree in the Accounts Store View and configured to connect successfully, you need to retrieve the certificates, keys, and secrets from the directory instances.

1. From the Accounts Store View, expand the key vault or secrets manager node.
2. Right-click the directory instance, and then select **Refresh <instance name>** to connect to the instance and retrieve its metadata.
3. Right-click the directory instance again, and then select **Refresh Accounts List <instance name>** to connect to the instance and retrieve the certificates, keys, and secrets and populate them into the directory tree.

You can now create password change jobs to change the secrets in the key vault or secrets manager.



For more information on creating the password change job, please see "[Manage Secrets in Key Vaults and Secrets Managers](#)" on page 293.

## Discover Privileged Accounts

This section covers account discovery and password propagation.

The discovery data displayed in the management console is not used for password change jobs. Rather, this data is meant to provide an idea of what is available to be changed, or what may be affected during a change job with password propagation. Every password change job that involves propagation performs its own discovery every time it runs to ensure it is working with the most up-to-date information.



**Note:** *There are some systems and devices that Privileged Identity can manage (password rotation) but cannot discover accounts on.*

## Refresh Operations

A refresh job operation gathers system information for all selected systems and stores it in the primary data store. A refresh initiates an administrative connection to the selected machine(s) and queries the system for system information. If an administrative connection cannot be made using the current logon credentials, any pre-existing credentials or alternate administrative credentials previously stored are attempted. If one or more systems fail during the refresh, and the job is a scheduled job, the failed systems will be queued again for retry according to the global retry policy.

All jobs and their settings are saved and are visible in the jobs monitor together with the job log. This makes it easy to re-run a job or examine its log file.

After a successful refresh, you can view account information for a set of systems by choosing **View > Account Store View** from the menu.

There are multiple ways to create a refresh job:

- Select the systems, and then press **F5**, or right-click, and then select the any of the 4 refresh options from the menu:
  - **Refresh System Information**
  - **Refresh Trusted Domains and Systems for this System**
  - **Refresh System and Account Information**
  - **Refresh System and Discover Local Account Usage**
- Select nothing, and then press **F5**, or right-click in the empty space below the list of systems, and then select **Refresh System(s)** from the menu. This will refresh all systems in the current managed group.



**Note:** *F5 refreshes Windows and Linux/Unix systems only.*

- Select **Systems List > Refresh Information** from the menu, and then select any of the 3 refresh options from the menu:
  - **Refresh System Information**
  - **Refresh System and Account Information**
  - **Refresh System Trust Information**



**Note:** *Computer information is also refreshed whenever a password spin job or account discovery job is run against that computer.*

## Refresh Accounts in the Management Console

Except where noted, use the **Account Store View** in the console for the steps in the following sections. Use the **SSH Key View** for SSH key discovery and management when targeting Linux and Unix and related systems.

To select a single system, click it. To select multiple systems, press the **Ctrl** key while clicking each system. To select a range of systems, click one system and then hold down the **Shift** key while selecting the last system in the range.

# Alternate Administrator Accounts

The **Alternate Administrator Accounts** feature allows additional credentials to administer systems in multiple domains and work groups or to perform system refreshes of Linux/Unix systems when system credentials do not yet exist.

Privileged Identity automatically uses the current login credentials, previously cached credentials, stored credentials, or any of the alternate administrator credentials when it performs operations. Disabling the use of alternate administrative credentials does not clear the cache of good credentials if there are already connections to target computers. To clear the cache and disable alternate credentials, first turn off alternate administrative credentials and then perform a test connection from the **Alternate Administrator Accounts** dialog.

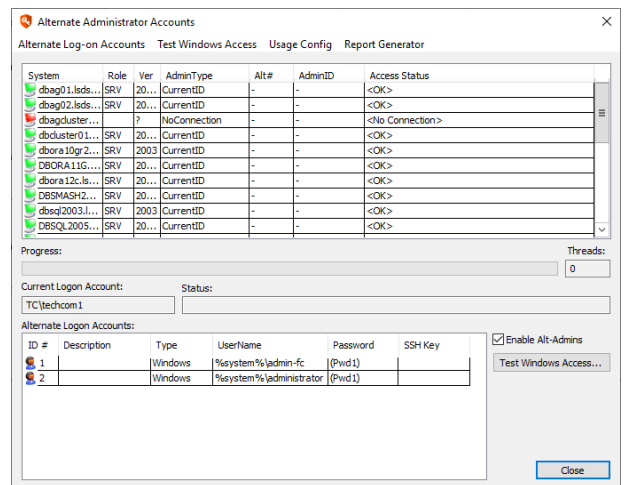
The **Alternate Administrator Accounts** feature has limitations and cannot be used in all instances. With Microsoft, an alternate administrator credential is seen as an impersonated authentication attempt. The remote **COM** and **DCOM** interfaces on Windows systems do not work with impersonated connections. As a result, any element that uses these interfaces, such as **Scheduled Tasks**, **IIS** or **DCOM**, fails to propagate or possibly is not even discovered. Simple password changes and service management however are successful as they do not use these interfaces.

When using alternate administrator credentials, it is normal to experience delays on some systems during operations because the program must wait for bad credentials to timeout before trying alternate credentials. Credentials are tried in order, so if many alternate credential entries are defined, the first in the list is first and then so on until the entire list of credentials is tried or a successful set of credentials is found.

To access this feature, go to **Settings > Alternate Administrator Accounts**.

The top list shows the list of systems in the current management set and any previous information recorded about the systems. The lower left of the dialog lists the alternate administrator accounts. The **Status** field shows the current status of any task that has begun and has not yet completed. The **Threads** field shows how many threads are actively working on the current task (**0** displays when work is complete or no operation in progress). The **Progress** bar is an approximation of task completion. The **Current Logon Account** is the account the solution is opened as. The **Enable Alt-Admins** check box is a program wide option that allows the use of alternate administrative credentials for all connections made through the tool.

Alternate administrator accounts can be added, edited or deleted using the **Alternate Log-on Accounts** menu.



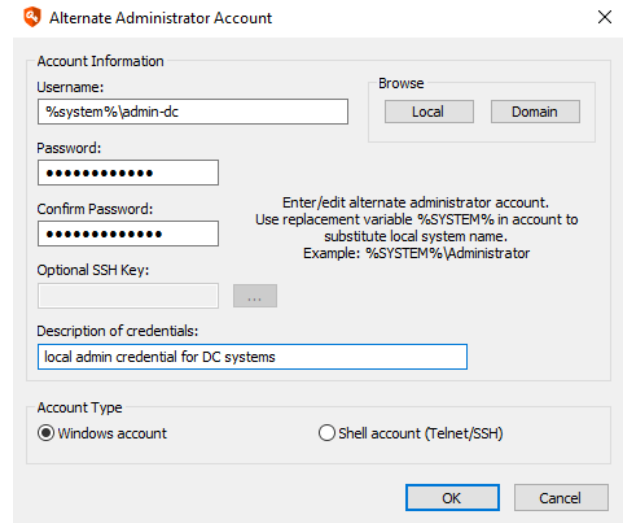
## Add, Edit, or Delete Alternate Administrator Accounts

To edit or delete one of the entries, highlight the entry, then select **Alternate Log-on Accounts**, and then select either **Edit** or **Delete**. To add a new alternate administrator, select **Add**. These options are also available through the right-click menu of the **Alternate Logon Accounts** list.

Manually enter the name of the alternate administrator (use the DomainName\UserName format or UserName formats), or for Windows systems, use the **Local** or **Domain** browse buttons. You may also use a substitution, such as **%system%**, to replace the system name for local account changes to multiple systems.

For example, the local machine name is DCTR1, is a domain controller in domain DOMAIN, and has an account named CustomUser. The target machines each have local accounts named CustomUser, but can also be accessed by the account DOMAIN\CustomUser. By specifying **%system%\CustomUser**, the local CustomUser account on each machine is specified, rather than the domain account DOMAIN\CustomUser account on each machine.

If configuring an alternate account for a non-Windows platform, such as Linux, Unix, or OS X, change the account type to **Shell Account (Telnet/SSH)**.



## Test Administrator Account Access

Check the **Enable Alt-Admins** box to use all alternate credentials when accessing systems.

To test access, highlight one or more systems (if none are selected, all systems in the list are tested for access), and then click the **Test Access...** button (or go to the menu item **Test Access > Start**). This test identifies which systems are online in and which credentials worked with which systems. The testing is complete when the **Threads** counter equals **0**.

The **AdminID** and **Password** columns show which account/password provided administrator access to each remote system. If there is a number in the **ALT#** field, this corresponds to the ID# of the alternate administrator account that successfully connected. If a dash (-) is in the **ALT#** field, it means that an alternate administrator account was not used to connect to the computer. If none of the entries worked, this is reflected in the **Access Status** field. Lack of appropriate administrator credentials is shown by an error code of 5 - Access Denied. Other error codes (i.e. 53, 1722) usually indicate an offline system.

## Enable Alternate Administrators

Typically, the logon account is used for connections. To have the program try alternates in case of problems authenticating, check the **Enable Alt-Admins** box. Be aware that not every feature in the solution may work through alternate administrators as there may be limitations on impersonation imposed by Microsoft.

## Report Generator

Export the results of an authentication test using the built in **"Report Generator"** on page 533.

# Configure Account Discovery and Password Propagation

Propagation configuration settings define how Privileged Identity searches for credential references, and updates those references when the source credential is changed. For example, if you're using an account in a configuration file to connect to a database as part of an application running, when Privileged Identity updates the account password, it updates the password within that configuration file automatically as part of the password update.

To configure what is enumerated when you select the **Refresh System and Discover Local Account Usage** option, as well as what is included in a password change job by default, select **Settings > Discovery and Propagation Defaults** from the top menu.

The top pane identifies what items are configured by default (with default configurations that cannot be edited). These items can be enabled or disabled by selecting them and clicking either **Enable** or **Disable**. This toggles the **Enabled** column to True (Enabled) or False (disabled). If an item is disabled, it is removed from the tree view under the target system-account.

The bottom pane identifies user-defined discovery and propagation elements. These items can be edited for scope, related settings, and management set scope by clicking the **Edit** button.

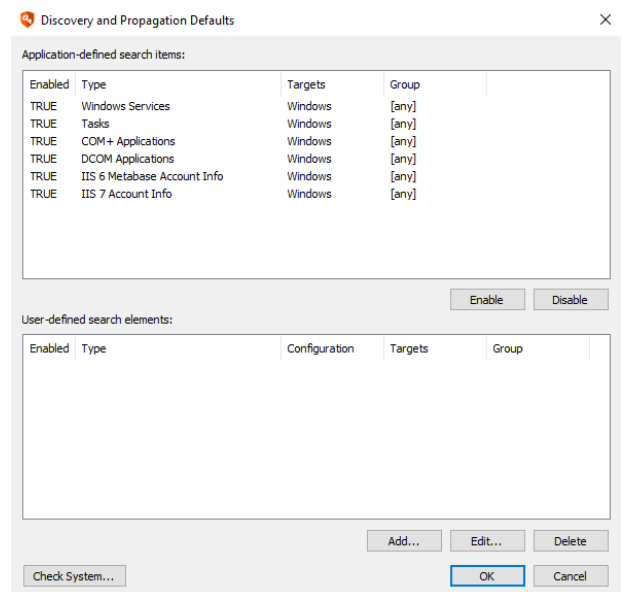
Items are processed in the order presented, from the top of the dialog to the bottom.

Configuration of propagation types affects what ports are required during discovery or propagation of a particular target. For example, Windows Services propagation occurs over port 445 (or 137-139) which is the same basic communication ports required for basic local password changes while Windows COM Applications use port 445 (system refresh), and port 135 (RPC port mapper) and ephemeral ports (varies by target OS).

Click **Check System** to specify a system name (as currently entered in the solution) to see what is propagated against the target system by default.

Possible target propagations include:

- **Windows Services:** Affects the identity used to logon for Windows services. Default has no configurations. Windows Services attempts to identify if each Windows service is clustered; if so, various cluster APIs are leveraged. If the service is not clustered, the service control manager (SCM) APIs are leveraged. Each Windows service is stopped and evaluated for dependencies. If any dependent services are found, they are also stopped. Each Windows service and any dependant service is then restarted in the correct order. User-added propagation allows configuration of service auto-restart functionality.
- **Windows Scheduler Task RunAs Identities:** Affects credentials used to run Windows Scheduled tasks. There are no configuration options for this propagation type. Use **Settings > Program Options > Performance** to enable a timeout case. The timeout case uses a performance check to determine how long it takes to enumerate the tasks on a target system and if that timeout is exceeded. If the timeout is exceeded, this operation is skipped.
- **Windows Scheduler AT Service Account:** Affects the identity used for AT tasks (deprecated after Windows Server 2012). There are no configuration options for this propagation type.
- **COM Application Identities:** Affects COM Application Identities. There are no configuration options for this propagation type.
- **DCOM Object RunAs identities:** Affects DCOM application RunAs identities. There are no configuration options for this propagation type. Use **Settings > Program Options > Performance** to enable a timeout case. The timeout case uses a performance check to determine how long it takes to enumerate the DCOM applications on a target system and if that timeout is exceeded. If the timeout would be exceeded, this operation is skipped.





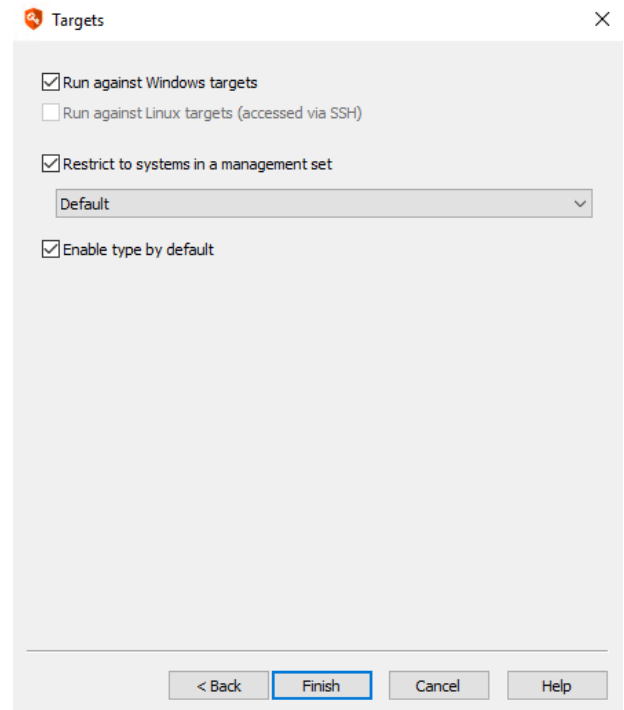
- **IIS6 Metabase Account Information:** For IIS6 (Server 2003), affects users configured to run application pools, configured as the anonymous account for a web site or virtual directory, and the account configured for network access. There are no configuration options for this propagation type. If a target system is detected as Windows Server 2008 or later, this operation is skipped.
- **IIS7 Account Info:** For IIS7 and later (Server 2008 and later), affects users configured to run application pools, configured as the anonymous account for a web site or virtual directory, and the account configured for network access. There are no configuration options for this propagation type. If a target system is detected as Windows Server 2003 or earlier, this operation is skipped.
- **SCOM RunAs Accounts:** Affects credentials configured as RunAs identities within Microsoft System Center Operations Manager (SCOM). Use of this propagation requires copying the correct SCOM SDK binary files (all files from the SDK binaries directory of the SCOM host to the installation directory of the management console and/or zone processor. There are no configuration options for this propagation type.
- **Credentials in SQL Server:** Lists accounts under the Credentials node as seen in SQL Management Studio. The target SQL database instances must be defined.
- **Accounts in .NET Config Files:** List accounts configured in the Connection Strings component of ASP.NET under IIS 7 or later. This searches for the following elements: User, UserID and UID. There are no configuration options for this propagation type.
- **String Replacements in Files:** Configure the path to a file that is parsed for password replacement using the proper RegEx expression.
- **Run Arbitrary Process:** Configure the path to a program or script to run (on the host system or target system), configuration of that program runs on the host or target system and who the program runs as and if any files need to be copied to the target system (Windows only).
- **SharePoint:** There are no configuration options for this propagation type. Use **Settings | Program Options > General** to specify the SharePoint admin port that should be used for management. If you are attempting to manage multiple SharePoint farms, they all need to be on the same port. This propagation element deploys a temporary service to the target machine to perform the propagation which self-terminates and removes itself.
- **IBM WebSphere Application Server:** Configure the use of SSL and the target port. This propagation type is typically not used as the functionality has been supplanted by directly managing the WebSphere server using the IBM WebSphere node.
- **Oracle WebLogic Server:** Configure the use of SSL and the target port. This propagation type is typically not used as the functionality has been supplanted by directly managing the WebLogic server using the Oracle WebLogic node.
- **SAP Server:** Configure the use of the Netweaver gateway or direct connect and related information. This propagation type is typically not used as the functionality has been supplanted by directly managing the SAP system using the SAP node.
- **Aggregation of multiple base types:** Allows adding of multiple propagation steps in a user defined order.
- **Update Logon Cache:** Affects identities stored in the Windows Logon Cache (cached logons). There are no configuration options for this propagation type.
- **Update Auto Logon Account:** Updates the registry setting for the configured auto-logon account on a Windows system. If the system is not configured to use auto-logon or if the configured auto-logon account is not the same as the account being updated, the propagation will have no effect.
- **Local Cache for Java Client:** Affects the credentials stored in the local Java Cache client (part of the legacy SDK provided with this product) if the local java cache is deployed and running to target systems. There are no configuration options for this propagation type. This specific propagation uses an RMI connection.
- **SQL Reporting Services:** Affects the account account for the specific SQL Reporting Services instance (SSRS). There are no configuration options for this propagation type.

All propagation types can be given a custom label, limited to Windows or Linux systems, and can be limited to a particular management set.

The following options may be set on the **Targets** tab for all propagation types:



- **Run against Windows targets:** Enables this propagation type to be used when propagating to Windows targets.
- **Run against Linux targets (accessed via SSH):** Enables this propagation type to be used when propagating to systems under the Linux/Unix node when SSH (not Telnet) is used as the connection protocol.
- **Restrict to systems in a management set:** Restricts this propagation type to limit to a specific list of machines as found in a particular management set. Consider a service account used for windows services, tasks, in shell scripts, and also for SharePoint. By placing the SharePoint server in a management set designated for the SharePoint servers, as well as in the target management set for propagation, and restricting this propagation to the SharePoint servers management set, when the propagation runs against all of those systems, the SharePoint propagation is only attempted against the SharePoint servers and not on any other system. If the restriction is not set, the SharePoint propagation would be attempted against every system, including Linux machines, even if they don't actually run SharePoint. This can cause a job that could run in a few minutes to take considerably longer and could lead to other problems such as misleading failure notifications.
- **Enable type by default:** When enabled, this propagation is included for discovery jobs and is added to password change jobs, though it can be removed at job creation time for password change jobs.



**i** You can also configure propagation types from the web application. For more information on configuring propagation types using the web application, please see: "[Configure Propagation Types in the Web Application](#)" on page 512.

# Discover Linux, Unix, OSX and Solaris Privileged Accounts & Keys

Privileged Identity supports discovery of accounts on most flavors of Linux, OS X, Unix, and Solaris. SSH is used to connect to each system and enumerate the contents of the `/etc/shadow` file or the `/etc/passwd` file. The `shadow` file is attempted first and, if access to the file fails because the file does not exist or the login account does not have permissions, the `passwd` file is then attempted. If that fails, account discovery for that system fails.

Failure of account discovery does not mean that password change jobs will be unsuccessful, it just means that account list enumeration has failed.

## Account Discovery Considerations

Privileged Identity will scan and attempt to identify the user accounts local to the target system. However, there are many system accounts already managed by the system, and as such, may not be useful to display to you, the administrator. To aid in this account discovery process Privileged Identity provides account scan rules.

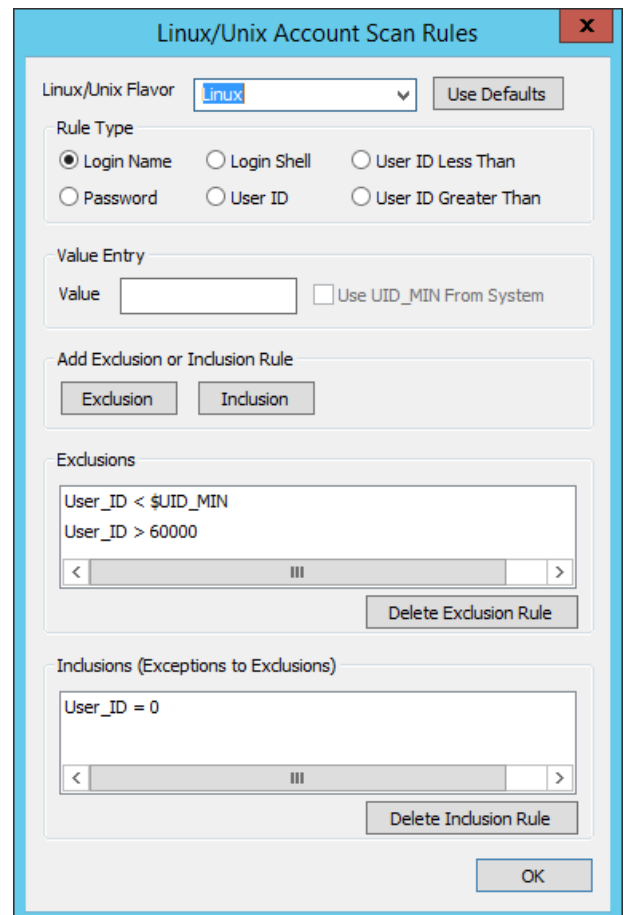
To open the account scan rules, right-click on the Linux/Unix Systems node and select **Configure Linux/Unix Account Scan Rules**.

The goal of the account scan rules is to provide exclusion and inclusion conditions when scanning the `passwd` file. These conditions are based on fields in the `passwd` entry. The Login Name, Password, Login Shell and User ID fields are all equality conditions. The last two, also based on User ID are Less Than and Greater Than conditions. The User ID rules can specify a number or the value `UID_MIN` which is contained in the `/etc/login.defs` file on Linux only. The `UID_MIN` value is only used on Linux. An exclusion rule specifies which `passwd` file entries to exclude. The inclusion rules are exceptions to the exclusions. In the case of the Linux rules shown above, all User IDs less than `$UID_MIN` are excluded as they are system accounts which, with few exceptions, cannot be logged into. But the exception to this exclusion is the User ID 0, which is the root account (and `toor` on BSD systems) which are accounts that are logged into and are the administrator accounts.

The pull down menu at the top allows you to select which flavor the rules apply to. The Rule Type group offers which of the password fields the rule applies to. The Value Entry group provides a value entry to compare against the `passwd` field selected, or the `UID_MIN` value on Linux in the case of the 3 User ID rules. The Exclusion or Inclusion Rule group is where you select whether this rule is an exclusion rule or an inclusion rule. Once clicked, the Value field is blanked and the rule is placed in the appropriate text box below. The Value field can be a regular expression for comparing the 3 strings fields. The regular expression syntax used is the [extended POSIX regular expression grammar](#).

To remove a rule, select one or more rules in either box (multi-select from both boxes will not remove all selected) and click either Delete Exclusion Rule or Delete Inclusion Rule buttons.

If after modifying the rules for a flavor, you decide that the accounts being shown are not correct, or you want to return to the original settings, click the **Use Defaults** button to restore the original settings. A dialog will appear to tell you that all the settings will be removed and replaced by the defaults and you must verify that is what you intend.



After scanning a Linux/Unix system with scan rules in place, the accounts shown in the Account Store View are a small subset of what appeared before.

## Refresh Operations

For authentication to a Linux/Unix system during refresh operations, the following logic is evaluated in this order for account enumeration:

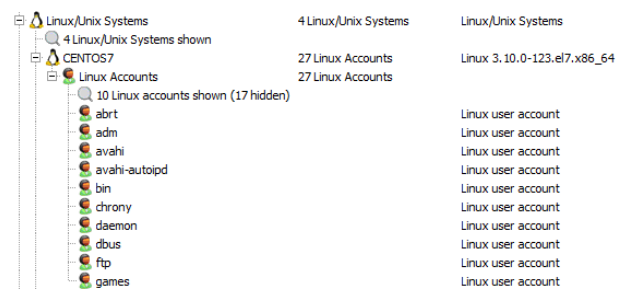
1. SSH key authentication
  - a. If an SSH key is configured and mapped as root, attempt to read files directly.
  - b. If an SSH key is mapped to "not root", attempt to use sudo to read files.
2. No SSH key authentication
  - a. If root is available, attempt root authentication to read files directly. If root is available but fails, look for "not root" account and attempt sudo to read files.
  - b. If root is not available, attempt sudo to read files.
3. If all options fail, then fail the lookup.

For the sudo operations there are two additional considerations:

- The login account must not be configured for "requiretty" in the sudoers file. This is most quickly achieved by commenting out **Defaults requiretty** in the sudoers file on the host or by adding a line such as this to the sudoers file: **Defaults:myuser !requiretty**
- The login account must be set to **NOPASSWD:** in the sudoers file.

If no credentials have been managed/stored for the target system and you wish to test connectivity and refresh, you will need to add an alternate administrator account for the system using the Alternate Administrators configuration dialog under the Settings menu of the management console. After you add the alternate administrator or manage/store a password for the system and add the systems to the system set, right-click the system(s) and select either:

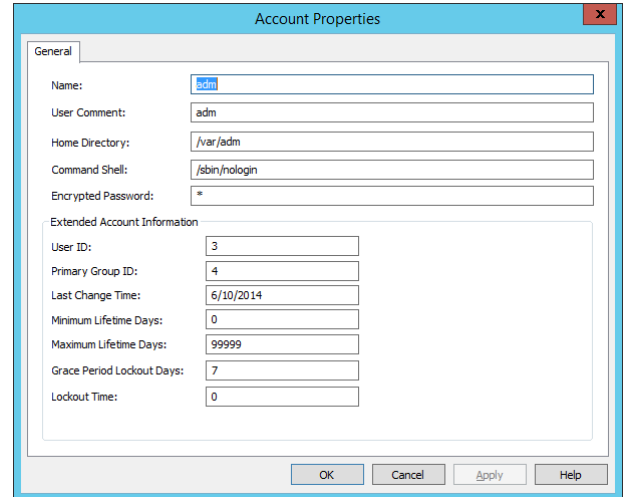
- **Refresh System and Accounts Information:** Discovers only the local accounts.
- **Refresh System and Discover Local Account Usage:** Discovers local accounts and checks for any defined (and enabled) propagation targets. See below for more info.



## Account and System Details

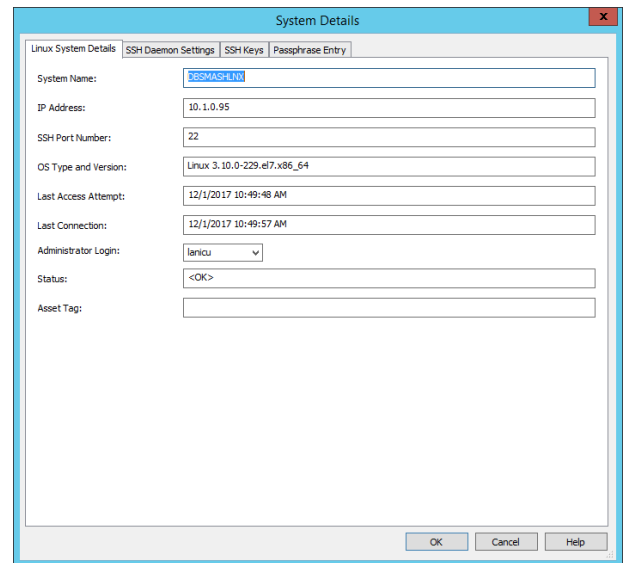
To view read-only information about an account in Linux/Unix/OS X, right-click the account and select **Account Properties**.

Right-click on the system and select System Details to see more information about the system.



While much information on the system details page is read-only, some information can be changed:

- **SSH Port Number:** when a system is enrolled, it will default to the global default SSH port number defined in program options. This value can be changed on a per-system basis. The port defined here will be used for all password change jobs and refresh operations unless overridden by a specific answer file.
- **Administrator Login:** this indicates the system specific account used to perform refresh and discovery operations against the host. The value will change to the user last successfully used to connect to the system. If desired, you may choose from other stored accounts associated with the system. However, if that account should fail to connect to the target system, Privileged Identity will begin the enumeration process to find a stored account that can be used to connect successfully.
- **Asset Tag:** the asset tag for the system. Set a value if required. This value will be visible in the web application.



Previous to version 5.5.3.0, any non-Windows system that was added as a "Linux/Unix System" was identified as Linux and its accounts as Linux Accounts. They are now properly identified by the correct operating system name. The supported flavors are

- Linux
- MacOS
- Solaris
- AIX
- BSD
- HP-UX

BSD systems that are identified as "BSD" are FreeBSD, OpenBSD and NetBSD.

Privileged Identity can also enumerate other items from Linux/Unix hosts including:

- SSH Keys
- SSH Access Rules
- SSHD Configurations
- Sudoers Configurations

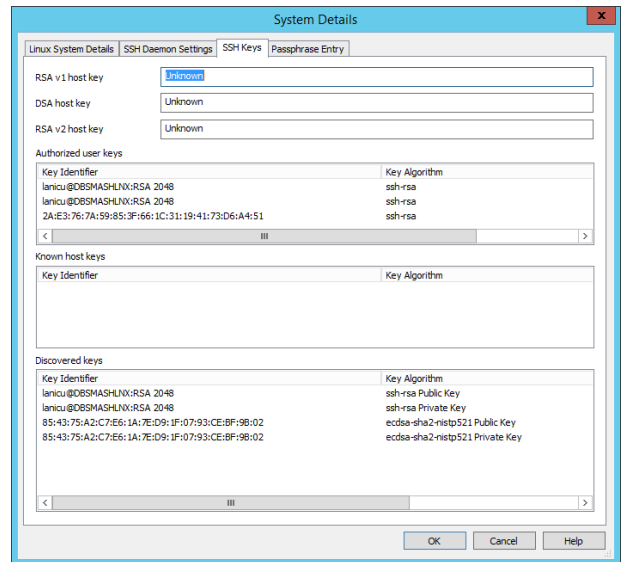
See the subsequent for more information.

## SSH Key Discovery

SSH key discovery is only valid for Linux and Unix variants added to the Linux/Unix node. Multiple key types and their properties can be discovered. As of version 5.5.2.1, the product is no longer limited to only RSA, DSA and ECSD keys nor is the product limited to key discovery based on the root profile. Rather, all key types can be successfully enumerated and all profiles (as enumerated from /etc/shadow) will be parsed for keys.

### To Discover SSH Keys

1. From the management console, open the Account Store View and select the desired systems in the list.
2. Right-click and select **Refresh system and account information** or **Refresh system and discover local account usage**. The various subsystems will be scanned to retrieve the list of accounts, SSH keys, SSHD settings, Access settings, and sudoers settings.
3. To view the keys on a specific system, right-click on the systems and select system properties and select the SSH Keys tab.



To help discover SSH keys with pass phrases, you can define a list of passphrases to use for the system during discovery using the **Passphrase Entry** tab in System Details.

Once the Passphrase Entry dialog is opened, there are two mechanisms whereby the necessary information can be entered.

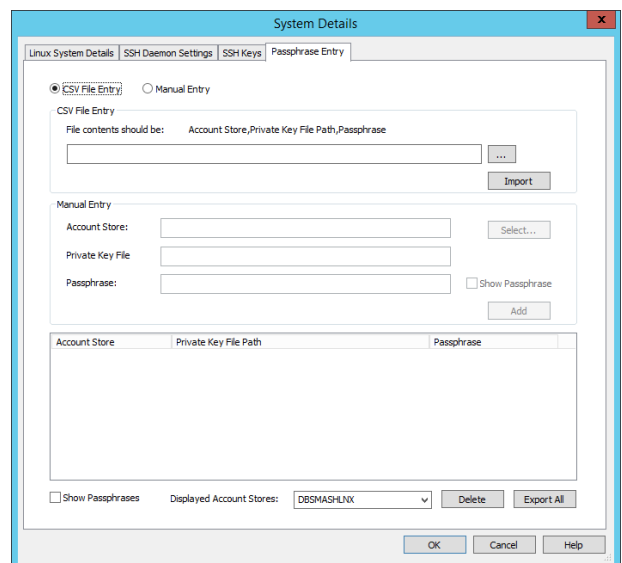
- One-by-one manual entry, in those cases where there are not a lot of passphrase protected keys
- By importing a comma-separated-values (CSV) file containing the required information.

The CSV file method requires the same information as the manual entry with the 3 required items on each line separated by commas.

1. Account Store
2. Private Key File Path
3. Passphrase

In the case of Manual Entry, the passphrase will be obscured unless the Show Passphrase box is checked. Un-checking the box will obscure the passphrase again. When the Add button is pressed, the information will be secured into the database and shown in the list box.

When CSV File Entry is used, the records are read from the file and placed in the database and shown in the list box.



All lines in the list box will have the passphrase column obscured unless the **Show Passphrases** box is checked. This check box is under the list box and is not connected to the Show Passphrase check box in the Manual Entry section. Clicking this check box for the first time will bring up a dialog asking for a password to view the passphrases. The password is the same as when attempting to open the Stored Passwords dialog. Once entered, it will not need to be entered again as long as the dialog is active.

Clicking on the Export All button will also query the user for the password as the exported file will contain the passphrases in clear-text. Obviously, it is very important that the user use this option with great caution as it could make the passphrases for many SSH Keys readily available to the bad guys. If the password is entered correctly, the user will be warned that the passphrases will be written to a file in clear text and they will be asked if they are sure they want to do this.

The **Displayed Account Stores** combo box changes the list box contents to display only the passphrase protected keys on the account stores specified. By default, it will be the account store which was right-clicked on to arrive at the context menu for whichever view was displayed on the console. The options in this combo box are:

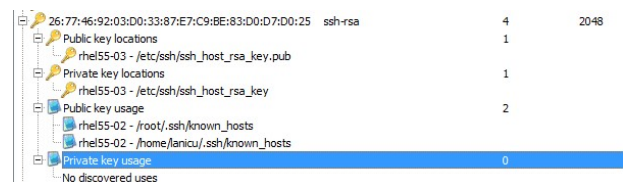
- Name of the Account Store
- Current Management Set
- All Account Stores.

The button named Delete will delete the selected passphrase specifications in the list box. Verification will not be requested when you press the Delete button. The selected entries will be deleted and a message will be displayed verifying the deletion action.

## View SSH Keys

View all discovered keys by choosing SSH Key View (click **View | SSH Key View**).

To get the list of directories to scan for SSH keys, connection made via SFTP to examine `/etc/passwd` to get the user's home directory. Any values that are subsequently overwritten by additional configuration files, such as `sssd.conf`, are not taken into account. For system keys, the `/etc/ssh` directory is examined.



Path	Type	Count	Usage
26:77:46:92:03:D0:33:87:E7:C9:BE:83:D0:D7:D0:25	ssh-rsa	4	2048
Public key locations		1	
rhel55-03 - /etc/ssh/ssh_host_rsa_key.pub		1	
Private key locations		1	
rhel55-03 - /etc/ssh/ssh_host_rsa_key		2	
Public key usage		2	
rhel55-02 - /root/.ssh/known_hosts			
rhel55-02 - /home/lanicu/.ssh/known_hosts			
Private key usage		0	
No discovered uses			



### IMPORTANT!

*If the logon user cannot parse the `/etc/passwd` and `/etc/ssh` directories, or if SFTP is not accessible, SSH key discovery and management will not work.*

Discovered keys may not be imported for a number of reasons, including when the key cannot be decoded (possibly due to password protection) or because the key is not OpenSSH compatible.

If no keys are shown when in the SSH Key view, right-click in the white space and select **Show Filter Options**. Change the filter options to an appropriate value.

Most keys stored in Privileged Identity will be those found by a system scan. It is possible, however, to Import, Generate, Extract and Restore Keys and when that happens, the path to the private and public keys will be different in the SSH Key View.

- **Imported Keys** - When the private OpenSSH key file is locally accessible from where the management console is running, it can be imported into from **Settings | Manage User Keys**. Unlike the previous version, only the private key is imported. The public key is extracted from the private key. When viewed in SSH Key View, the private and public key file paths will be in the format `RED-IM_Imported_Public_Key_<DATE> <TIME>.<MS>`.

- **Generated Keys** - Like imported keys, generated keys do not have paths names of their key files. When a generated key is viewed in SSH Key View, its name will be in the format RED-IM\_Generated\_Public\_Key\_<DATE> <TIME>.<MS>.
- **Extracted Keys** - When a scan of a Linux system encounters a private key file without a matching public key file, Privileged Identity will use the private key to extract the public key. When this is shown in SSH Key View, the public key portion will have a name in the format RED-IM\_Extracted\_Public\_Key\_<DATE> <TIME>.<MS>.
- **Restored Keys** - When a deleted and archived key is restored to the current keys, its path changes to signify that the key has been restored from the archive and placed back with the current keys. When this is shown in SSH Key View, the key paths will have a name in the format RED-IM\_Restored\_Public\_Key\_<DATE> <TIME>.<MS>.



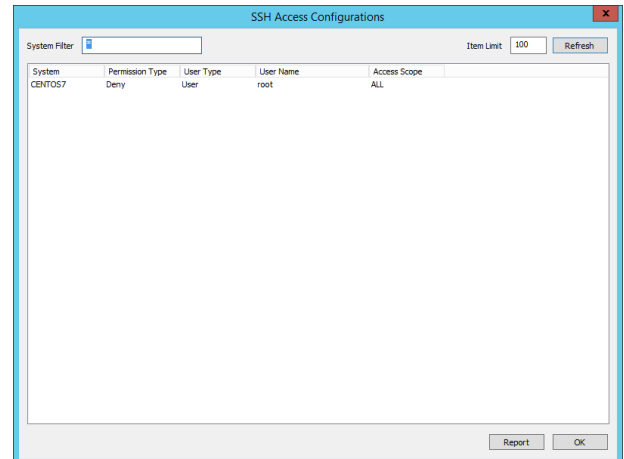
## SSH Access Rules Discovery

Access configuration discovery is only valid for Linux and Unix variants added to the Linux/Unix node. Privileged Identity examines the **access.conf** file on target systems. That file is expected to be located at **/etc/security/access.conf**.

### To View Access Settings on Linux and Unix Systems

1. From the management console, open the Account Store View and select the desired systems in the list.
2. Right-click and select either **Refresh System and Account Information** or **Refresh System and Discover Local Account Usage**. The various subsystems will be scanned to retrieve the list of accounts, SSH keys, SSHD settings, Access settings, and sudoers settings.
3. To view the information, right-click the Linux/Unix systems node in the Account Store view and choose **Show Access Rules**. If a system has no active settings, then nothing will be displayed for that system.

To generate reports on the information in this dialog, use the **Report** button in the lower right-corner of the dialog.



# SSHD Configuration Settings Discovery

SSHD configuration discovery is only valid for Linux and Unix variants added to the Linux/Unix node.

## To View SSH Daemon Settings on Linux and Unix Systems

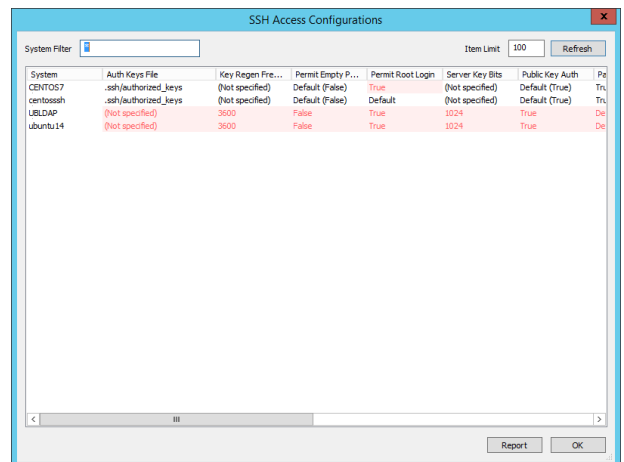
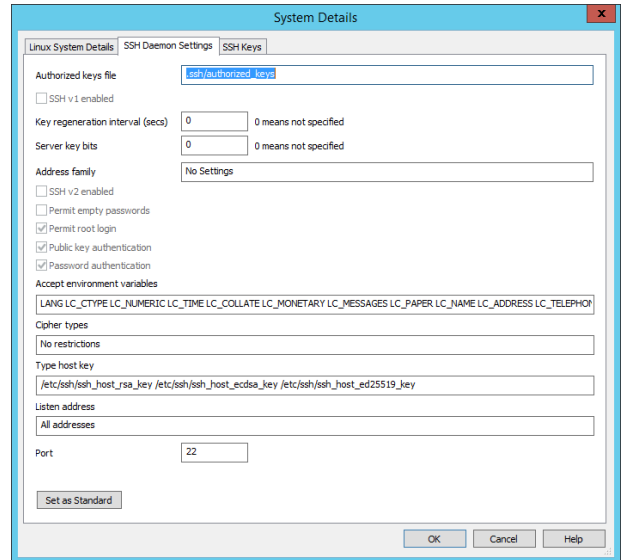
1. From the management console, open the Account Store View and select the desired systems in the list.
2. Right-click and select either **Refresh System and Account Information** or **Refresh System and Discover Local Account Usage**. The various subsystems will be scanned to retrieve the list of accounts, SSH keys, SSHD settings, Access settings, and sudoers settings.
3. To view the information, right-click the Linux/Unix systems node in the Account Store view and choose **System Details** and click the **SSH Daemon Settings** tab.

The **SSH Daemon Settings** tab has information about the system's SSH daemon (`/etc/ssh/sshd_config`) and certain configuration parameters for the ssh daemon on that system.

Reports can be generated from this dialog using the **Report** button in the lower right corner.

Note the **Set as Standard** button in the lower left corner of the page. This button is used to help generate a report of all SSH daemon configurations and compare them. To generate a comparison report of all SSH daemon settings for all Linux/Unix machines in the management set, right-click the Linux/Unix node and select **Show SSH Daemon Configurations**.

The SSH Access Configurations dialog highlights all settings that do not match the configuration that was set as the standard. Items with parenthesis around them are not explicitly TRUE or FALSE, but rather implied as being TRUE or FALSE. Also be aware that there are other settings on a Linux/Unix host that could possibly override these settings. This is simply meant to be a report of what is in the `sshd_config` files.



## Sudoers Configuration Settings Discovery

Sudoers configuration discovery is only valid for Linux and Unix variants added to the Linux/Unix node. Privileged Identity examines the sudoers files on target systems. That file is expected to be located at `/etc/sudoers`.

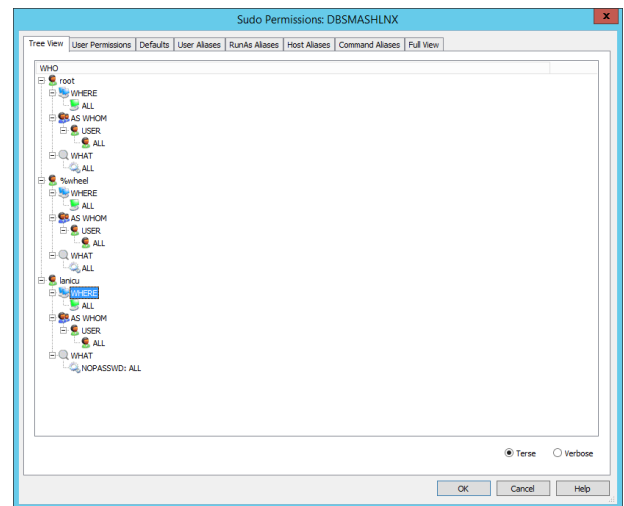
### To View Sudoers Settings on Linux and Unix Systems

1. From the management console, open the Account Store View and select the desired systems in the list.
2. Right-click and select either **Refresh System and Account Information** or **Refresh System and Discover Local Account Usage**. The various subsystems will be scanned to retrieve the list of accounts, SSH keys, SSHD settings, Access settings, and sudoers settings.
3. To view the information, right-click a system and select **Sudo Permissions**. Note, if multiple systems are selected, all tabs (except for Full View) will display the settings for the first system highlighted. The Full view tab will show all settings across all systems regardless of management set.

### Tree View

The tree view is used to simply identify who can do what as whom and where can they do it? Expand each user to see what commands the user can run, as whom, and where.

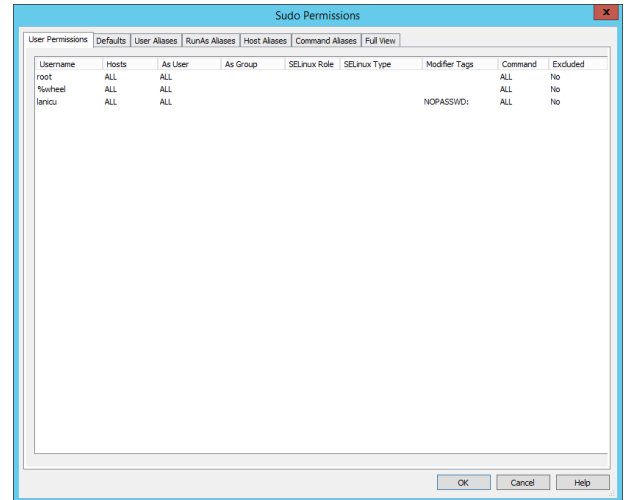
At the bottom of the dialog are the Terse and Verbose radio buttons. This will control the amount of information that is presented in the expanded tree view. Simple users, such as those displayed in the following screen shot will not offer more information. However complex users who have defined host and command aliases. By selecting the Verbose radio button, the expanded view will change to reflect the rule as it was originally written, with the alias names in place. This will give a better visual representation of the original text of the sudoers files. Commands the user is not allowed to run (prefixed with a !) will be displayed in red text in this dialog.



## User Permissions

The User Permissions Tab lists all explicit entries for user/group access in the sudoers file.

- **Username** - Identifies either a username, alias, or a group when prefixed with a % symbol that may run a command.
- **Hosts** - Which hosts (or ALL) the user can perform the command on.
- **As User** - The list of users (or ALL) that the command rule can be run as.
- **As Group** - The list of groups (or ALL) that the command rule can be run as.
- **SELinux Role** - If SELinux is used on the target, these roles further define what a user, who might otherwise be granted root access via sudo, may actually do when using sudo.
- **SELinux Type** - All files and processes are labeled with a type: types define a domain for processes and a type for files.
- **Modifier Tags** - Lists sudo command modifiers, such as NOPASSWD.
- **Command** - The list of commands (or ALL) that will be run as root or as (Run As).
- **Excluded** - Values are Yes or No. Identifies if this sudo rule is an inclusion rule or exclusion rule. For example, a rule exists that would allow the root user to run all commands on all hosts (Exclude = No), but a second rule could then exist which might limit their sudo abilities on a particular host for a particular command (Exclude = Yes)



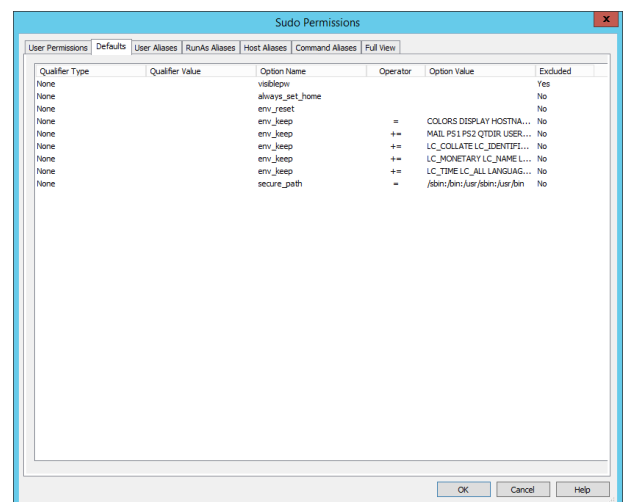
See man pages for sudoers file for more information.

## Defaults

The Defaults tab shows how the default settings are applied to this particular sudoers file.

- **Qualifier Type** - Will be none when no user is specified after a "Defaults" statement. Otherwise, qualifier would be user or group.
- **Qualifier Value** - Will list the user or group name for the setting.
- **Option Name** - The Defaults value that is being modified.
- **Operator** - Values are "+=", "-=", "EMPTY", "!" or "=". These values add or remove to or from previously defined Defaults.
- **Option Value** - The Defaults value being configured.
- **Excluded** - Values are Yes or No. Indicates if hits is an inclusion or exclusion rule.

See man pages for sudoers file for more information.

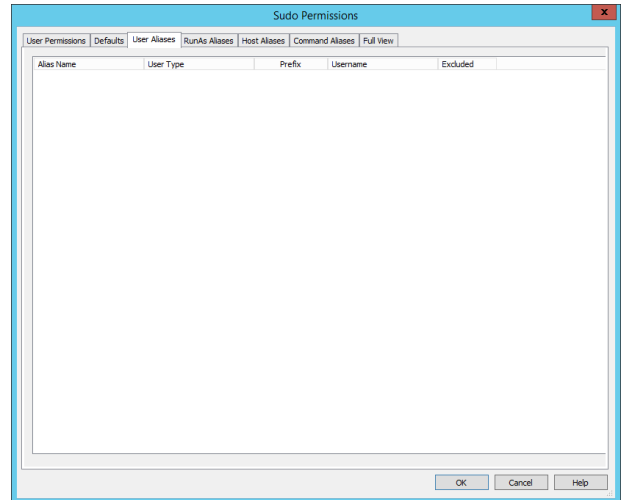


## User Aliases

The User Aliases tab shows all configured user aliases in the sudoers file. User aliases are used to specify groups of users.

- **Alias Name** - List any aliases defined for lists of users that will be included or excluded.
- **User Type** - Values are User Alias or User. Value will be User Alias when the alias consists of other aliases. Value will be User when the alias consists of user names.
- **Prefix** - When a \$ (dollar sign) appears in the prefix column, that indicates this item is an alias.
- **Username** - Lists the name of the Alias or the name of the person or group. If the Alias name is a list of usernames, this column displays those usernames.
- **Excluded** - Values are Yes or No. Indicates if hits is an inclusion or exclusion rule.

See man pages for sudoers file for more information.

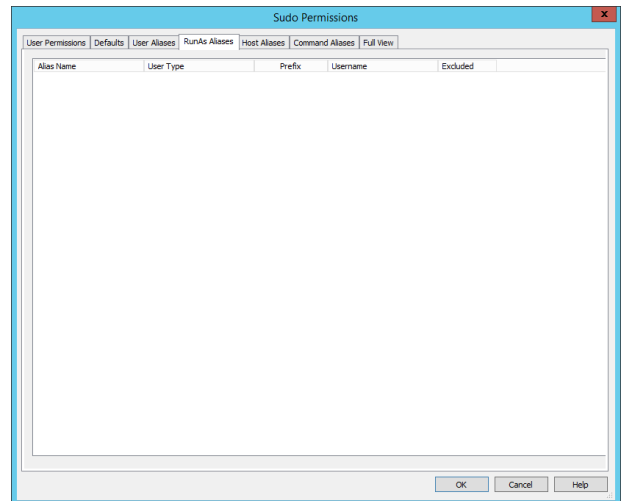


## RunAs Aliases

The **RunAs Aliases** tab shows all RunAs aliases in the sudoers file. Runas Aliases are almost the same as user aliases but you are allowed to specify users by uid's. This is helpful as usernames and groups are matched as strings so two users with the same uid but different usernames will not be matched by entering a single username but can be matched with a uid.

- **Alias Name** - Lists any aliases defined that the user may run a command as that will be included or excluded.
- **User Type** - Values are User Alias or User. Value will be User Alias when the alias consists of other aliases. Value will be User when the alias consists of user names.
- **Prefix** - Identification prefixes:
  - No prefix indicates this is a user name entry.
  - # prefix is a uid entry.
  - % prefix is a group name entry.
  - %# prefix is a gid entry.
  - %: prefix is a non-Unix group.
  - %:# is a non-Unix group id.
  - + prefix is an NIS group.
- **Username** - Lists the name of the Alias or the name of the person or group the command may be run as.
- **Excluded** - Values are Yes or No. Indicates if hits is an inclusion or exclusion rule.

See man pages for sudoers file for more information.

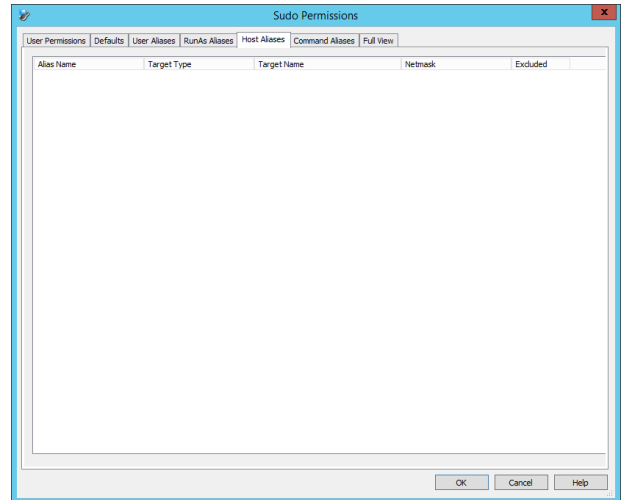


## Host Aliases

The **Host Aliases** tab shows all configured host aliases. A host alias is a list of hostname, ip addresses, networks and netgroups.

- **Alias name** - The alias name for the system, IP, network, or netgroup that will be included or excluded.
- **Target Type** - Identifies if the entry is for a system, IP, network, or netgroup.
- **Target Name** - List the system name, network, netmask, or netgroup identified in the alias.
- **Netmask** - If defined, lists the subnet mask or bits for the network ID. E.g. /24 or 255.255.255.0.
- **Excluded** - Values are Yes or No. Indicates if hits is an inclusion or exclusion rule.

See man pages for sudoers file for more information.

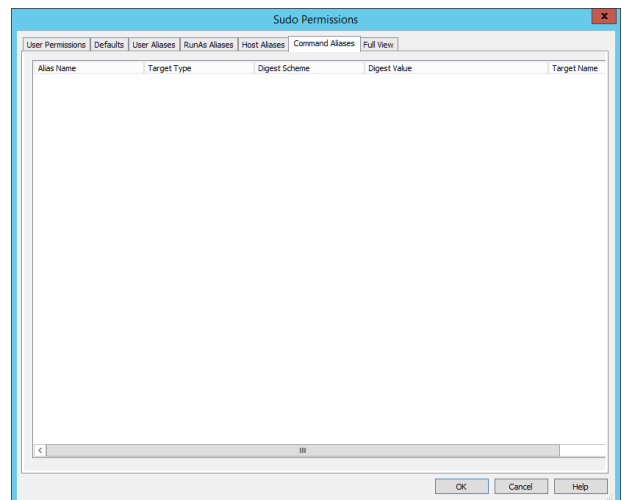


## Command Aliases

The **Command Aliases** tab shows all configured command aliases in the sudoers file. Command aliases are lists of commands and directories. These are used to specify a group of commands.

- **Alias Name** - The alias name of the list of commands that will be included or excluded.
- **Target Type** - Values are Command or Command Alias. Identifies if the entry is for a single command (Command) or command alias (Command Alias).
- **Digest Scheme** - Lists the checksum algorithm method for the command.
- **Digest Value** - The checksum of the command.
- **Target Name** - The name of the command or command alias.

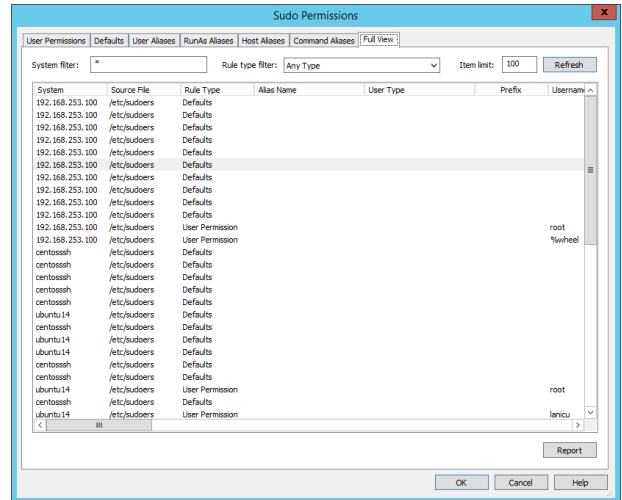
See man pages for sudoers file for more information.



## Full View

The **Full View** tab shows all sudoers settings across all systems, regardless of the management set.

To generate reports on the information in this dialog, use the **Report** button in the lower right-corner of the dialog.



# Discover Windows Privileged Accounts

Privileged Identity supports discovery of accounts on all flavors of Windows from Windows NT4 through Window Server 2016 (Windows networking permitting. See Microsoft documentation for more information on cross-version communication and management). This process is facilitated as either an LDAP or Net API call depending if the target system is a domain controller or not.

Account enumeration or discovery is not a required step for password change jobs, with or without propagation, to work. This is because the jobs target a specific account name and re-enumerate account usage against all target systems every time the job runs. This ensures that when jobs do include propagation that the most up to date information is used rather than outdated cached information.

To perform an account discovery once the system(s) are added to the system set, simply right-click on the system(s) and select either:

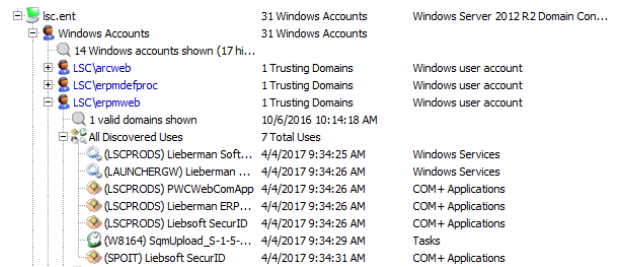
- **Refresh System and Accounts Information:** Discovers only the local accounts
- **Refresh System and Discover Local Account Usage:** Discovers local accounts and checks for any defined (and enabled) propagation targets. See below for more information.

Domain controllers have an additional option (ignored if not a domain controller):

- **Refresh trusted domains and systems for this System:** Discovers all trusted domains and systems in those domains. This information is then used to help show all places an account could be valid on, even if they are not being managed.

The accounts store view displays all items from the context of the system hosting the account or subsystems. This means to see a domain account and its usage, expand the domain controller, locate the account, expand the account, and see all usage across all discovered systems. Alternatively, expand each system and respective subsystems to see which local or domain accounts are in use on that system.

Privileged Identity supports a variety of discovery of account usage discovery for Windows systems.



Account Name	Usage	System
31 Windows Accounts	31 Windows Accounts	Windows Server 2012 R2 Domain Con...
14 Windows accounts shown (17 hi...		
LSC\arcweb	1 Trusting Domains	Windows user account
LSC\erpmdefproc	1 Trusting Domains	Windows user account
LSC\erpmweb	1 Trusting Domains	Windows user account
1 valid domains shown	10/16/2016 10:14:18 AM	
All Discovered Uses	7 Total Uses	
(LSCPRODS) Lieberman Soft...	4/4/2017 9:34:25 AM	Windows Services
(LAUNCHERGW) Lieberman ...	4/4/2017 9:34:26 AM	Windows Services
(LSCPRODS) PWCWebComApp	4/4/2017 9:34:26 AM	COM+ Applications
(LSCPRODS) Lieberman ERP...	4/4/2017 9:34:26 AM	COM+ Applications
(LSCPRODS) Liebssoft SecurID	4/4/2017 9:34:26 AM	COM+ Applications
(WB164) SqmUpload_S-1-5...	4/4/2017 9:34:29 AM	Tasks
(SPOIT) Liebssoft SecurID	4/4/2017 9:34:31 AM	COM+ Applications



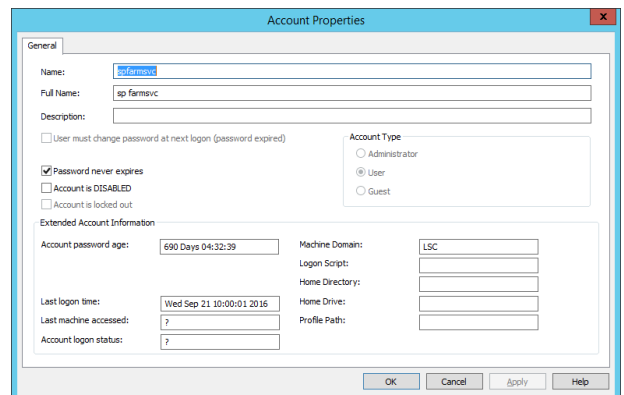
For more information about configuring discovery and propagation defaults, please see ["Configure Account Discovery and Password Propagation"](#) on page 179.

To view information about an account within Windows, right-click on the account and select **Windows Account Properties**.

The following user elements can be edited from within the Windows Account Properties dialog:

- User cannot change password
- Password never expires
- Account is disabled
- Full name
- Description

In addition, on the context menu of a Windows users account, options are available to enable, disable, or delete the user account.

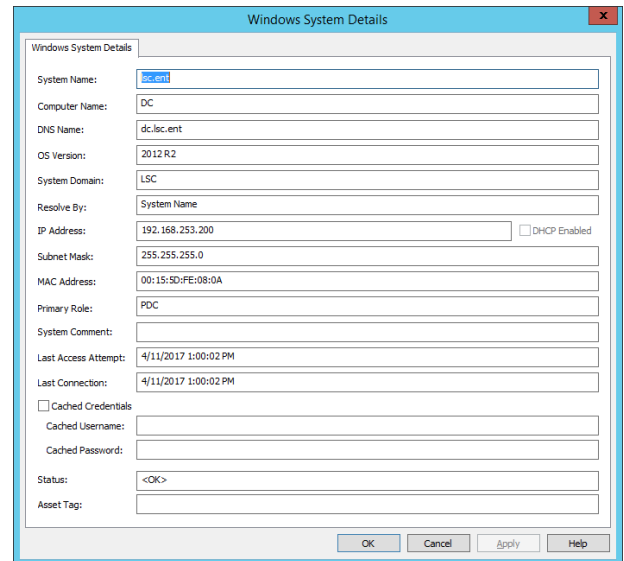




To view information about the system itself, right-click on the system and select **System Details**.

From the system properties it is possible to view or change the cached credentials that are used to connect to the system. Cached credentials will be automatically configured when alternate administrator accounts or other well-known accounts (rather than integrated authentication) are used to manage the system.

It is also possible to change the asset tag of the system from the System Details view.



Windows System Details

System Name:	dc.lsc
Computer Name:	DC
DNS Name:	dc.lsc.ent
OS Version:	2012 R2
System Domain:	LSC
Resolve By:	System Name
IP Address:	192.168.253.200 <input type="checkbox"/> DHCP Enabled
Subnet Mask:	255.255.255.0
MAC Address:	00:15:5D:FE:08:0A
Primary Role:	PDC
System Comment:	
Last Access Attempt:	-4/11/2017 1:00:02 PM
Last Connection:	-4/11/2017 1:00:02 PM
<input type="checkbox"/> Cached Credentials	
Cached Username:	
Cached Password:	
Status:	<OK>
Asset Tag:	

OK Cancel Apply Help

## Discover Database Privileged Accounts

Privileged Identity supports changing passwords for the explicit accounts contained within a database, such as SA in Microsoft SQL Server, or System in an Oracle database. This section documents how to discover local database accounts for various database types.

### Discover IBM DB2 Privileged Accounts

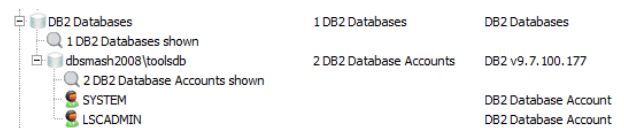
You can discover privileged accounts on a DB2 instance after it is enrolled.



**Note:** Additional OLE DB drivers are not required to manage passwords within DB2. They are only required to enumerate accounts associated with DB2. To display DB2 accounts in the user interface, install the Microsoft supplied DB2 OLE DB driver on the host system. This is not a required step to change passwords.

### To Discover DB2 Privileged Accounts Using the Management Console

1. In the management console, choose **View | Accounts Store View**.
2. Right-click the DB2 instance and choose **Refresh accounts list for account store**.



There are no account properties gathered for MySQL or MariaDB instances.

The following query is performed to get the list of accounts:

```
"SELECT * FROM SYSIBMADM.AUTHORIZATIONIDS"
```

The following query is performed to get the DB2 version information:

```
"SELECT SERVICE_LEVEL FROM TABLE (SYSPROC.ENV_GET_INST_INFO())"
```

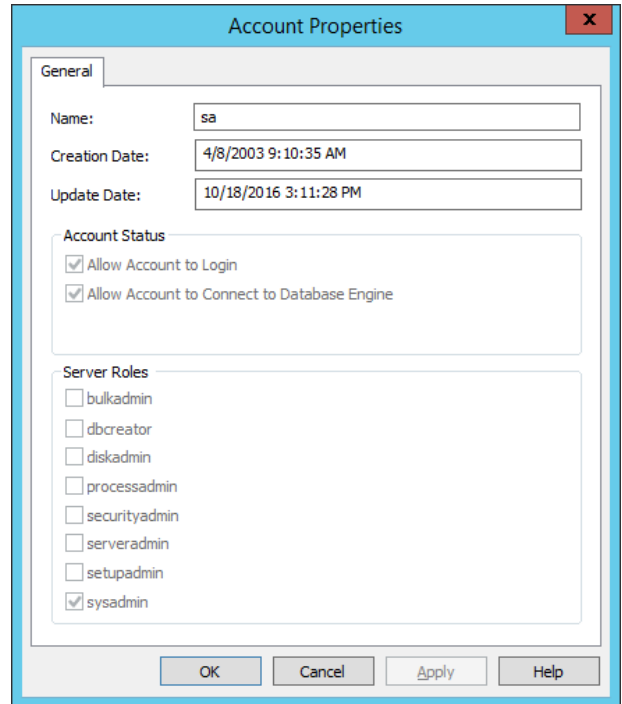
### Discover Microsoft SQL Server Privileged Accounts

You can discover privileged accounts on a SQL Server instance after it is enrolled.

## To Discover Privileged Accounts Using the Management Console

1. In the management console, choose **View | Accounts Store View**.
2. Right-click the database instance and choose **Refresh accounts list for instance**.

Right-click the Microsoft SQL database account and select **Account Properties** to view more information about the account.



The following query is performed to get the list of accounts:

```
"SELECT * FROM MASTER..SYSLOGINS WHERE PASSWORD IS NOT NULL"
```

The following query is performed to get the version information:

```
"EXEC sp_server_info @attribute_id=2"
```

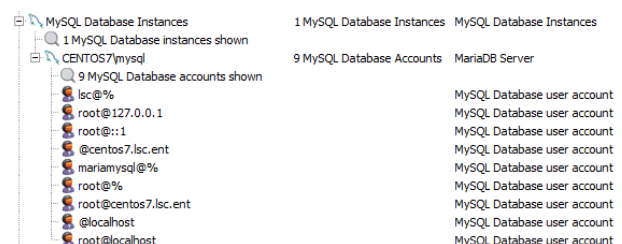
## Discover MySQL and MariaDB Privileged Accounts

You can discover privileged accounts on a MySQL or MariaDB instance after it is enrolled.

## To Discover Privileged Accounts Using the Management Console

1. In the management console, choose **View | Accounts Store View**.
2. Right-click the MySQL or MariaDB database instance and choose **Refresh accounts list for instance**.

There are no account properties gathered for MySQL or MariaDB instances.



The following query is performed to get the list of accounts:

```
"SHOW VARIABLES LIKE \"%version%"
```

The following query is performed to get the version information:

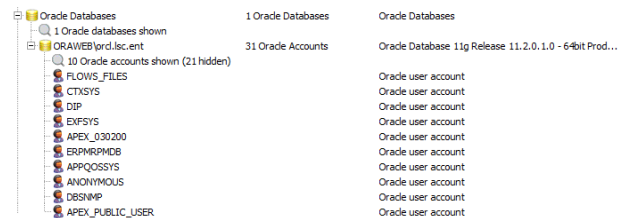
```
"SELECT host, user FROM user"
```

## Discover Oracle Database Privileged Accounts

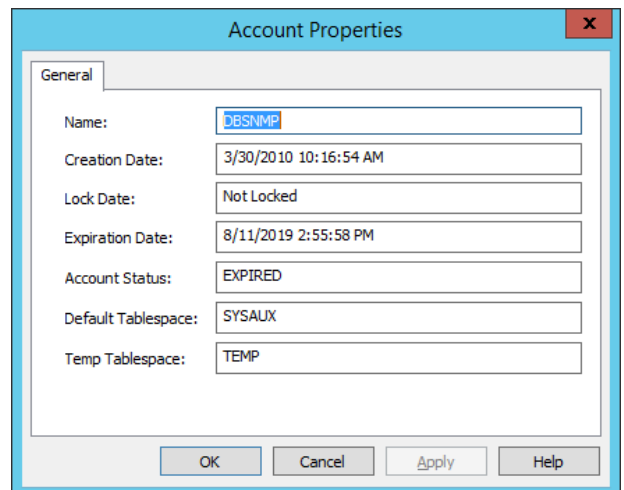
You can discover privileged accounts on an Oracle database instance after it is enrolled.

### To Discover Privileged Accounts Using the Management Console

1. In the management console, choose **View | Accounts Store View**.
2. Right-click the Oracle database instance and choose **Refresh accounts list for instance**.



Right-click the Oracle database account and select **Account Properties** to view more information about the account.



The following query is performed to get the list of accounts:

```
"SELECT banner FROM v$version where banner like 'Oracle%'"
```

The following query is performed to get the version information:

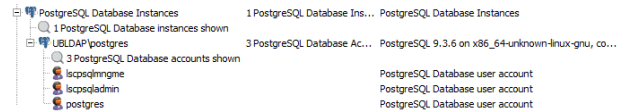
```
"SELECT username, user_id, account_status, lock_date, expiry_date, default_tablespace, temporary_tablespace, created, profile FROM DBA_USERS"
```

## Discover PostgreSQL Privileged Accounts

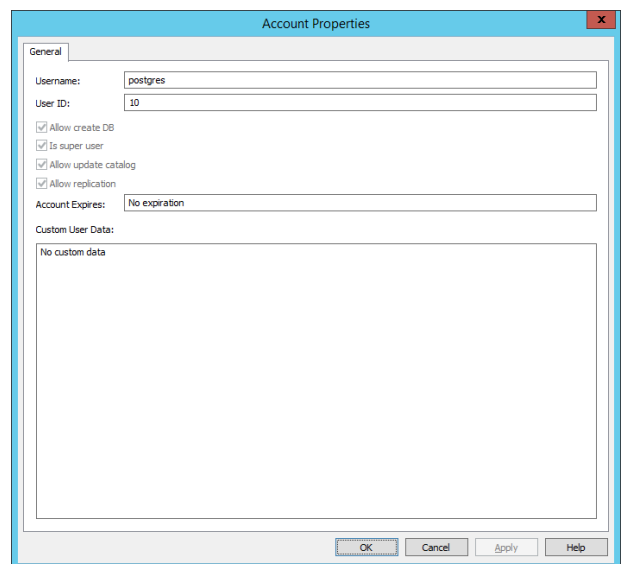
You can discover privileged accounts on a PostgreSQL instance after it is enrolled.

### To Discover Privileged Accounts Using the Management Console

1. In the management console, choose **View | Accounts Store View**.
2. Right-click the PostgreSQL instance and choose **Refresh Accounts List for Instance**.



Right-click the PostgreSQL database account and select **Account Properties** to view more information about the account.



The following query is performed to get the list of accounts:

```
"SELECT username, usesysid, usecreatedb, usesuper, usecatupd, userepl, valuntil, useconfig FROM pg_user;"
```

The following query is performed to get the version information:

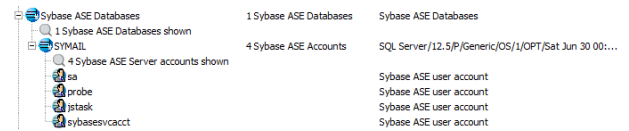
```
"SHOW server_version;"
```

## Discover Sybase ASE Privileged Accounts

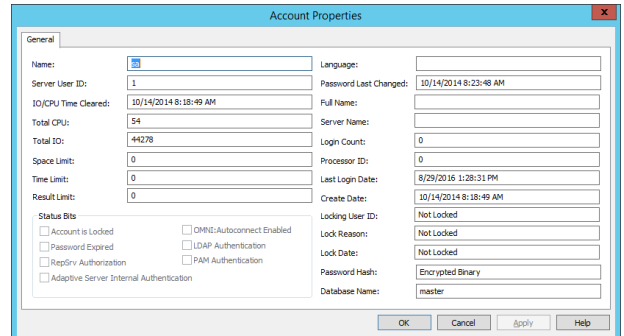
You can discover privileged accounts on a Sybase instance after it is enrolled.

## To Discover Privileged Accounts Using the Management Console

1. In the management console, choose **View | Accounts Store View**.
2. Right-click the Sybase instance and choose **Refresh accounts list for instance**.



Right-click the Sybase database account and select **Account Properties** to view more information about the account.



The following query is performed to get the list of accounts:

```
"EXEC sp_server_info @attribute_id=2;"
```

The following query is performed to get the version information:

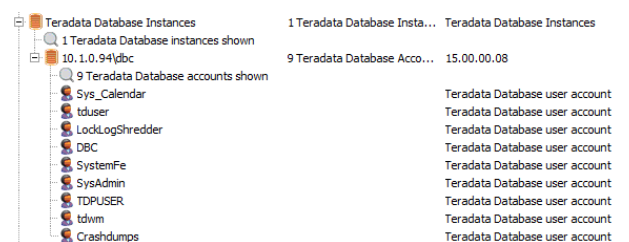
```
"SELECT suid, status, accdate, totcpu, totio, spacelimit, timelimit, resultlimit, dbname, name, password, language, pwdate, fullname, srvname, logincount, procid, lastlogindate, crdate, locksuid, lockreason, lockdate FROM master..syslogins"
```

## Discover Teradata Privileged Accounts

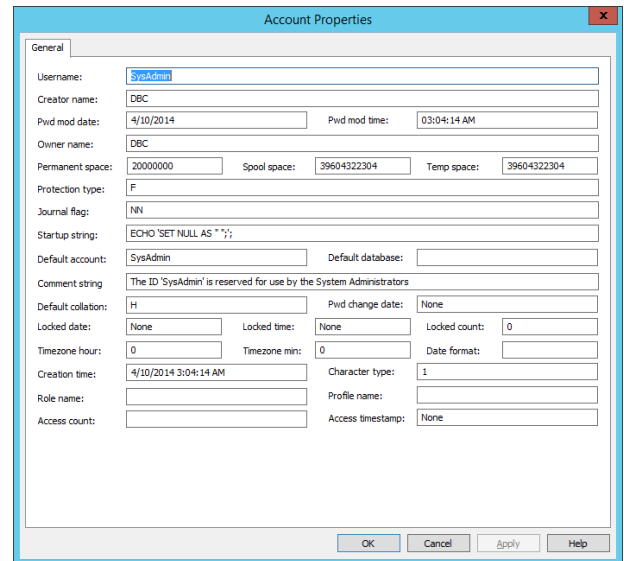
You can discover privileged accounts on a Teradata database instance after it is enrolled.

## To Discover Privileged Accounts Using the Management Console

1. In the management console, choose **View | Accounts Store View**.
2. Right-click the Teradata instance and choose **Refresh Accounts List for Instance**.



Right-click the Teradata database account and select **Account Properties** to view more information about the account.



The screenshot shows the 'Account Properties' dialog box for the 'SysAdmin' user. The 'General' tab is active, displaying the following fields:

- Username: SysAdmin
- Creator name: DBC
- Pwd mod date: 4/10/2014
- Pwd mod time: 03:04:14 AM
- Owner name: DBC
- Permanent space: 20000000
- Spool space: 39604322304
- Temp space: 39604322304
- Protection type: F
- Journal flag: NN
- Startup string: ECHO 'SET NULL AS \*';
- Default account: SysAdmin
- Default database:
- Comment string: The ID 'SysAdmin' is reserved for use by the System Administrators
- Default collation: H
- Pwd change date: None
- Locked date: None
- Locked time: None
- Locked count: 0
- Timezone hour: 0
- Timezone min: 0
- Date format:
- Creation time: 4/10/2014 3:04:14 AM
- Character type: 1
- Role name:
- Profile name:
- Access count:
- Access timestamp: None

The following query is performed to get the list of accounts:

```
"SELECT UserName, CreatorName, PasswordLastModDate, PasswordLastModTime, OwnerName, PermSpace, SpoolSpace, TempSpace, ProtectionType, JournalFlag, StartupString, DefaultAccount, DefaultDataBase, CommentString, DefaultCollation, PasswordChgDate, LockedDate, LockedTime, LockedCount, TimeZoneHour, TimeZoneMinute, DefaultDateFormat, CreateTimeStamp, LastAlterName, LastAlterTimeStamp, DefaultCharType, RoleName, ProfileName, AccessCount, LastAccessTimeStamp FROM dbc.users;"
```

The following query is performed to get the version information:

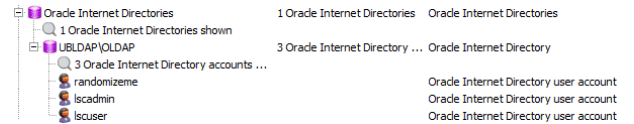
```
"SELECT InfoData from dbc.dbcinfo WHERE InfoKey = 'VERSION';"
```

## Discover LDAP Directory Privileged Accounts

Once a directory is added, Privileged Identity can discover the accounts in the directory and manage those accounts. Discovery is not necessary to take part in password management.

### To Discover LDAP Privileged Accounts Using the Management Console

1. In the management console, choose **View | Accounts Store View**.
2. Right-click the LDAP instance and choose **Refresh accounts list for account store**.



What is discovered depends on what is configured in the base LDAP container for the registration and property identifiers. To make changes to the configuration, right-click the instance and select **Change Server Registration Settings**.

No account properties are gathered during these operations.



# Discovering Middleware, Application Server, and Enterprise Software Privileged Accounts

This section documents how to discover privileged accounts and account usage on supported middleware, application servers, and enterprise software.

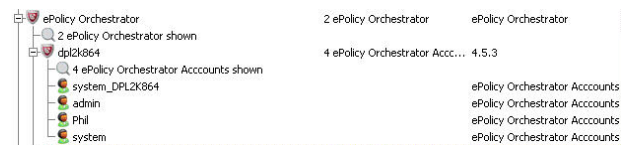
## Discover McAfee EPO Privileged Accounts

Once an EPO instance is added, Privileged Identity can discover the accounts in the directory and manage those accounts. Discovery is not necessary to take part in password management.

### To Discover McAfee EPO Privileged Accounts Using the Management Console

1. In the management console, choose **View | Accounts Store View**.
2. Right-click the EPO instance and choose **Refresh accounts list for account store**.

Account Properties are not retrieved during this operation. Users are read from the ORION database table.

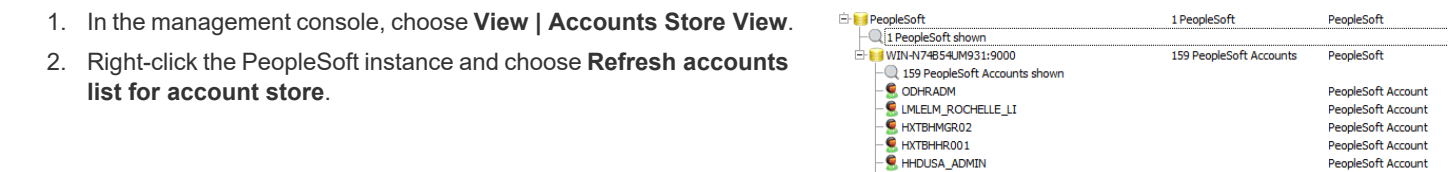


## Discover Oracle PeopleSoft Privileged Accounts

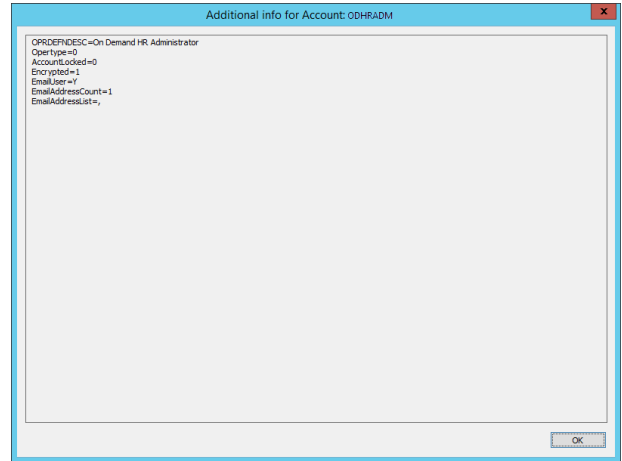
Once a PeopleSoft instance is added, Privileged Identity can discover the accounts in the directory and manage those accounts. Discovery is not necessary to take part in password management.

### To Discover PeopleSoft Privileged Accounts Using the Management Console

1. In the management console, choose **View | Accounts Store View**.
2. Right-click the PeopleSoft instance and choose **Refresh accounts list for account store**.



Right-click the Oracle PeopleSoft account and select Account Properties to view more information about the account.



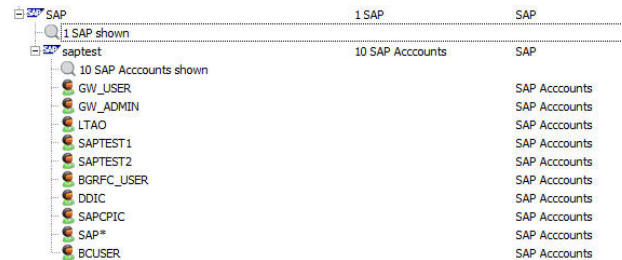
## Discover SAP Privileged Accounts

Once an SAP instance is added, Privileged Identity can discover the accounts in the directory and manage those accounts. Discovery is not necessary to take part in password management.

### To Discover SAP Privileged Accounts Using the Management Console

1. In the management console, choose **View | Accounts Store View**.
2. Right-click the SAP instance and choose **Refresh accounts list for account store**.

There are no account properties gathered for SAP instances.



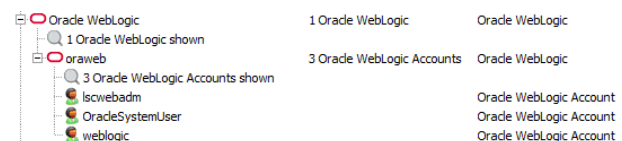
## Discover Oracle WebLogic Privileged Accounts

Once a WebLogic instance is added, Privileged Identity can discover the accounts in the directory and manage those accounts. This discovery is performed using a web call (either HTTP or HTTPS depending on how the server/integration is enrolled). Discovery is not necessary to take part in password management.

### To Discover WebLogic Privileged Accounts Using the Management Console

1. In the management console, choose **View | Accounts Store View**.
2. Right-click the WebLogic instance and choose **Refresh accounts list for account store**.

There are no account properties gathered for WebLogic instances.



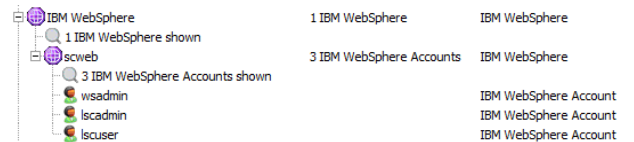
## Discover IBM WebSphere Privileged Accounts

Once a WebSphere instance is added, Privileged Identity can discover the accounts in the directory and manage those accounts. This discovery is performed using a web call (either HTTP or HTTPS depending on how the server/integration is enrolled). Discovery is not necessary to take part in password management.

### To Discover WebSphere Privileged Accounts Using the Management Console

1. In the management console, choose **View | Accounts Store View**.
2. Right-click the WebSphere instance and choose **Refresh accounts list for account store**.

There are no account properties gathered for WebSphere instances.



# Discover Network Device Privileged Accounts

Account discovery is not supported for most network devices. This section documents network devices that do support account discovery.

## Discover IPMI Privileged Accounts

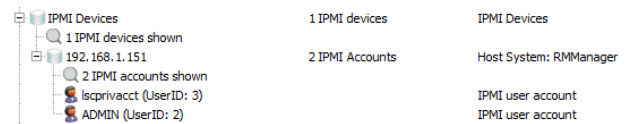
You can discover privileged accounts on a IPMI (Lights out) devices after it is enrolled.

### To Discover Privileged Accounts Using the Management Console

1. In the management console, choose **View | Accounts Store View**.
2. Right-click the IPMI device and choose **Refresh Accounts List for Instance**.

There are no account properties gathered for IPMI devices.

To view information about the system itself, right-click on the system and select **System View/Change IPMI Device Settings**.



x
IPMI Device Settings

Target Name (DNS/IP):

Asset Tag:

IP Address:

Access Account Username:

Access Account Password:

Load password for connection if stored

Anonymous Login Enabled

Null Usernames Allowed

Non-Null Usernames Allowed

User Level Authentication Enabled

OEM ID:  Manufacturer:

Device ID:  IPMI Device:

Product ID:  Product Name:

Firmware Version:

Host System:

Host OS:

System Asset Tag:

System Service Tag:

Chassis Service Tag:

Platform Model:

Host OS Status:

# Discover Cloud Services Privileged Accounts

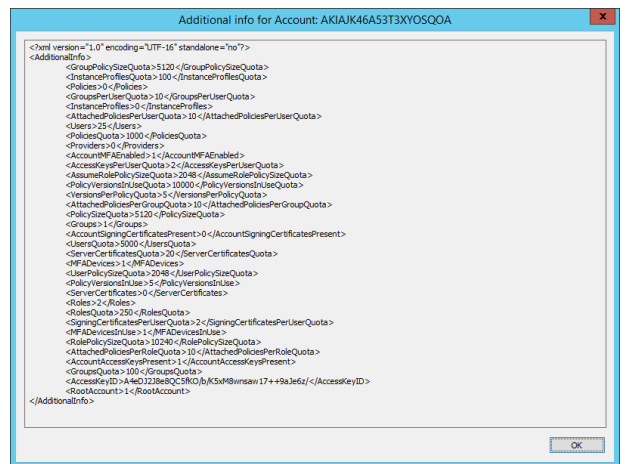
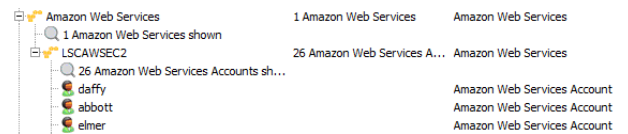
The cloud service providers in this section support account discovery.

## Discover Amazon Web Services Privileged Accounts

You can discover IAM accounts on an AWS instance after Amazon Web Services is enrolled. Federated accounts and imported accounts that may be visible in IAM are not discovered or managed.

### Discover Privileged Accounts Using the Management Console

1. In the management console, select **View > Account Store View** from the menu.
2. Right-click the **Amazon Web Services** instance, and then select **Refresh Accounts List for Amazon Web Services**.
3. To view details about an account, right-click the account, and then select **Account Properties**.

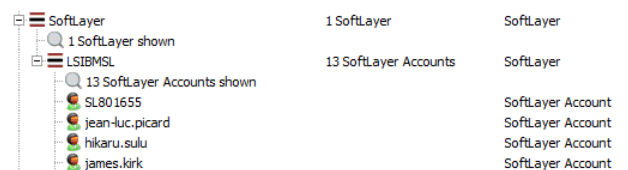


## Discover IBM SoftLayer Privileged Accounts

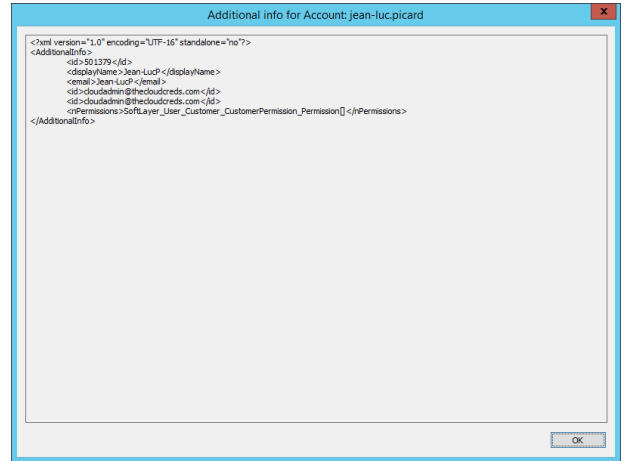
You can discover accounts on a SoftLayer instance after SoftLayer is enrolled. Federated accounts and imported accounts that may be visible in the native SoftLayer directory are not discovered or managed.

### Discover Privileged Accounts Using the Management Console

1. In the management console, select **View > Account Store View** from the menu.
2. Right-click the **SoftLayer** instance, and then select **Refresh Accounts List for SoftLayer**.



- To view details about an account, right-click the account, and then select **Account Properties**.

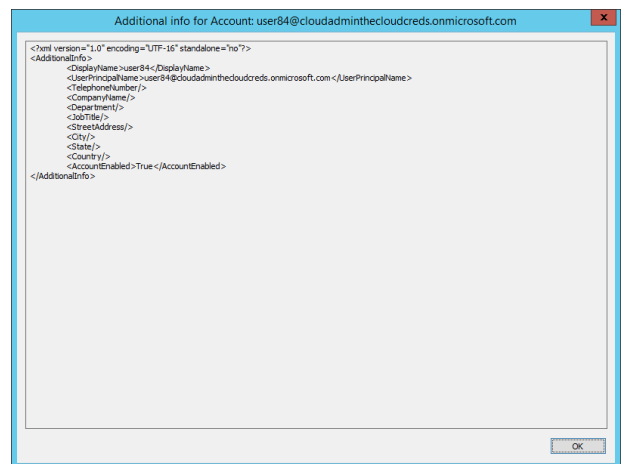
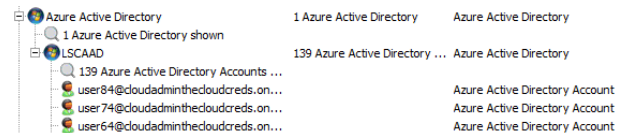


## Discover Microsoft Azure Privileged Accounts

You can discover accounts on a Microsoft Azure Active Directory instance after it is enrolled. Federated accounts and imported accounts such as "Microsoft accounts" or "Local Active Directory" accounts that may be visible in Azure AD are not discovered or managed.

### Discover Privileged Accounts Using the Management Console

- In the management console, select **View > Accounts Store View** from the menu.
- Right-click the Azure Active Directory instance, and then select **Refresh Accounts List for Azure Active Directory**.
- To view details about an account, right-click the account, and then select **Account Properties**.

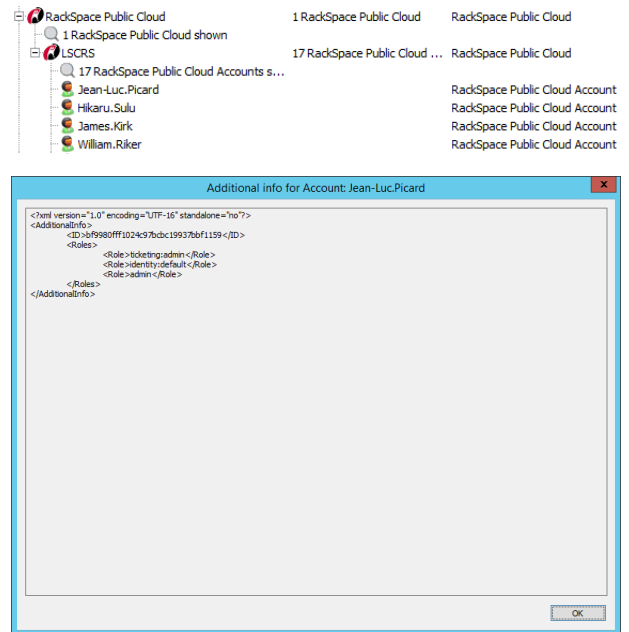


## Discover RackSpace Privileged Accounts

You can discover accounts on a Rackspace Public Cloud instance after Rackspace is enrolled. Federated accounts and imported accounts that may be visible in the native Rackspace Public Cloud directory are not discovered or managed.

## Discover Privileged Accounts Using the Management Console

1. In the management console, select **View > Accounts Store View** from the menu.
2. Right-click the RackSpace instance, and then select **Refresh Accounts List for Rackspace Public Cloud**.
3. To view details about an account, right-click the account, and then select **Account Properties**.

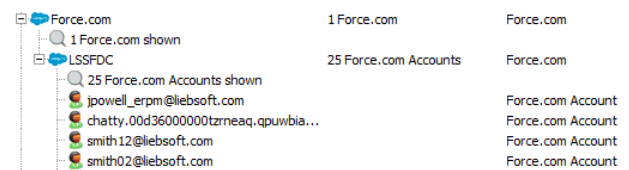


## Discover Salesforce Privileged Accounts

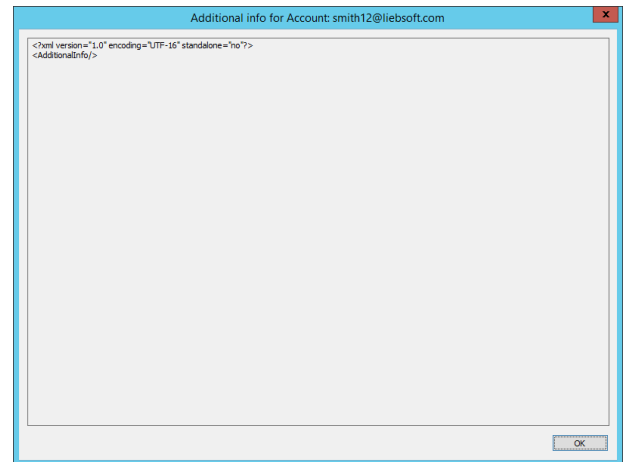
You can discover accounts on a Salesforce (Force.com) instance after Salesforce is enrolled. For Privileged Identity to find the account, the account must be enlisted with the Chatter service in Salesforce. Federated accounts and imported accounts that may be visible in the native Force.com directory are not discovered or managed.

## Discover Privileged Accounts Using the Management Console

1. In the management console, select **View > Accounts Store View** from the menu.
2. Right-click the Salesforce instance, and then select **Refresh Accounts List for Salesforce Cloud**.



3. To view details about an account, right-click the account, and then select **Account Properties**. If no additional information is configured for the account, the account info dialog will be nearly empty as depicted below.





## Discover VMware ESX Privileged Accounts

Privileged Identity can discover privileged user accounts on VMware ESX when management is performed using the VMware native connection object rather than using SSH. Account discovery is not supported if ESX is enrolled as a Linux/Unix system.

You can discover accounts on a VMware ESX/ESXi instance after it is enrolled.

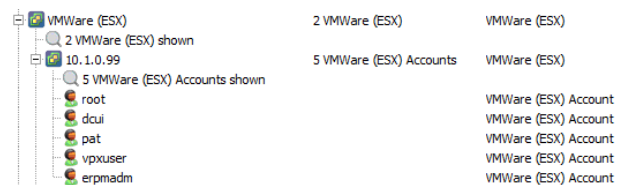


**Note:** Do not enable ESX/ESXi Lockdown mode. Doing so blocks all connections and will prevent management of the ESX host. Only the vCenter Server can manage the host when Lockdown mode is enabled.

### To Discover Privileged Accounts Using the Management Console

1. In the management console, choose **View | Accounts Store** view.
2. Right-click the VMware ESX instance and choose Refresh Accounts List for VMware ESX.

There are no account properties gathered for VMware ESX instances.



## Discover Custom Account Store Privileged Accounts

You can add custom account stores for systems and devices that can be managed via Telnet or SSHv2. Account discovery is also supported for custom account store nodes if the system or device follows the Linux/Unix paradigm of listing accounts in a **/etc/passwd** or **/etc/shadow** file that will be managed with SSH. If the device/operating system does not follow this convention, then account discovery is not supported.

To perform discovery of accounts on the target device/system, right-click the node(s) under the custom account store and choose **Refresh Accounts List for Account Store**.

## Configure Scheduled Job Options

Depending on the job type created, for example, password change job or system refresh job, additional information such as password constraints, pre-run operations or job schedule will need to be configured.

This chapter outlines how to configure all password change options for all job types.

This chapter also outlines how to configure job priority. Job priority allows jobs to be run outside of the default order they may have otherwise been run.

### Job Priority

The Privileged Identity job queue processes jobs on a first-in-first-out basis, based on their job schedule. If two jobs are set to run at the same time, the job with the lower job ID (older) will be run first. While this tie breaker works fine for most cases, it may be desired to give priority to certain jobs types. For example, three jobs are created in the following order: password change job, system refresh job, password change job. Assuming all three jobs are set to run every day at noon, they would be processed in order they were created, based on the job ID being the tie breaker.

As certain job types may hold a higher value than others, job priority allows manipulating the tie breaker process. All jobs are created with a default job priority of "50". The default priority of a job type or a job created by a certain interface (web service vs console vs web application), can be manipulated to give priority to one type of job over another. In addition, individual job priorities can be assigned directly to a job.

Using the original scenario above, three jobs are created in the following order: password change job, system refresh job, password change job, but changing the default priority for password change jobs to "51", the processing order would be password change job, password change job, system refresh job. The two password change jobs having the same priority would break their tie using the lower job ID (older) as mentioned above.

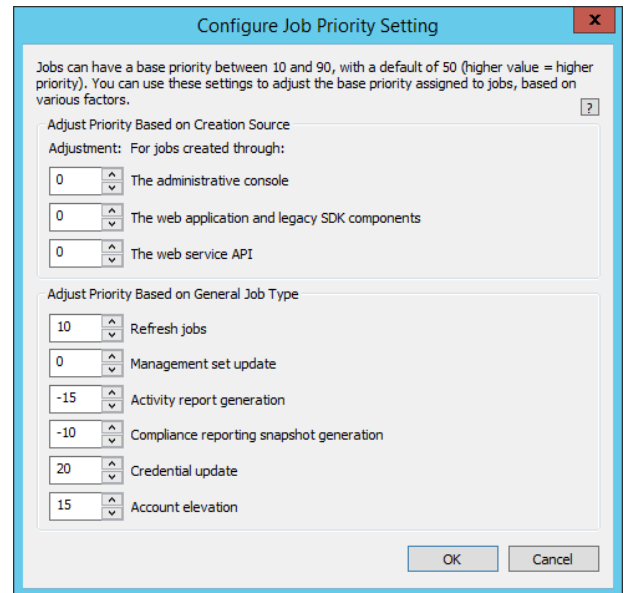
### Change Default Job Priority

Job priorities for existing jobs will not be automatically updated if the base priority is adjusted. A new base priorities only affect newly created jobs. Individual jobs may have their priorities changed by right-clicking on the job and selecting **Set Job Priority**.

Jobs can have a base priority between 10 and 90 with a default value of 50. A higher value equals higher priority.

1. From the **Actions** pane in the management console, click **Jobs**.
2. In the **Jobs** dialog, select **Job Options > Default Job Priority**.

3. Select to adjust the base priority for creation source, job type, or both. Values can be both positive and negative as they will adjust the job priority up or down from the default priority of 50.
4. Click **OK**.



**Configure Job Priority Setting** [X]

Jobs can have a base priority between 10 and 90, with a default of 50 (higher value = higher priority). You can use these settings to adjust the base priority assigned to jobs, based on various factors.

Adjust Priority Based on Creation Source [?]

Adjustment: For jobs created through:

0	The administrative console
0	The web application and legacy SDK components
0	The web service API

Adjust Priority Based on General Job Type

10	Refresh jobs
0	Management set update
-15	Activity report generation
-10	Compliance reporting snapshot generation
20	Credential update
15	Account elevation

OK Cancel

## Change Default Job Priority

The Privileged Identity job queue processes jobs on a first-in-first-out basis, based on their job schedule. If two jobs are set to run at the same time, the job with the lower job ID (older) will be run first. While this tie breaker works fine for most cases, it may be desired to give priority to certain jobs types. For example, three jobs are created in the following order: password change job, system refresh job, password change job. Assuming all three jobs are set to run every day at noon, they would be processed in order they were created, based on the job ID being the tie breaker.

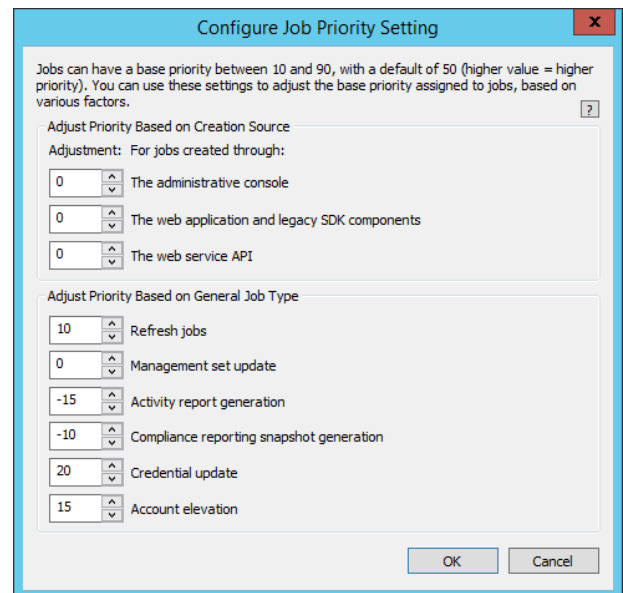
As certain job types may hold a higher value than others, job priority allows manipulating the tie breaker process. All jobs are created with a default job priority of "50". The default priority of a job type or a job created by a certain interface (web service vs console vs web application), can be manipulated to give priority to one type of job over another. In addition, individual job priorities can be assigned directly to a job.

Using the original scenario above, three jobs are created in the following order: password change job, system refresh job, password change job, but changing the default priority for password change jobs to "51", the processing order would be password change job, password change job, system refresh job. The two password change jobs having the same priority would break their tie using the lower job ID (older) as mentioned above.

Job priorities for existing jobs will not be automatically updated if the base priority is adjusted. A new base priorities only affect newly created jobs. Individual jobs may have their priorities changed by right-clicking on the job and selecting **Set Job Priority**.

Jobs can have a base priority between 10 and 90 with a default value of 50. A higher value equals higher priority.

1. From the **Actions** pane in the management console, click **Jobs**.
2. In the **Stored Jobs** dialog, select **Job Options > Default Job Priority** from the top menu.
3. Select to adjust the base priority for creation source, job type, or both. Values can be both positive and negative as they will adjust the job priority up or down from the default priority of 50.
4. Click **OK**.



X

**Configure Job Priority Setting**

Jobs can have a base priority between 10 and 90, with a default of 50 (higher value = higher priority). You can use these settings to adjust the base priority assigned to jobs, based on various factors. ?

**Adjust Priority Based on Creation Source**

Adjustment: For jobs created through:

0	▲ ▼	The administrative console
0	▲ ▼	The web application and legacy SDK components
0	▲ ▼	The web service API

**Adjust Priority Based on General Job Type**

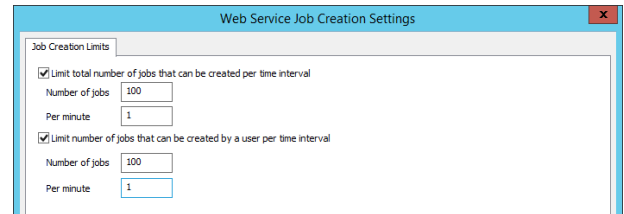
10	▲ ▼	Refresh jobs
0	▲ ▼	Management set update
-15	▲ ▼	Activity report generation
-10	▲ ▼	Compliance reporting snapshot generation
20	▲ ▼	Credential update
15	▲ ▼	Account elevation

OK
Cancel

## Throttle Job Creation

Programmatic job creation via the web service or PowerShell can overload the web service host if the target web service host is not given sufficient resources to process all the incoming requests.

1. From the **Actions** pane in the management console, click **Jobs**.
2. In the **Jobs** dialog, select **Job Options > Web Service Job Creation Limits**.
3. Enable one or both of the limiting options and define the maximum number of jobs that can be created in the per-minute interval.
4. Click **OK**.



Web Service Job Creation Settings

Job Creation Limits

Limit total number of jobs that can be created per time interval

Number of jobs

Per minute

Limit number of jobs that can be created by a user per time interval

Number of jobs

Per minute

## Create Custom Communication Types

Custom communication types represent a special configuration for SSH and Telnet targets to specify a default answer file be used for any and all management operations against a list of machines (defined by management set) or a specific machine.

Normally, when a password change job is performed against an SSH or Telnet target, the job uses a singular answer file. This means all target systems need to use the same exact configuration (answer file) and port configuration. This in turn forces you to create multiple jobs for different sets of machines even when the same overall process is being followed across all machines.

### Example 1:

A root account connects remotely and issues the `passwd` command to change its own password. After the `passwd` command is issued, the target systems all ask for the new password and to re-type the new password. In the case of one Linux distro, following the second password input, it returns a message of "All login tokens successfully updated". In the case of a second Linux distro, following the second password input, it does not return any message on success but returns simply to the # prompt. In the case of all distros, whether success or failure is reached they always return to a prompt that includes a #.

In this example, if you modified the final StdOut to look for a # the job would always report success because on either success or failure, there is a # prompt.

Normally, you would create two password change jobs using two different answer files. The first answer file would look for the "successfully updated" prompt and the second answer file would create an Exclude step that indicates so long as "Error" (or another failure message) was not returned, consider the job a success.

### Example 2:

You create lots of SSH or Telnet-based jobs on a regular basis. The creation process always defaults to the default response file, Response. You have to then browse to the correct answer file every time.

### Example 3:

You must use an SSH Tunnel to connect to a target machine.

With custom communication types, you can create a configuration such that when a system or list of systems is included in a job, they always end up using a particular answer file, no matter what is specified on the job. When this function is triggered, the job log will indicate there is a custom communication type configured and that it was used for the job on that particular system.

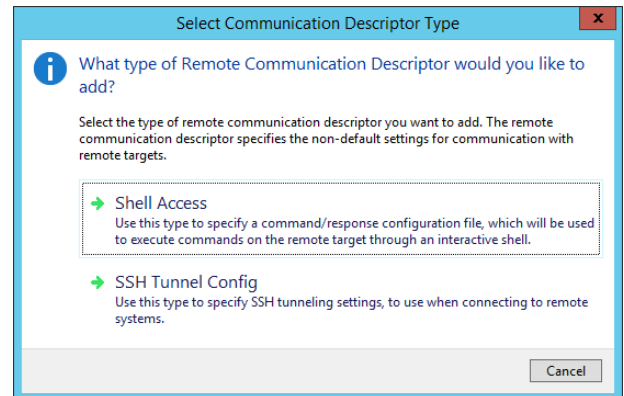
There are two types of custom communication types:

- **Shell Access:** Use this when connection directly to a shell on a target system.
- **SSH Tunnel:** Use this when an SSH Tunnel must be configured and established prior to connecting to the target system.

## Create a Custom Communication Type for Shell Access

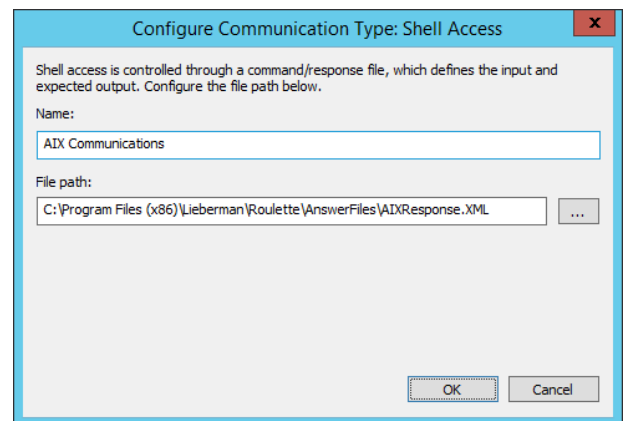
1. In the management console, go to **Systems List > Custom Communication Types > Configure Communication Types**.
2. On the **Configure Communication Types** dialog, click **Add**.

3. Select **Shell Access** for the Communication Descriptor Type.



4. Provide a friendly name for the communication type and provide the path in the local file system to the custom response file.

5. Click **OK** to add the custom communication type.



6. Associate the custom communication type with system.

**i** For more information on associating custom communications with targets, please see "[Associate Custom Communications with Targets](#)" on page 222.

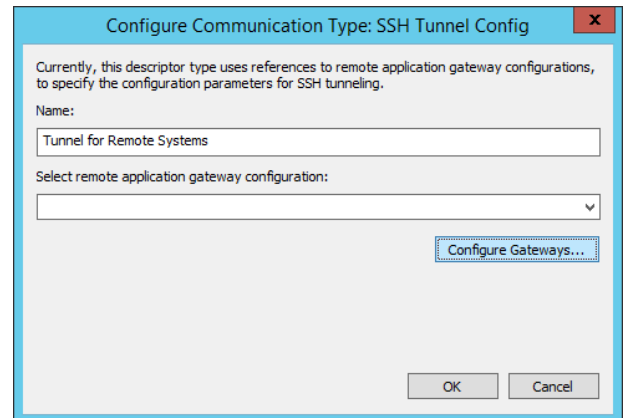
## Create a Custom Communication Type for SSH Tunnel Access

Configuration of SSH Tunnels is not a typical network scenario and will involve the use of extra configuration items such as an SSH proxy.

1. In the management console, go to **Systems List > Custom Communication Types > Configure Communication Types**.
2. On the **Configure Communication Types** dialog, click **Add**.
3. Select **SSH Tunnel Config** for the Communication Descriptor.

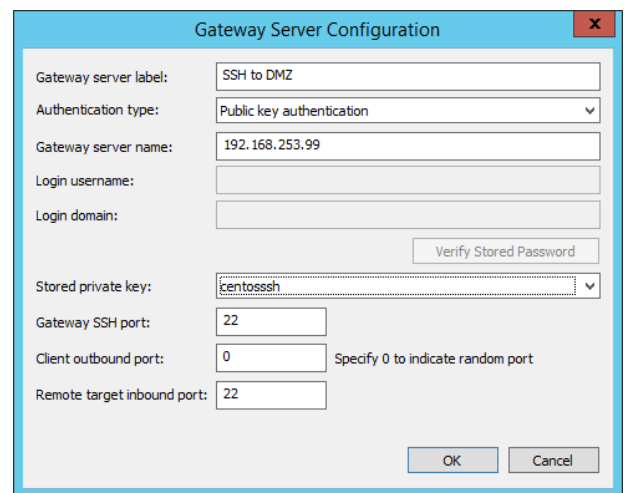


4. Provide a friendly name for the communication type, and then click **Configure Gateways**.

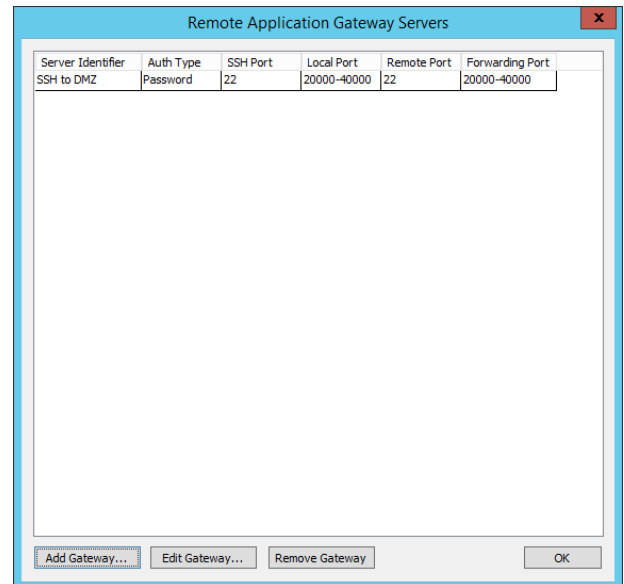


5. On the **Remote Application Gateway Servers** dialog, click **Add Gateway**.
6. Supply the following information on the **Gateway Server Configuration** dialog:

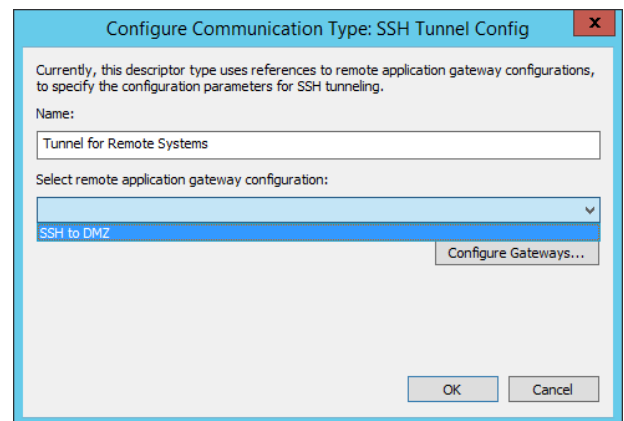
- **Gateway server label:** This is a friendly name that will appear in the management console.
- **Authentication type:** Configure as **Username and password** or **Public Key Authentication**. In both situations, the username or SSH key already must be stored in the password store.
  - **Username and password:** Supply the Login username and Login domain. The login domain will be the namespace of the account used to connect to the SSH gateway such as [Cisco] or [Linux].
  - **Public key authentication:** Select the stored private key to use for authentication.
- **Gateway server name:** Supply the name or IP of the system that will function as the SSH gateway. If using Username and password authentication, this system name must already have a stored/managed password present in the secure password store.
- **Gateway SSH port:** The target port to connect to on the SSH Gateway Server.
- **Client outbound port:** Specify an outbound port or leave as '0' to indicate a random port (recommended).
- **Remote target inbound port:** Specify the target inbound port of the target system on the remote side of the SSH gateway.



7. Click **OK** to add the custom communication type. If any errors were made, click **Edit Gateway** to edit the entry.
8. Click **OK**.



9. From the **Select remote application gateway configuration** drop-down, pick the SSH gateway configuration you just created.



10. Associate the custom communication type with system.

**i** For more information on associating custom communications with targets, please see "[Associate Custom Communications with Targets](#)" on page 222.

**i** For more information on available namespaces, please see "[Namespace Values](#)" on page 589.

## Associate Custom Communications with Targets

Once a custom communication type has been configured, it must be associated with one or more systems.

1. In the management console, go to **Systems List > Custom Communication Types > Associate Type with Targets**.
2. On the **Configure Communication Type Usage** dialog, click **Add**.

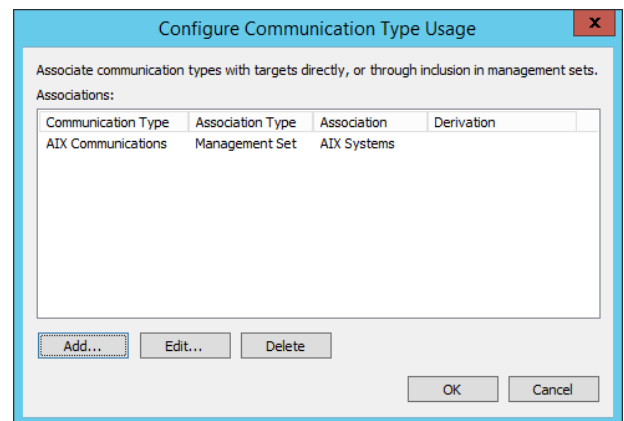
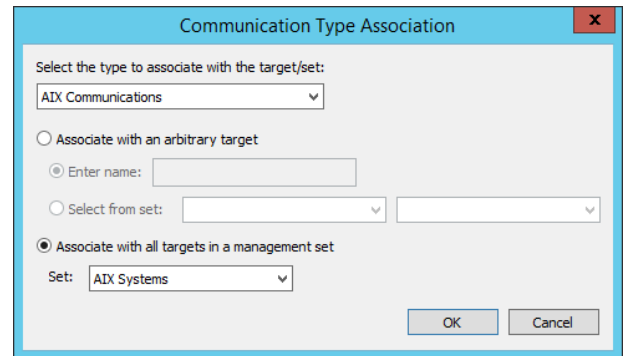
3. On the **Communication Type Association** dialog, supply the following configurations:

- **Select the type to associate with the target/set:** Select one of the custom communication types previously configured.
- **Associate with...:**

- **Associate with an arbitrary target:** Either type in the name of a single system or select a single system from a particular management set.
- **Associate with all targets in a management set:** Select a management set to target the custom communication list with. All SSH/Telnet access for all SSH/Telnet devices in that list will adhere to the custom communications configurations.

4. Click **OK**.

5. Click **OK** to finish.



When you create your password change jobs, none of this information will be visible in the password change job dialog. That means your answer files will still reflect the default values or whatever you specify for the answer file. However, for systems that are configured with a custom communication type, they will always use the custom communication types despite any other configurations.



**IMPORTANT!**

*Do not create overlapping custom communication types. For example, do not configure the same system to use two different answer files. Privileged Identity does not provide a control for which answer file would be used on any job run.*

## Enable Account Pooling

Account pooling is an optional feature. Account pooling allows rotation of the run as identity information and the password of the account being rotated to.

Consider a scheduled task and a service running as svcpool-1 across 100 servers. For reasons not yet known, three servers cannot be successfully managed. Without account pooling, the result is that 97 servers are updated successfully and come up using the new credential, and three attempt to use the old credential. The result is an account lockout scenario for svcpool-1 that will likely lead to a service disruption on the other 97 servers. Notifications and alerts can be sent and action-responses can be triggered, but more can be done to avoid such a situation in the first place.

The solution is account pooling. Using the scenario above but with account pooling, a pool of accounts will be configured using: svcpool-1, svcpool-2, svcpool-3. These three accounts are duplicates of each other in every way, especially in terms of rights and permissions.

When it is time to change the password, rather than updating the password on svcpool-1, the password of svcpool-2 will be rotated. Everything that can be updated successfully, currently running as svcpool-1, will be set to run as svcpool-2.

The result with the second scenario will be that 97 servers will be running their services and tasks as svcpool-2 with the new password and the remaining three will be running as svcpool-1 with the old, unchanged password. The result, no service disruption. Based on retry settings, the solution will continue to attempt updating the failed three servers. On the subsequent job run, Privileged Identity will not randomize svcpool-1 or svcpool-2, but instead will randomize svcpool-3's password and update everything possible to run as svcpool-3.

The intent is to minimize any possible disruption during password changes and establish a new Window of opportunity for the administrator find unknown places of service account usage, such as embedded scripts and applications, as well as to fix any other problem servers.

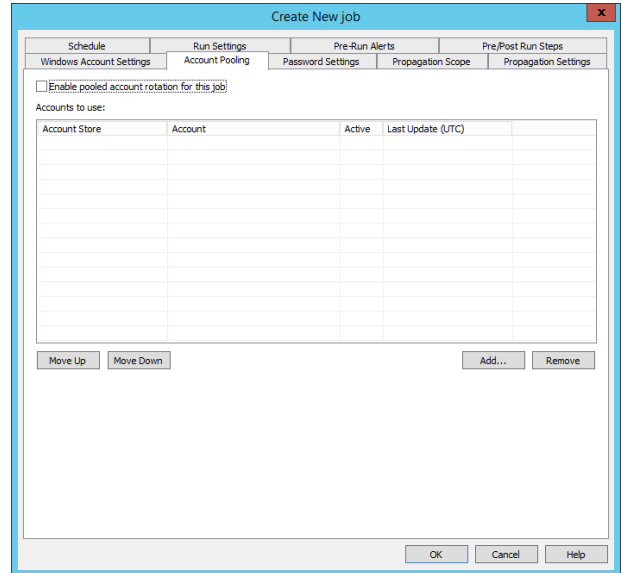
Account pooling is applicable to Windows accounts used on Windows servers used by the following items:

- Windows Services
- Windows Scheduled Tasks
- IIS 6 anonymous accounts
- IIS 6 application pools
- IIS 6 network credentials
- COM applications
- DCOM applications

To use account pooling, multiple accounts with all the appropriate rights must pre-exist and be valid for any and all locations being updated.

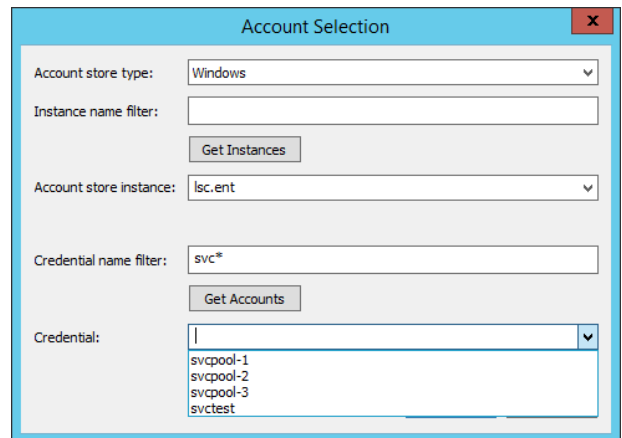
If the **Account Pooling** tab is not visible, the feature is not enabled by your license or you are not targeting a named account but are instead targeting any of the built-in types, such as Built-in Administrator or Built-in Guest or DSRM.

To create an account pool, when configuring the password change job, go the **Account Pooling** tab and click, select **Enable pooled account rotation for this job**, and then click **Add**.



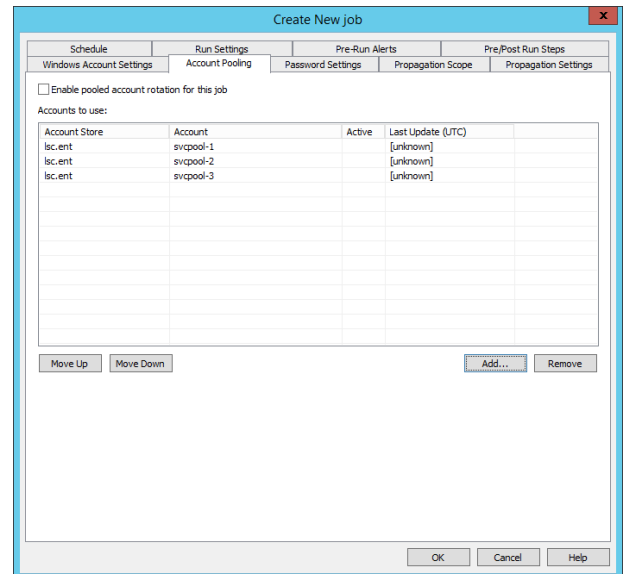
To add accounts to the pool:

1. Select the **Account store type**. The default is **Windows**.
2. Type the name of the account store instance or select it from the **Account store instance** drop-down list. This is the name of the system from which the managed accounts will be drawn. If the list of systems is too long, supply the name of the Use the **Instance name filter** to type in the partial name of an account store (Windows system) using the asterisk (\*) as the wild card holder then select the system name from the now filtered list for Account store instance.
3. Type in the name of the credential in the credential field or pick it from the list. If the list of account names is too long, supply the name of the Use the **Credential name filter** to type in the partial name of an account using the asterisk (\*) as the wild card holder then select the system name from the now filtered list for the credential.
4. Click **OK**.
5. Repeat the above steps to add as many accounts as desired/needed to define the account pool.



- To re-order the list and change the order in which the pooled accounts are applied, select the account and then use the **Move Up** or **Move Down** buttons to move an account up or down in the list. The currently active account will be identified as **[active]** in the active column.

The first account rotated will be the account named on the **Windows Account Settings** tab. That will then mark that account as the "Active" account by placing the word '[Active]' in the 'Active' column. The **Last Update (UTC)** column will then be updated with the UTC time stamp when the password was last updated. The next account to be updated will be the account immediately under the **Active** account. This information can be seen by examining the job details from the **Jobs** dialog (from the **Actions** pane, click **Jobs**).



## Configure Password Settings

When configuring a password change job against any platform, after the target (and login) account names have been specified, the password settings for the job need to be configured.

When configuring these password change settings, it is important to be aware of the following items:

- Maximum and minimum password length restrictions relevant to the target platform. Each platform, and even variations of some platforms have different password restrictions regarding length, types of characters, types of characters in certain positions, and so on. See Known Password Constraints for a list of known password limitations against target platforms.
- Whether or not the password should be re-randomized following retrieval of the password from the solution.
- Other options regarding the account such as unlocking the account during the password rotation.
- Whether or not it require more than one person to retrieve the password.

## Known Password Constraints



**Note:** The following notes represent a point in time list of known constraints based on available documentation from the specific vendor and should be validated against any device you are managing.



**Note:** Privileged Identity does not quote/group multi-word passwords or escape special characters. As a results, some special characters that may have been allowed by the device if they were grouped or escaped may not be used.

This list will be updated as new information is presented.

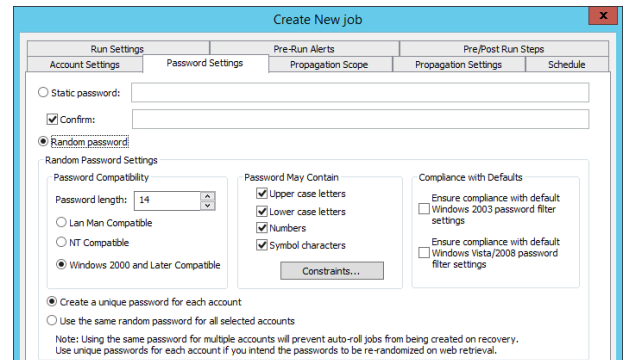
- Windows NT4:
  - Maximum length of 14 characters
- Windows 2000 and later:
  - Maximum length of 127 characters (Privileged Identity limitation).
- Cisco devices:
  - Maximum length of 25 characters.
  - Do not use these special characters: ? @ !
- Dell DRAC:
  - Maximum of 10 characters.
- Special characters should be limited to hyphen (-) or underscore (\_).Mainframe:
  - Special characters may not be allowed or may be restricted in your installation.
  - Maximum length of 8 characters.
- MySQL/MariaDB:
  - Recommended to limit special characters to: \_ or dot (.)
  - Maximum length of 41 characters.

- Oracle databases:
  - Maximum length of 30 characters.
  - The first character in an Oracle password must be a letter.
  - Only letters, numbers, and the symbols “#”, “\_” and “\$” are acceptable in a password.
  - Case sensitive passwords must be enabled in the Oracle database.
- PostgreSQL:
  - Recommended to limit special characters to: \_ or dot (.)
  - Maximum length of 63 characters.
- Sybase ASE:
  - Maximum length of 30 characters.
- Teradata
  - Only letters, numbers, and the symbols “#”, “\_” and “\$” are acceptable in a password.
  - Maximum length of 30 characters.

Be sure to check with your vendor and product documentation and your device/system administrator for other restrictions and policies that may apply.

## Password Settings Options

- **Static password:** Define a static password. It will be set once when the job is run. The admin who filled out the job information will know what the password is. The password will remain this value until it is reset with a new job. It will never roll to a new value following a password retrieval. Anyone who retrieves this password will then know the password until you re-set the password with a new job.
- **Random password:** Define password generation settings including length, allowed characters, minimum numbers of characters types and more using the password constraints. With default web application settings, when the password is retrieved via the web interface, the password will be triggered for re-randomization in the near future. This means the password will change not only based on the job schedule (schedule tab) but will also change a short time after a user retrieves the password from the web interface. Use the following settings to control the random password generation settings:
  - **Password Compatibility:** Ultimately these control the length of the password and allowed character types.
    - **Lan Man Compatible:** 14 character max, no lower case letters.
    - **NT Compatible:** 14 character max, all character types allowed.
    - **Windows 2000 and Later Compatible:** 127 character max, all character types allowed.
  - **Password May Contain:** Note the header is not WILL CONTAIN. Identifies the types of characters that may be contained in the password. Use the Constraints button to identify what it WILL contain.
  - **Compliance with Defaults:** Ensure compliance with default {OS VERSION} password filter setting represents the quickest way to turn on password constraint settings that will meet the target password filter requirements. For example, when you enable either one of these options, each character type will be set to require a minimum of one of the defined character types. For example, one upper case, one lower case, one symbol, and one number. Use the Constraints button to further define the requirements.



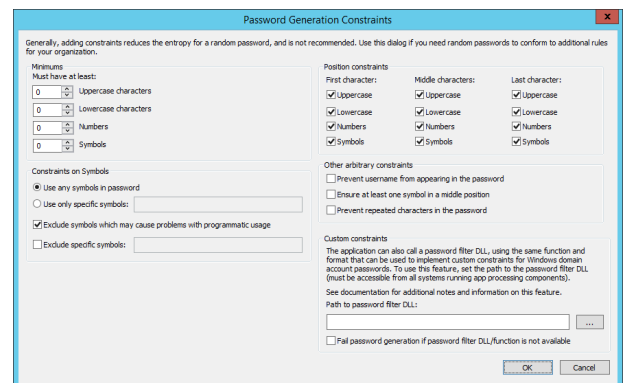


- **Create a unique password for each account:** With password generation being set to Random and this option selected (default) the following will be true:
  - If more than one system is included in the job, the named account on each system will receive a unique password based on the password generation settings.
  - When the password for the account is retrieved via the web site, the password will, by default, be triggered for re-randomization in just two hours (default) from that point.
- **Use the same random password for all selected accounts:** With password generation being set to Random and this option selected (default) the following will be true:
  - If more than one system is included in the job, the named account on each system will receive the exact same random password.
  - When the password for the account is retrieved via the web site, the password will NOT be triggered for re-randomized following. This makes this setting ideal for service accounts where you wish to re-randomize accounts like root or local administrator following use, but do not wish to incur a service disruption following password retrieval of a service account.

## Password Constraints

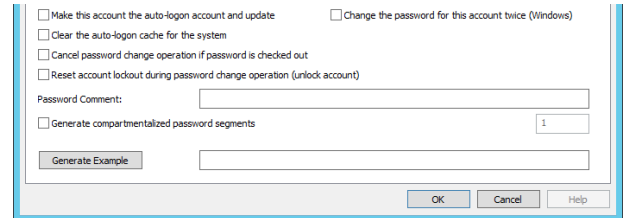
The password constraints sets up the various character restrictions or requirements when using a random password.

- **Minimums:** Defines the minimum numbers of a specific character type to include in the password. When set to '0', it means the password may contain this character type. When set to '1' or more, it means the password will contain one or more of this character type. Do not set a character requirement that would put the required length greater than the random password length. In other words, if your password is set to be 14 characters, do not set each of the minimums to be 4 characters each as that would require a 16 character password. In this scenario, the job will fail to generate a password and this will not run. It is better to leave the values lower to create a higher level of entropy.
- **Constraints on Symbols:** Use this section to define the types of special that may be included or excluded from a password. Choose to use any symbols or specific symbols or exclude specific symbols. By default certain symbols are excluded that may cause problems with programmatic usage. These symbols are: `\/:;'"`
- **Position Constraints:** Use this section to define the types of character constraints for relative position in a password. For example, some platforms require the first character be a letter or number but cannot be a symbol.
- **Other arbitrary constraints:** Use this section to further limit possible password generation settings, such as preventing the username from appearing in the password, ensuring at least one symbol in the middle position and preventing repeated characters (two or more of the same character in a row).
- **Custom constraints:** If you have been provided or created a custom password filter DLL, those settings can be read by this solution and used to automatically configure the password constraints according to the password filter DLL settings. This password filter is not provided by BeyondTrust.



## Other Options During Password Update

- **Make this account the auto-logout account and update:** For Windows systems only, the target account will be configured as the automatic login account on the target system. Note this does not push out credentials to systems that are part of a propagation scope, it only affects the target system where the password is being changed.
- **Clear the auto-logout cache for the system:** For Windows systems only, clears all auto-logout settings on the target system.
- **Cancel password change operation if password is checked out:** For all system types, if the password is currently checked out and the password re-randomization interval occurs or the normal job schedule interval occurs, the job will not run and instead will log a failure and retry the job according to the global job retry policy.
- **Reset account lockout during password change operation:** Enabling this option will unlock a locked account for the following system types:
  - Oracle Database
  - Windows
- **Change the password for this account twice (Windows):** Also called "Double Tap", is used to change the password two times in a row. This achieves two goals:
  - Forces urgent replication in Active Directory which is useful when a single service account is used across multiple AD sites to reduce replication latency and improve uptime.
  - Resetting twice the built-in Key Distribution Service account (KRBTGT) password will make invalid any golden tickets created with the previously stolen KRBTGT hash as well as all other Kerberos tickets.
- **Password Comment:** Add a comment to the password change job. The password comment will be visible in the web site.
- **Generate compartmentalized password segments:** Also referred to as "4 Eyes", allows you to specify the password be divided into a number of segments (up to the number of characters in the password). This in turn will require that number of people be required to obtain the entire password.



The screenshot shows a dialog box with the following elements:

- Checkboxes:
  - Make this account the auto-logout account and update
  - Change the password for this account twice (Windows)
  - Clear the auto-logout cache for the system
  - Cancel password change operation if password is checked out
  - Reset account lockout during password change operation (unlock account)
  - Generate compartmentalized password segments
- Text input field: Password Comment: \_\_\_\_\_
- Number input field: \_\_\_\_\_ (with a small '1' in a box next to it)
- Button: Generate Example
- Buttons: OK, Cancel, Help



For more information, please see *"Work with Compartmentalized Passwords (Four Eyes)"* on page 446.

# Use Pre and Post Run Steps to Run Scripts and Applications

Use the **Pre/Post Run Steps** tab to run scripts and other applications before and after a job runs. For example, you can open a software-defined networking (SDN) connection just prior to running a job, and then immediately close the connection when the job finishes.



## IMPORTANT!

*Be careful when using this feature in an environment that has more than one zone processor running on the same system at the same time. If the zone processors both run jobs that utilize the same scripts at the same time, a race condition can result.*

To configure pre-run and post-run steps, complete the form fields on the **Pre/Post Run Steps** tab.

- **Run application before starting operation:** (Optional) Enable this option to run a script or another application before the job runs.
  - **Application path:** Enter the absolute path of the script or application file to execute before the job runs. For scheduled jobs, your deferred processor service account must have **Read** and **Execute** access to the program.
  - **Input arguments:** (Optional) Enter input arguments in the appropriate format for the target application. Arguments are passed to the command line exactly as entered. For example, if you enter input arguments for **example.exe** as **p1 p2 p3**, your application runs as **example.exe p1 p2 p3**.
  - **Wait for application to exit:** (Optional) Enable this option if the pre-run program must fully complete and return a result code before the rest of the job can be run.
  - **Maximum wait time (seconds):** (Optional) Set a maximum length of time to wait for the pre-run program to return a result code. If the pre-run program does not return a code within the allotted number of seconds, continue with the job process.
  - **Abort operation if application returns non-zero exit code:** (Optional) Normally, an application that terminates correctly returns an exit code of 0. If this option is enabled, the entire password change job aborts if the target returns an exit code that is not zero.
  
- **Run application after finishing operation:** (Optional) Enable this option to run a script or another application immediately after the job runs.
  - **Application path:** Enter the absolute path of the script or application file to execute after the job runs. For scheduled jobs, your deferred processor service account must have **Read** and **Execute** access to the program.
  - **Input arguments:** (Optional) Enter input arguments in the appropriate format for the target application. Arguments are passed to the command line exactly as entered. For example, if you enter input arguments for **example.exe** as **p1 p2 p3**, your application runs as **example.exe p1 p2 p3**.
  - **Wait for application to exit:** (Optional) Enable this option if the post-run program must fully complete and return a result code before counting the job as complete.
  - **Maximum wait time (seconds):** (Optional) Set a maximum length of time to wait for the post-run program to return a result code. If the post-run program does not return a code within the allotted number of seconds, mark the job process as complete.

Pre-run and post-run steps include two variables you can use in your input arguments:

- **\$JobID**: ID of the currently running job.
- **\$JobComment**: Comment associated with the job as defined by **Job Details > Schedule > Job Comments**.

## Configure Pre-Run Alerts

A pre-run alert allows an email notification to be sent to a semi-colon delimited list of email addresses a certain number of hours before the job is scheduled to run.

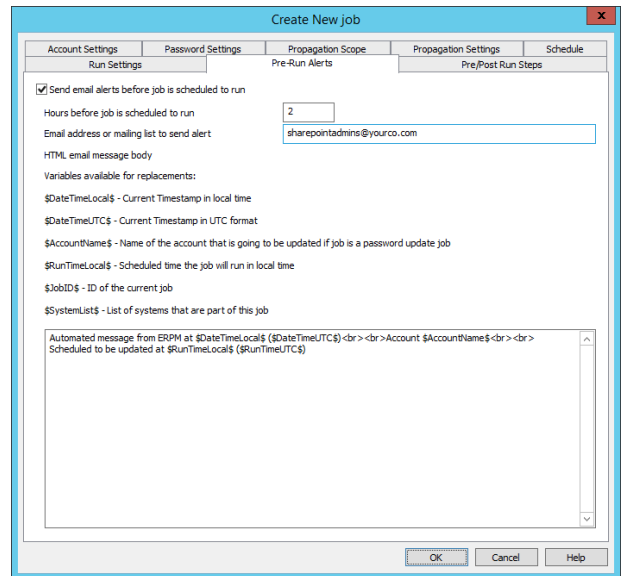
With the option enabled, configure the following settings:

- **Hours before job is scheduled to run:** The number of hours before the job runs that the email will be sent.



**Note:** This requires the deferred processing service to be running and email settings to be configured in order for the alert to be sent.

- **Email address or mailing list to send alert:** Enter either a single email address or a semicolon-delimited list of email addresses to receive the alert.
- **Message:** Enter a custom message to send. Use HTML encoding for any special formatting, such as **<br>** for a line break. There are multiple variables that can be replaced in the custom message:
  - **\$DateTimeLocal\$:** The current local time according to the zone processor when the alert was sent
  - **\$DateTimeUTC\$:** The current local UTC time according to the zone processor when the alert was sent
  - **\$AccountName\$:** The name of the target account if performing a password change job
  - **\$RunTimeLocal\$:** The local time (non-UTC) that the job will be ran
  - **\$JobID\$:** The ID of the job to be run
  - **\$SystemsList\$:** The list of every system that is part of the job



**Create New job**

Account Settings | Password Settings | Propagation Scope | Propagation Settings | Schedule

Run Settings | Pre-Run Alerts | Pre/Post Run Steps

Send email alerts before job is scheduled to run

Hours before job is scheduled to run: 2

Email address or mailing list to send alert: sharepointadmins@yourco.com

HTML email message body

Variables available for replacements:

- \$DateTimeLocal\$ - Current Timestamp in local time
- \$DateTimeUTC\$ - Current Timestamp in UTC format
- \$AccountName\$ - Name of the account that is going to be updated if job is a password update job
- \$RunTimeLocal\$ - Scheduled time the job will run in local time
- \$JobID\$ - ID of the current job
- \$SystemsList\$ - List of systems that are part of this job

Automated message from ERPM at \$DateTimeLocal\$ (\$DateTimeUTC\$) <br> <br> Account \$AccountName\$ <br> <br> Scheduled to be updated at \$RunTimeLocal\$ (\$RunTimeUTC\$)

OK Cancel Help



**Note:** Once the alert is performed, if you re-schedule the job no secondary alert will be sent. The alert flag is reset only after the job runs.

## Configure Propagation Scope

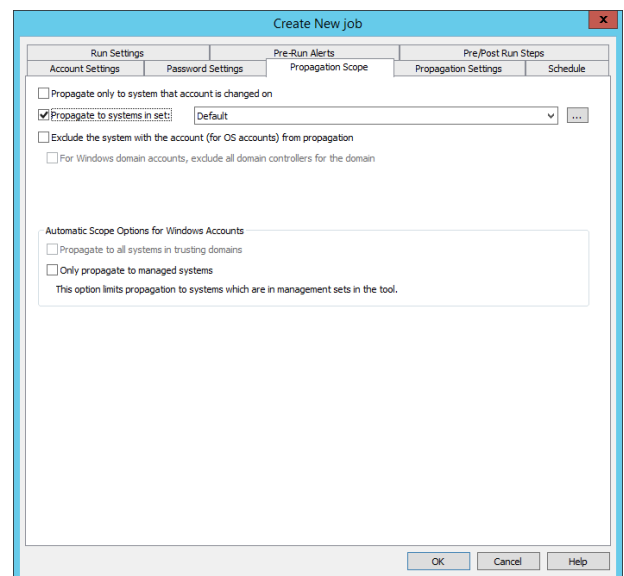
Password propagation is the dissemination of an account's password to all the systems and sub-systems using that account and password. For example, updating a database account password the propagating the newly updated password to the web applications that use that account to connect back to the database.

Propagation Scope is the definition of the systems that will have the new password disseminated to them, such as the web servers in the example above.

By default, all scope options are not enabled. This means no password propagation will occur. To enable propagation, you must define a scope and propagation settings.

Available scope are as follows:

- **Propagate only to system that account is changed on:** Used for a local password change where local resources (i.e. scripts, services, etc.) use the local account.
- **Propagate to system in set:** Used for distributed password changes where the target account exists in one location, such as Active Directory, and is used by services, processes, tasks, etc. on other systems. Use this setting to limit the propagation to a discrete list (management set) of systems. This is the most recommended option.
- **Exclude the system with the account (for OS accounts) from propagation:** Enable this option to exclude examining the target password change system from the propagation scope if it would otherwise be included.
  - For Window domain accounts, exclude all domain controllers for the domain - Enable this option to exclude domain controllers from password propagation operations if they would otherwise be included in the scope.
- **Propagate to all system in trusting domains:** Enabling this option instructs Privileged Identity to enumerate all trusting domains and attempt discovery and propagation on each and every system. This not only requires full access to every system in the forest and other trusting domains, but will incur a huge time penalty depending on the size of your enterprise.
- **Only propagate to managed systems:** Enabling this option effectively sets the propagation scope to every trusting system, so long as it appears in Privileged Identity.



## Configure Propagation Settings

Propagation settings define the sub-system to be scanned for machines defined in the propagation scope. For example, if the list were to contain Windows Services, the password would be updated on the target host, then Windows services would be examined for propagation against every system on the **Propagation Scope** tab.

Use the **Add Existing** button to add any propagation settings that have been previously defined using the **Discovery and Propagation Defaults** dialog as described in the "[Configure Account Discovery and Password Propagation](#)" on page 179 section.

Use the **Add New** button to add new settings for this job.



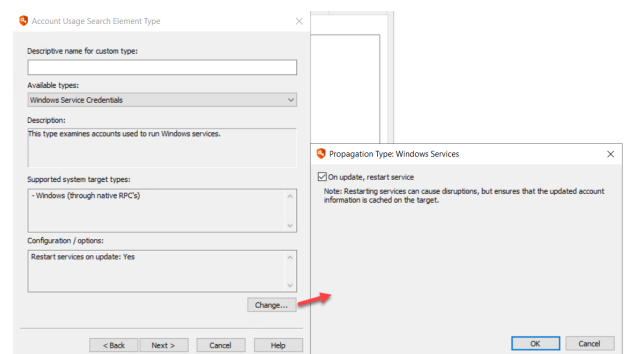
**Note:** These settings can also be saved during this process to the **Discovery and Propagation Defaults** dialog.

Possible target propagations include:

### Windows Services Propagation

- Windows Services:** Affects the identity used to logon for Windows services. Default has no configurations. Windows Services attempts to identify if each Windows service is clustered; if so, various cluster APIs are leveraged. If the service is not clustered, the service control manager (SCM) APIs are leveraged. Each Windows service is stopped and evaluated for dependencies. If any dependent services are found, they are also stopped. Each Windows service and any dependant service is then restarted in the correct order. User-added propagation allows configuration of service auto-restart functionality.

For user defined Windows Services propagation, click the **Change** button to configure that the Windows Service should be restarted following password propagation to the service (default).



**Note:** Windows Services must be restarted to use the new password.

### Windows Scheduled Tasks Propagation

- Windows Scheduler Task RunAs Identities:** Affects credentials used to run Windows Scheduled tasks. There are no configuration options for this propagation type. Use **Settings | Program Options > Performance** to enable a timeout case. The timeout case uses a performance check to determine how long it will take to enumerate the tasks on a target system and if that timeout will be exceeded. If the timeout would be exceeded, this operation will be skipped.
- Windows Scheduler AT Service Account:** Affects the identity used for AT tasks (deprecated after Windows Server 2012). There are no configuration options for this propagation type.

## Windows COM & DCOM Propagation

- **COM Application Identities:** Affects COM Application Identities. There are no configuration options for this propagation type.
- **DCOM Object RunAs identities:** Affects DCOM application RunAs identities. There are no configuration options for this propagation type. Select **Settings > Program Options > Performance** to enable a timeout case. The timeout case uses a performance check to determine how long it will take to enumerate the DCOM applications on a target system and if that timeout will be exceeded. If the timeout would be exceeded, this operation will be skipped.

## Internet Information Services (IIS) Propagation

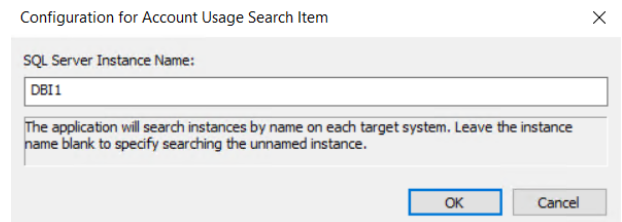
- **IIS6 Metabase Account Information:** For IIS6 (Server 2003), affects users configured to run application pools, configured as the anonymous account for a web site or virtual directory, and the account configured for network access. There are no configuration options for this propagation type. If a target system is detected as Windows Server 2008 or later, this operation is skipped.
- **IIS7 Account Info:** For IIS7 and later (Server 2008 and later), affects users configured to run application pools, configured as the anonymous account for a web site or virtual directory, and the account configured for network access. There are no configuration options for this propagation type. If a target system is detected as Windows Server 2003 or earlier, this operation is skipped.

## SCOM RunAs Accounts Propagation

- **SCOM RunAs Accounts:** Affects credentials configured as RunAs identities within Microsoft System Center Operations Manager (SCOM). Use of this propagation requires copying the correct SCOM SDK binary files (all files from the SDK binaries directory of the SCOM host to the installation directory of the management console and/or zone processor. There are no configuration options for this propagation type.

## Credentials in SQL Server Propagation

- **Credentials in SQL Server:** Lists accounts under the Credentials node as seen in SQL Management Studio. Configure the target database instances to check.



Configuration for Account Usage Search Item

SQL Server Instance Name:  
DB11

The application will search instances by name on each target system. Leave the instance name blank to specify searching the unnamed instance.

OK Cancel

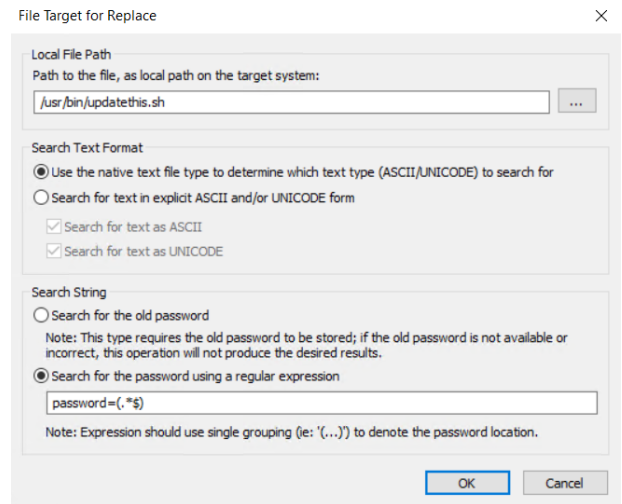
## ASP.NET Connection Strings Propagation

- **Accounts in .NET Config Files:** List accounts configured in the Connection Strings component of ASP.NET under IIS 7 or later. This searches for the following elements: User, UserID and UID. There are no configuration options for this propagation type.



## String Replacements Propagation

- **String Replacements in Files:** For all platforms, configure the path to a file that will be parsed for password replacement using the proper RegEx expression. Add as many string replacement options per job as required. Supply the following information:
  - **Path to the file to update on the target system.**
  - Identify the **Search Text Format** as using the native type or forcing a specific type. Use the specific format option if you know the target file to be a specific format or that it must be updated in a specific format.
  - Identify the **Search String:** It is NOT recommended to use the **Search for the old password** option as this can lead to problems when the username and password match and also requires Privileged Identity to have an accurate accounting of the current password in the file. It is instead recommended to use a proper regular expression to map out the target string to replace with the new password.




**Note:** If there are multiple matches for a given string replacement, they will all be updated.

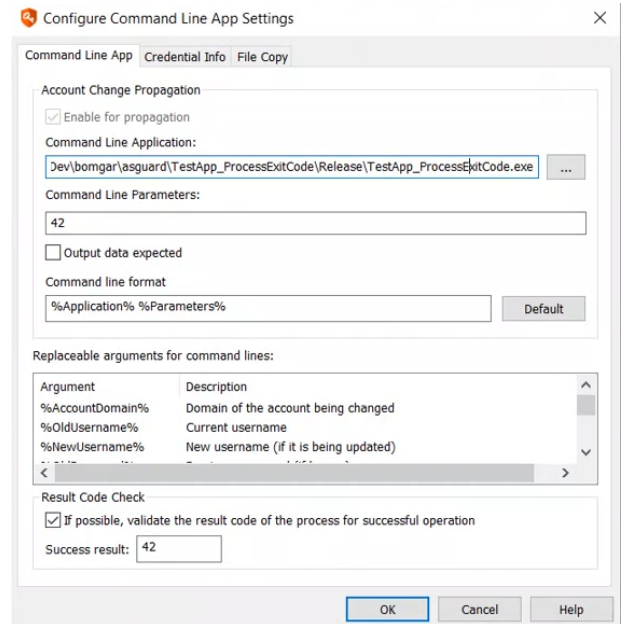
## Run Arbitrary Process Propagation

- **Run Arbitrary Process to Find/Update Credentials:** For all platforms, configure the path to a program or script to run (on the host system or target system). Configuration requires defining if that program will run on the host or target system, who the program will run as, and if any files need to be copied to the target system (Windows only).

## Arbitrary Process Command Line App Tab

Supply the following information:

- **Command Line Application:** The full path of the program to run on the machine that will be running the program.
- **Command Line Parameters:** Any command line parameters for the application. Multiple replacement variables are offered from Privileged Identity such as account names and passwords and system names.
- **Command line format:** Typically this should never be changed. Default value is %Application% %Parameters%. This indicates that the program name and path will be entered first followed by any command line parameters.
- **Result Code Check:** This option is disabled by default. When this option is disabled, the propagation attempts to run the program you specified but does not check the result code for the process to determine if the process ran successfully or not. Enable this option and enter a result code that constitutes a success for the program or script that you are running so that you will know whether or not it ran successfully. The default value is 0 as many Windows programs use that value for a successful exit code.



Argument	Description
%AccountDomain%	Domain of the account being changed
%OldUsername%	Current username
%NewUsername%	New username (if it is being updated)



**Note:** This option currently works only for processes running on Windows target systems.

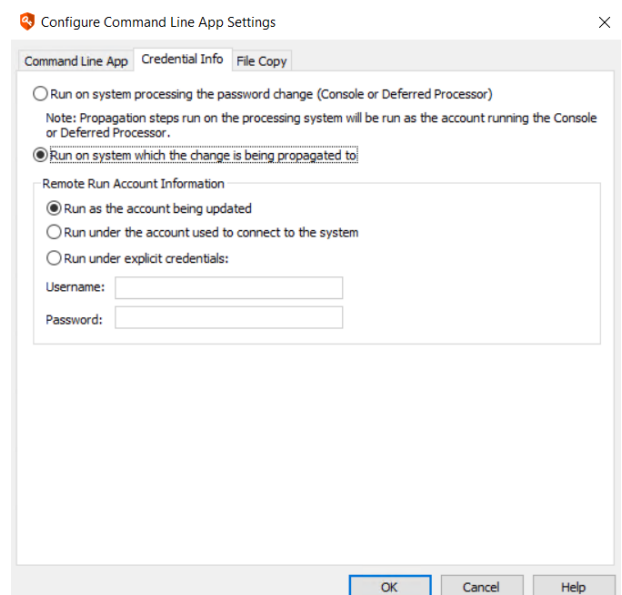
## Arbitrary Process Credential Info Tab

The credential info tab identifies which system will be running the arbitrary process and under what context.

- **Run on system processing the password change:** Indicates the program to run is hosted on a Privileged Identity host and will be run from that system. Programs run in this context will run as the interactive or deferred processing service account.
- **Run on system which the change is being propagated to:** Indicates the program to run is located on the target machine and will be run from the target machine.

When running commands on the remote system, specify the credentials to use when making the connection or running the program. For Windows systems, a scheduled task is created that will run at the next minute. For Linux/Unix systems, an interactive session is created and the process is run in real time:

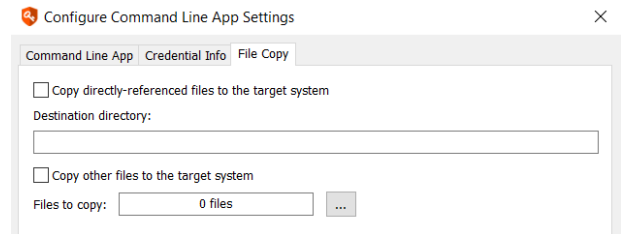
- **Run as the account being updated:** If the target account is admin, the task will be run as admin using the new password.
- **Run under the account used to connect to the system:** For Linux/Unix systems, this is the login account specified on the job.
- **Run under explicit credentials:** User defined credentials to use when running this process.



### Arbitrary Process File Copy Tab

The file copy tab is used to copy files from one location to the target Windows system. This option cannot be used for any platform other than Windows at this time.

- **Copy directly-referenced files to the target system:** Assumes the file on the Command Line App tab exists on the Privileged Identity host at the referenced location and will copy it directory specified in the Destination directory field on this tab.
- **Copy other files to the target system:** Click the ellipses (...) to specify the source files and target location for a file copy to occur then click **Add** to add a file entry (source and destination path).



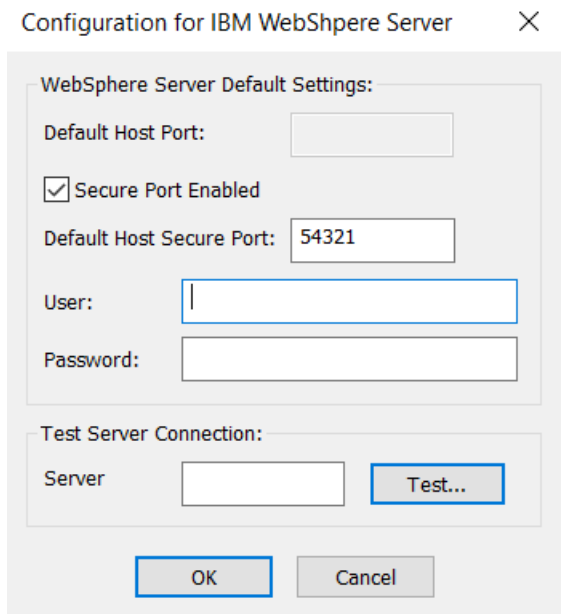
## SharePoint 2007 - Current Propagation

There are no configuration options for the SharePoint propagation type. Use **Settings > Program Options > General** to specify the SharePoint admin port that should be used for management. If you are attempting to manage multiple SharePoint farms, they will all need to be on the same port. This propagation element will deploy a temporary service to the target machine to perform the propagation which will self-terminate and remove itself.

## WebLogic and WebSphere Propagation

- **IBM WebSphere Application Server:** Configure the use of SSL and the target port. This propagation type is typically not used as the functionality has been supplanted by directly managing the WebSphere server using the IBM WebSphere node.
- **Oracle WebLogic Server:** Configure the use of SSL and the target port. This propagation type is typically not used as the functionality has been supplanted by directly managing the WebLogic server using the Oracle WebLogic node.

Click the **Change** button to set the target port and if SSL will be used to connect to the target. Then supply the username and password to connect to the instance with. As desired, supply a server name and click **Test** to test the port and credentials provided.



## SAP Propagation

- **SAP Server:** Configure the use of the Netweaver gateway or direct connect and related information. This propagation type is typically not used as the functionality has been supplanted by directly managing the SAP system using the SAP node.

Click the **Change** button to set the required SAP configurations regarding gateway, system number, client number, etc. Contact your SAP administrator for more information on these settings. The supply the username and password to connect to the instance with. As desired, supply a server name and click Test to test the port and credentials provided.

Configuration for SAP Server
✕

---

**SAP Server Default Settings:**

Is a gateway server?

System No.:

Client:  Destination:

Port:   Use SSL

Path:

---

**Logon:**

User:

Password:

---

**Test Server Connection:**

Server Name:

## Aggregation of Multiple Base Types Propagation

- **Aggregation of multiple base types:** Allows adding of multiple propagation steps in a user defined order.

## Windows Logon Cache Update Propagation

- **Update Logon Cache:** Affects identities stored in the Windows Logon Cache (cached logons). There are no configuration options for this propagation type.

## Local Java Cache for Legacy SDK Propagation

- **Local Cache for Java Client:** Affects the credentials stored in the local Java Cache client (part of the legacy SDK provided with this product) if the local java cache is deployed and running to target systems. There are no configuration options for this propagation type. This specific propagation uses an RMI connection.

## SQL Reporting Services (SSRS) Propagation

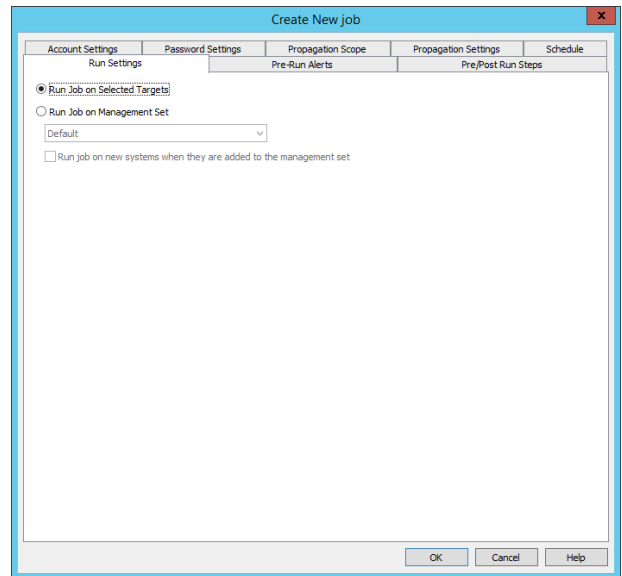
- **SQL Reporting Services:** Affects the account account for the specific SQL Reporting Services instance (SSRS).

## Define Run Settings

The **Run Settings** tab is used to define if this job should target selected systems only, or target a specific management set.

The following are guidelines to help you determine the settings to use:

- **Run Job on Selected Targets:** Use this setting when any of the following are true:
  - You have selected one or more systems from a management and the job should target only those machines and nothing else.
  - You are performing a password change job with propagation settings where the the target account is referenced in one or more locations on other systems. For example, you need to update an account named webaccessacct that exists in a MySQL database but is referenced by many web applications on multiple remote machines.
- **Run Job on Management Set:** Use this setting when any of the following are true:
  - You selected one or more systems from a management set but the job should target every system in the management set and you wish for the job to update itself to target new systems as they are added or remove systems from the job that are removed from the management set. This option is most used for accounts that target, administrator, root, or firecall accounts.
  - You have a job that targets a local account and will propagate to only the local system, not to any other machines.



If the option to **Run Job on Management Set** is enabled, the sub-option to **Run job on new systems when they are added to the management set** becomes available. If this option is enabled, when the management set update runs and new systems are added, those systems will have this job run against them without waiting for the normal job schedule duration to occur. This helps limit the time a machine may be on the network without its accounts being managed.

After the settings are made, and the job created, to edit these settings again, edit the job and adjust the settings listed in the lower right corner of the **Systems** tab.

## Set the Job Schedule

Use the **Schedule** tab to identify when the jobs should run. The available options change based **Job Scheduling Period** selected. For example, if this job will run yearly, you will see options to select the day of the month and month to run the job.



**Note:** All times must be set using a 24 hour clock.

Jobs will run in either the interactive context of the logged on user or that of the deferred processor, depending on the scheduling option selected.

Interactive credentials are used for jobs set to run immediately. Jobs set to run immediately will run as soon as you finish creating the job. Subsequently, if choose to run the job now by right-clicking the job in the management console and selecting **Run Job Interactively**, or by clicking **Run** on the job in the web application, the job will run immediately using interactive credentials.

For any other job schedule type, the deferred processor service account credentials will be used to run the job.

Options for scheduled jobs in the management console include:

- **Immediately:** The job will run directly after saving the job.
- **One time:** Set the Month, day, and minutes into the hour when the job will run. This job will never run again, except interactively. In fact, to use the deferred processor to run this job again, once run, its schedule must be set to a recurring schedule first (such as hourly, daily, weekly, etc.) or it will not reset the flag indicating it has been previously run.
- **Every hour:** Set the minutes into the hour when the job will run.
- **Every day:** Set the hour and minutes into that hour to run. This job will attempt to run every day at that time.
- **Every week:** Set the day of the week, hour of the day, and minutes into that hour to run.
- **Every month:** Set the day of the month, hour, and minutes for the job to run. If a value greater than 28 is used, the job will be scheduled to run on the last day of the target month.
- **Every year:** Set the month, day of the month, hour, and minutes to run the job. If a value greater than 28 is used, the job will be scheduled to run on the last day of the target month.
- **Every N days:** Set the period of days (for example a 60 day interval), hour and minutes for the job to run. The timer will start from today.
- **Interactive Only:** The job will not be scheduled to run. You must manually run the job by right-clicking the job in the management console and selecting **Run Job Interactively**, or by clicking **Run** on the job in the web application.
- **Every N hours:** Set the period of hours (for example, an 8 hour interval), and minutes into that hour for the job to run.

If any of the epoch definitions are in the past, the job will be set to run in the future. For example, if the current time is 12:30 and while configuring an hourly job you set the minutes epoch to 29 minutes, that epoch has passed, thus the job will next run at 13:29. However, if you set the minutes epoch to 31 minutes, the job will next run at 12:31.

Jobs are often time sensitive and businesses have change windows defined, which means changes must occur within a given time period, for example, every Friday night from 12:00AM to 2:00AM. To ensure your job will not run outside of the change window, enable the option

to **Only run job if within time Window** and define the number of minutes where the Window is available. This means a job set to run at 12:30, may be backed up, depending on other jobs, but if it hasn't run within 90 minutes (2:00AM) the job should not run at all and will simply be rescheduled.

## Manage Passwords and SSH Keys

This chapter documents how to configure password change jobs and schedule password changes to happen on an ongoing basis. Privileged Identity can also manage SSH keys that have been discovered and imported into the system. Privileged Identity cannot be used (directly) to disseminate the physical keys it discovers or creates.

### Notes on Managing Passwords

The primary goal of Privileged Identity is to make password changes very easy. The three most common tasks are:

- **Account Discovery:** Enumerate all accounts on target systems supporting discovery.
- **Local account password management:** Change built-in administrator or root account passwords.
- **Domain account password management:** Change service, process, or domain-level fire call account passwords.



**Note:** For password change jobs to occur on a scheduled basis, the deferred processor service (or zone processors) must be installed and correctly configured.

The structure of password change jobs are account-based, rather than system-based. This means it is very easy to change the same account on many systems at once with the same job. This choice also means that changing multiple accounts on the same system will require multiple jobs, one for each specific account. In most cases, after jobs have been created, they will be set to run either once or indefinitely and will not require user interaction.

The steps to configure a password change job are as follows:

1. Select the system(s) to be included in the job.
2. Enter the name of the account to be changed. You can specify the account explicitly by name, choose one of the built-in account types, or select the account.
3. Supply the new password settings. You can set the password for the account on all selected systems to a static value, or you can set it to be generated randomly in compliance with compatibility and complexity settings.
4. Set the schedule for the password update job. The scheduling option dictates whether the job runs once, runs right away, runs at a later time, or runs on an ongoing basis.
5. Configure other job options. There are multiple options you can configure for password change jobs such as account pooling, pre and post run operations, and more.



## Prepare to Manage SSH and Telnet Targets

This section describes the methods used to connect to SSH and Telnet targets so that you can manage passwords on those systems. The management of these systems and devices is controlled by a response file that provides input as **<StdIn>** and parses the output as **<StdOut>**. Systems that use other terminal types such as TN3270 or TN5250 are also supported for management, with or without SSL.

BeyondTrust supports the following encryption methods when using SSH:

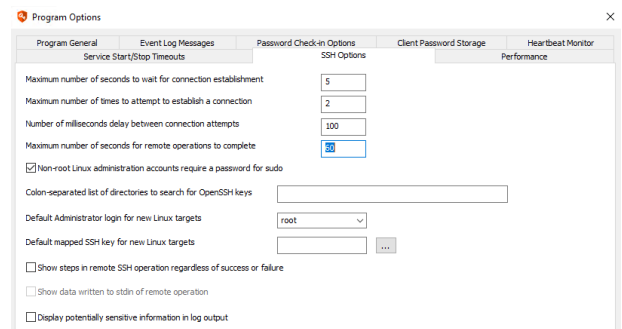
- AES 256 CBC
- AES 192 CBC
- 3DES CBC
- AES 128 CBC
- AES 256 CTR
- AES 192 CTR
- AES 128 CTR

BeyondTrust supports the following HMAC algorithms when using SSH:

- HMAC-SHA1
- HMAC-MD5
- HMAC-SHA1-96
- HMAC-MD5-9
- HMAC-SHA2-256 (additional configuration required)

Other options, such as retry attempts and operation timeouts, are controlled with SSH options located at **Settings > Program Options > SSH Options** or within the answer file used to perform the job. On this dialog, you configure the following options:

- **Maximum number of seconds to wait for connection establishment:** If a connection attempt does not succeed within this time, the connection attempt is considered a failure. The default is 5 seconds.
- **Maximum number of times to attempt to establish a connection:** The number of times a connection attempt is made if there is a failure. The default is 2 attempts.
- **Number of milliseconds delay between connection attempts:** The number of milliseconds to wait between each successive connection attempt. The default is 100ms.
- **Maximum number of seconds for remote operations to complete:** If a job does not complete successfully within this time, the job is considered a failure. The default is 60 seconds.
- **Non-root Linux administration accounts require a password for sudo:** When enabled, if a non-root account performs a refresh against a target Linux/Unix system, it requires the use of a password to run sudo.
- **Colon-separated list of directories to search for OpenSSH keys:** Choose which directories to include when discovering SSH keys on Unix, Linux, and similar systems. If left blank, all directories are searched.
- **Default Administrator login for new Linux targets:** Populates the list with alternate administrators. This list is populated with any administrators found in the database for existing machines, limited to the 20 most commonly found.
- **Default mapped SSH key for new Linux targets:** Choose a default SSH key to use when creating a new Linux target.



## About Response Files

For non-Windows operating systems and devices, XML response files (also called answer files) are used to facilitate connections to Telnet and SSH targets. Response files specify the steps to take when interacting with a target system. Specifically, response files specify the protocol, port, target OS node, and steps to take in the form of input (StdIn) to, and output (StdOut) from, the target system.

The default response file named **response.xml** is located in the installation directory. A large number of additional response files are also provided in the **AnswerFiles** sub-folder of the installation directory. These response files do not address every scenario, however. To make customizing response files easier, a utility called **Remote Command Builder** is provided to help build response files. This utility is available from the **Start** menu on the host system (**BeyondTrust > Remote Command Builder**). Command Builder documentation is also available from the **Start** menu. You do not have to use this utility to build command files; you can copy an existing response file and edit it using your preferred text editor.



**Note:** If you need to modify a response file, create a copy and then modify it. This is important because during an upgrade the original file in the **AnswerFiles** folder will be replaced by the installer.

While the response files do have names indicating what they could be used for, the nodes are not type definitive. If there is a different device or OS type not expressly mentioned, it can likely be added to most any node that supports an SSH or Telnet option or a custom account store may be created. The response file may simply need to be modified to support the platform. The only caveats are that the target must support either SSHv2 or Telnet; SSHv1 is not supported.



**Note:** Response files initially created as real XML files, once run, are copied and stored in the database for later re-use. After being copied to the database, the physical file will not be used. When a response file is used, the database will be checked first. If the answer file is not in the database, the file system is then checked. If the answer file is found in the file system, it uploads a copy of the file to the database. Once a response file is present in the database, the file system version will no longer be used. To make changes to a response file stored in the database, either edit the response file in the response file editor via the management console or delete the version stored in the database, edit the text file, and re-run the job or re-upload the file.

## View and Manage Response Files Stored in the Database

1. Open the console and choose **Settings > Response Files**.
2. Select to add a new answer file, edit an existing answer file, or delete an existing answer file.

## Upload a Modified Response File to the Database

It is often more comfortable for administrators to use their favorite text editor to modify response files due to limited editor-like functionality in the response file dialog editor, such as the ability to collapse a section or search. Similarly, the remote command builder outputs its results to a real file. Shortcuts for the editor are

- CTRL+A to select all text in the editor.
- CTRL+C to copy the selected text in the editor.
- CTRL+V to past the text on your clipboard into the editor.

Once the response file is ready for use, it must be uploaded in the management console.

1. Edit the file-system version.
2. Open the console, and then choose **Settings > Response Files**.

3. Delete the copy stored in the database, if it exists.
4. Add the file-system version, or [re-]upload the response file to the database by running the job. A job auto-uploaded by running a job will be added by its full name, including the file path where the file was originally located.

## Use the Stored Response Files Dialog

- **Edit Schema:** Click to view or make changes to the **Response.xsd** schema file stored in the database (not recommended). To save your changes, click **OK**.
- **Reset All:** Click to reset all response files stored in the database to their original versions.
- **Reset Selected:** Click to reset the selected response file stored in the database to its original version.
- **Add:** Click to create a new response file in the database. In the "Response File" dialog, click **Browse** to locate and load the XML. The file will be added by its simple name.
- **Edit:** Click to edit the selected response file stored in the database. To save your changes, click **OK**.
- **Delete:** Click to remove the current version of a response file from the database. This command does not delete XML files from the file system. If the deleted response file is a file that ships with the solution, it will be replaced with a copy of the file that ships with the solution.

## Understanding Response File Formatting and Syntax

The response file is divided into sections for SSH and Telnet, denoted by the **<SSH>** and **<TELNET>** tags respectively. Within the SSH and Telnet sections are specific commands, denoted by **<Command>**. Under each set of commands are at least three additional items:

- **<Port>**: The SSH or Telnet port to use with this specific command.
- **<TotalTimeout>**: For each step in the job that is NOT an exclude step, specifies the amount of time for the output stream to validate that the required data appears. This option should be adjusted for slow systems or cases where the match output must parse a long stream of output.
- **<UnmatchTimeout>**: For each EXCLUDE step in the job, the amount of time for the output stream to validate that the excluded value does not appear. This option should be adjusted for steps that take longer than others.

Each of the above parameters must be presented in a specific order as determined by the corresponding XSD (XML Schema Definitions) file.

Additional parameters are available. The proper order is:

1. Port
2. TotalTimeout
3. UnmatchTimeout
4. TerminalType
5. HostCodePage
6. SSL
7. SSHAuthModePassword
8. Encryption

The descriptions of the additional values are:

- **TerminalType** is an optional command parameter, used for Telnet only, to control what type of Terminal emulator to use when connecting to the system. Valid options are:

- **Not present:** default Telnet terminal
- **3270:** TN3270 Terminal Emulation.
- **5250:** TN5250 Terminal Emulation.


**IMPORTANT!**

Managing systems using a 3270 or 5250 terminal type also requires separate purchase and installation of Quick3270 or Quick3270 Secure, available from DN-Computing at [www.dn-computing.com](http://www.dn-computing.com).

- **HostCodePage** is an optional command parameter used for Telnet only, to control what character sequences are sent when particular commands or characters are passed to the target system. The HostCodePage for TN3270 emulation should be set to **37**. For a list of valid host code pages, please reference the IBM documentation at: [http://pic.dhe.ibm.com/infocenter/pcomhelp/v6r0/topic/com.ibm.pcomm.doc/reference/pdf/hcp\\_referenceV58.pdf](http://pic.dhe.ibm.com/infocenter/pcomhelp/v6r0/topic/com.ibm.pcomm.doc/reference/pdf/hcp_referenceV58.pdf). Commands are wrapped in greater than and less than symbols. In XML, these are represented as **&gt;** and **&lt;**.
- **SSL** is an optional command parameter, used for Telnet only. If this option is present valid options are:
  - **false:** default
  - **true:** enabled
- **SSHAuthModePassword** is an optional command parameter, used for SSH only, to control what password input method Privileged Identity will use when passing credentials. Valid options are:
  - **0:** Forces keyboard interactive authentication
  - **1:** Forces password-based authentication
  - **2:** default option if SSHAAuthModePassword value is not present. This option will attempt to auto-negotiate keyboard interactive or password-based authentication.
- **Encryption** is an optional command parameter, used for SSH only, used to control what method and in what order Privileged Identity will attempt to authenticate with the SSH target. Encryption values are:
  - **ALL:** AES 256 CBC, AES 192 CBC, 3DES CBC, AES 128 CBC, AES 256 CTR, AES 192 CTR, AES 128 CTR
  - **CBC:** AES 256 CBC, AES 192 CBC, AES 128 CBC, 3DES CBC
  - **CTR:** AES 256 CTR, AES 192 CTR, AES 128 CTR
  - **Legacy:** AES 256 CBC, 3DES CBC

If a specific value is not identified, the command will default to **Legacy**.




**Note:** If a response file is missing the Encryption parameter, the encryption value defaults to Legacy. To support all CBC and CTR ciphers, set the **Encryption** parameter to **ALL** by adding the following line to each command stanza that requires it:

```
<Encryption>ALL</Encryption>
```

The **Encryption** parameter should be the last parameter specified. See the list (above) for the proper order of parameters. Below is a sample Command stanza:

```
<Command>
  <Name>TestSSHConnection</Name>
```



```

<Port>22</Port>
<TotalTimeout>30</TotalTimeout>
<UnmatchTimeout>5</UnmatchTimeout>
<Encryption>ALL</Encryption>
<Parameters>
  <Parameter>
    <Name>DUMMY</Name>
    <Display>DUMMY</Display>
    <Default></Default>
  </Parameter>
</Parameters>
<Execute>
  <Step>
    <StdIn></StdIn>
    <StdOut></StdOut>
  </Step>
</Execute>
</Command>

```

Each series of commands is wrapped in the **<COMMAND>** tag. Within each set of commands is a **<NAME>** for the command and a set of **<PARAMETERS>**. The **<PARAMETERS>** outline the defined variables allowed within the command. The variable are statically named and cannot be changed and more cannot be added. Following the **<PARAMETERS>** section is the **<EXECUTE>** section that identifies the input-to and output-from the system.

The basic premise is **<StdIn>** is your interactive input and **<StdOut>** is the systems output or response to your input. If the expected **<StdOut>** is received, then the step is successful and the process should continue.

For example: If the command entered is **passwd** and the expected output is **Enter the new password**, the input and output step could be defined as:

```

<Step>
  <StdIn>passwd</Stdin>
  <StdOut>new</StdOut>
</Step>

```

Notice that only a partial output match is required. The best practice is to use as little of the StdOut as possible as some systems and network connections do not always buffer the I/O as expected.

In other cases, looking for a success is not possible. For example, when changing a root password in OS X, after a successful change the prompt would simply return to #, rather than indicating success with a success message. This means by looking for a message that indicates success, the job could never successfully complete, even though the password change was successful. In a case like this, a failure message must be sought. In the case of OS X there are error messages that indicate the command was not successful. Such messages include "sorry" or "eof" or "mismatch." In this example, it is better to look for a failure and conclude that if we don't see these messages, then the step is successful.

For example, after re-entering the new password for the root account we want to ensure that we don't see a message that includes "sorry" or "eof" or "mismatch":

```

<Step Exclude="1">
  <StdIn>$NewPwd</StdIn>
  <StdOut>sorry|eof|mismatch</StdOut>
</Step>

```

Notice the opening step has added a new value of **Exclude="1"**. This parameter indicates that the processing should stop if a failure is encountered. Steps including the Exclude flag have a timeout value controlled by the UnmatchTimeout parameter at the top of the command. This means exclude steps will essentially pause for the entire UnmatchTimeout period to wait for the error condition to be revealed before proceeding. That means jobs having exclude steps will take longer to process than jobs that are able to look solely for success indicators.

Also notice that the multiple items separated by the | (pipe) symbol are items that the system should search for in the output. To add greater compatibility to certain target operating systems, Privileged Identity can transmit one character at a time rather than sending over a buffer full of characters. This is done by including **Compat\_SendSingleCharacterAtATime="1"** to any step:

```
<Step Exclude="1" Compat_SendSingleCharacterAtATime="1" >
  <StdIn>$NewPassword</StdIn>
  <StdOut>failed</StdOut>
</Step>
```

Notice that the opening step includes the **Compat\_SendSingleCharacterAtATime="1"** parameter. Multiple parameters can be combined as in the example above, which includes both the **Exclude="1"** and **Compat\_SendSingleCharacterAtATime="1"** parameters.

A single response file can be used for multiple system types. Linux and Unix systems are very similar in most of their basic syntax, but sometimes the verbiage changes slightly. Where some distributions may indicate Password Successfully Changed, others may indicate simply Success or All Tokens Updated. The **<StdOut>** can search for multiple values. The values to search for are separated by the pipe character (|). The pipe is used as an operand meaning or, as in this OR this OR this. If there are multiple possible outputs that should be accounted for, use the pipe. For example:

```
<Step>
  <StdIn>$NewPwd</StdIn>
  <StdOut>Changed|Success|Updated</StdOut>
</Step>
```

Special characters in the expected output need to be escaped by a backslash: \

For example, if the expected output includes :: then each colon must be escaped so we do not treat them as command characters: \::

Special characters are standard regex command characters, such as: \ :: \$ <>()

Other characters perform other commands, such as menu navigation. These are Unicode representations:

- \u000a = line feed
- \u000d = enter
- \u0009 = tab

These can be combined

- \u000d\u000a = enter and line feed, which is equivalent to pressing enter.

## About Response File Sections

This section describes the commands and parameters found in a response file. Any section can be defined for use with any platform, and you can add additional steps provided that the default answer file parameters do not change.

In addition to the parameters included in every response section (command), every section of the answer file now supports a common set of global parameters that represent all possible variables. You may use the default parameters and global parameters at the same time. The following variables are available to all sections of the answer file:

- **\$(TargetAccount)**: The name of the account to be changed by the **AccessUsername**.
- **\$(OldPassword)**: The current password of the **TargetAccount**.
- **\$(NewPassword)**: The new password that will be set for the account being changed.
- **\$(AccessUsername)**: The account being used to log into the system or device.
- **\$(AccessPassword)**: The password for the account being used to log into the system or device.
- **\$(TargetSystem)**: The name as it appears in Privileged Identity for the target system.
- **\$(EnablePassword)**: The enable mode password of the specific Cisco device. This password is associated with an account named **EnableAccount**.
- **\$(UtilityAccount1\_Account)**: The account name of the first account in the Utility Accounts list. Utility accounts are discussed later in this topic. The index number starts at 1 and can be incremented as needed to match the number of utility accounts defined in the password change job.
- **\$(UtilityAccount1\_Password)**: The account password for the first account in the Utility Accounts list. The index number should be incremented to match the index number of the **\$(UtilityAccountn\_Account)** field.



**Note:** For SSH, these parameters are available globally without having to make additional modifications. For Telnet, you must include these parameters in the parameters section of the specific command under Telnet.

In addition to the global parameters, legacy parameters are listed below within each section.

The answer file contains various command sections. These sections correspond to the check boxes that are available in the management console when you create a password change job. Legacy answer file parameters are also noted in these sections. You can use these values in all versions of Privileged Identity.

## The Answer File Sections

- **ChangeRootPwd**: Linux/Unix node password changes. The login account will change its own password and the system will not ask for the current password.
  - Configuration:
    - Update login account
    - Login account is root
  - Default Answer file parameters:
    - **NewPassword**: The password to be set.
- **ChangeLoginPassword**: Linux/Unix node password changes. Login account will change its own password and will be asked for the current password.

- Configuration:
  - Update login account
- Default Answer file parameters:
  - **NewPassword:** The password to be set.
  - **OldPassword:** The current password for the login account.
- **SuChangeRootPassword:** Linux/Unix node password changes. Login account will `su` to another account and that account will change its own password and will not be asked for the current password.
  - Configuration:
    - Changed account is a root level account
  - Default Answer file parameters:
    - **User:** The user to `su` to.
    - **NewPwd:** The password to be set.
    - **OldPwd:** The current password for the login account.
- **SuChangePassword:** Linux/Unix node password changes. Login account will `su` to another account and that account will change its own password and will be asked for the current password.
  - Configuration:
    - No options selected
  - Answer file parameters:
    - **User:** The user to `su` to.
    - **NewPwd:** The password to be set.
    - **OldPwd:** The current password for the login account.
- **RootChangePwd:** Linux/Unix node password changes. Login account is a root account that will change another user account password.
  - Configuration:
    - Login account is root
  - Answer file parameters:
    - **User:** The user to `su` to.
    - **NewPwd:** The password to be set.
- **ChangeCiscoPWD:** Cisco devices node password changes. This section would be used when changing accounts other than the enable account such as VTY or ACS accounts. This presumes the login user does not have enable level privileges and will need to issue the enable command.
  - Configuration: default
- **ChangeCiscoEnablePassword:** Cisco devices node password changes when managing the enable password. Login account will issue the `enable` command and change the password to access enable.



- Configuration: default
- Answer file parameters:
  - **Username:** The user that will issue the `enable` command.
  - **Current password:** Current password for enable.
  - **NewPassword:** The password to be set for enable.
- **AS400:** AS400 node password changes. Login account change its own password and will be asked for the current password.
  - Configuration: default
  - Default Answer file parameters:
    - **Username:** The user that will change its password.
    - **CurrentPassword:** The current password of the login account.
    - **Password:** The password to be set for the login account.
- **OS390:** OS390 node password changes. Login account will change the password of another account and may be asked for the current password of the account being changed.
  - Configuration: default or nothing selected
  - Default Answer file parameters:
    - **LoginName:** The user to login.
    - **LoginPassword:** The password of the login account.
    - **ChangeAccount:** The account being changed.
    - **CurrentPassword:** Password for the account being changed.
    - **NewPassword:** The password to be set for the account being changed.
- **CustomAccountStore:** Custom Account Store nodes can be created. A Custom Account Store is treated as a Linux/Unix node with respect to the sections of the answer file it uses and the options selected.
  - Configuration: default, custom, or nothing selected.
  - Answer File Parameters: Varies, see above.
- **TestSSHConnection:** Used to refresh the system and obtain the list of user accounts during an SSH connection. If the machine has never been managed, it will use the default response file settings. If the system has been managed, it will use the settings from the most recent password change job.
- **TestTelnetConnection:** Used to refresh the system and obtain the list of user accounts during a Telnet connection. If the machine has never been managed, it will use the default response file settings. If the system has been managed, it will use the settings from the most recent password change job.
- **TestTN3270:** Used to refresh the system during a Telnet connection using the TN3270 node in the product. If the machine has never been managed, it will use the default response file settings. If the system has been managed, it will use the settings from the most recent password change job.

## About Utility Accounts

Utility accounts allow you to specify additional accounts as part of an SSH or Telnet password change job. For example, a utility account may be required to use provide alternate username/password when running the Cisco IOS `enable` command when logging in as one account, switching to enable mode, then changing the password of another account.

To add utility accounts to a job, select the accounts in the **Create New Job** dialog. (Utility accounts are specified on the **Account Settings** tab. To access utility account information in a response file, use the following syntax:

- **\$(UtilityAccount1\_Account)**: The account name of the first account in the Utility Accounts list. The index number starts at 1 and can be incremented as needed to match the number of utility accounts defined in the password change job.
- **\$(UtilityAccount1\_Password)**: The account password for the first account in the Utility Accounts list. The index number should be incremented to match the index number of the **\$(UtilityAccountn\_Account)** field.



**Note:** *If a specified stored account is not found in the store at the time of the operation, then both the Account and Password argument will be replaced with 'NULL' when the job runs. Utility accounts are exposed through the web services (including the PowerShell cmdlets) and are preserved correctly when jobs are cloned or when password change jobs that reference existing jobs are created.*

## Connect with SSH Keys

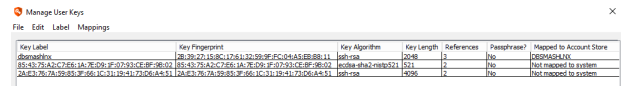
Privileged Identity can use SSH keys to connect to systems or devices added to the **Linux/Unix Systems** node or a custom account store. OpenSSH keys of type RSA are supported for systems management and discovery. RSA keys that are discovered or manually imported are made available for management. Other types of keys, such as DSA and ECDSA, are not presently supported for management. However, all keys can be discovered and reported on. Presently there is no programmatic method to import keys.

SSH keys can be used to connect to systems and devices for the purposes of system, account, and key discovery, and for password rotation and SSH key rotation and updates. Privileged Identity cannot be directly used to disseminate the physical keys it discovers or creates.

### View Keys in the Management Console

To view all user keys in the management console, go to **Settings > Manage User Keys**. This dialog shows keys discovered and keys manually imported.

The **Manage User Keys** dialog defines the following elements:



Key Label	Key Fingerprint	Key Algorithm	Key Length	References	Passphrase	Mapped to Account Store
domain	38:39:2D:A5:8C:17:61:3D:59:9F:FC:14:A5:EB:8B:11	ssh-rsa	2048	3	No	PERMASHLN
	85:23:7A:4C:71:5C:14:7E:59:CF:07:53:CF:8F:8B:03	ssh-rsa	2048	2	No	Not mapped to system
	34:23:7A:7A:59:81:9F:56:1C:31:32:41:73:D6:A5:51	ssh-rsa	2048	2	No	Not mapped to system

- **Key Label:** The label assigned to the key. If not defined by the user during manual import or by renaming, the key from the **SSH Keys** view is set to either the SSH key signature or labeled by system name (key type).
- **Key Fingerprint:** The derived signature of the SSH key.
- **Key Algorithm:** The type of key discovered, for example, SSH-RSA.
- **Key Length:** The length of the key in bits.
- **References:** The number of references across all discovered systems.
- **Age:** This item is automatically hidden. Derived age of the SSH key.
- **Passphrase:** Indicates whether the key has a passphrase associated with it. This is applicable only to private keys.
- **Mapped to Account Store:** Identifies which system, if any, the imported key is associated with.
- **Account Store Type:** The type of system the key is associated with. Presently, only systems or devices added to the **Linux/Unix** node are supported. This item is automatically hidden.
- **Username:** The user name to associate with the key.
- **Private:** Indicates if a private key is found for the referenced key. This item is automatically hidden.
- **Archival Date:** If this key is an archived key, this value indicates the date the SSH key was previously archived. This item is automatically hidden until the view is switched to **Archived Keys**.
- **Old Key Label:** If the key's label has been changed from the key signature, this indicates the previous label. This item is automatically hidden.

At the bottom of the dialog are options for changing the key view:

- **Current Keys:** Sets the view to show only non-archived keys.
- **Archived Keys:** Sets the view to show only archived keys.

The menus offer additional management options for the SSH keys:

- **File**
  - **Import Key:** Imports a new key.
  - **Export Key:** Exports the selected key to a file.

- **Edit**

- **Create New Key:** Creates a new SSH key for Privileged Identity to use when connecting to systems. These keys can potentially be distributed to target systems when mapping the keys to systems from this dialog. Any supported OpenSSH key type can be generated. RSA and EC key types also have selectable key sizes. For RSA, the size can be 2048, 3072 or 4096 bits. For EC, the size can be 256, 384 or 521 bits. An optional label and passphrase can also be provided. If no label is provided, the label is the key fingerprint. PuTTY keys cannot be generated.
- **Remove Key:** Removes the selected key or keys. You are asked to archive the key. The archived keys can be viewed by clicking on the **Archived Keys** radio button at the bottom of the screen. The display is the same as the **Current Keys** view with the addition of one column at the end of the display, **Archival Date**, which is the date and time that the key was placed into the old keys archive. If a key is mapped to multiple hosts, deleting the key removes the mappings for all hosts. Those mappings do not survive in the old keys archive and if the archived key is restored, the mappings are not restored.
- **Restore Archived Key:** Restores the selected key or keys.
- **Toggle All SSH:** Allows the key to be used for any system or device under the **Linux/Unix** node, as opposed to one specific system.



**Note:** If a key is mapped to multiple hosts, deleting the key removes the mappings for all hosts. Those mappings do not survive in the old keys archive and if the archived key is restored, the mappings are not restored.

- **Label**

- **Label Key:** Change the selected key's label.
- **Smart Labels:** Allows a selection of keys or all keys, if none are selected, to be automatically labeled. If user **redimadm** has a 2048 bit RSA key in their `.ssh` directory on host **CENTOS**, the smart label for that key is **redimadm@CENTOS:RSA 2048**. If that same key also exists in the `.ssh` directory for user **root** on host **CENTOS7**, the smart label is **redimadm@CENTOS,root@CENTOS7:RSA 2048**. Any time a conflicting label is generated, **(NUMBER)** is appended to the end of the label, where **(NUMBER)** is **(2)**, **(3)**, etc.
- **Use Fingerprint as Label:** Changes the key label to the key's fingerprint.

- **Mappings**

- **Map to Target:** Maps a key to a target so that it may be used for other management jobs, such as password change, discover, and key rotation.

When clicked, a new window appears with dropdowns to select the desired **Management Set** and **Account store type**. Account stores can be filtered by name, and the number of returned records can be limited. Once the target is selected, enter the **Account store login user name**, then click **OK**.

The window closes, and the mapped target appears in the lower portion of the **Manage User Keys** window.

- **Delete Mapping:** Removes a key mapping.
- **Map to All Targets:** You can select one or more keys, then right-click and click **Map to All Targets**. A new window appears to enter the user login name. Click **OK**, and the list of new mappings appears in the lower portion of the **Manage User Keys** window.

- **Mappings**

- **Map to Target:** Maps a key to a target so that it may be used for other management jobs, such as password change, discovery, and key rotation.

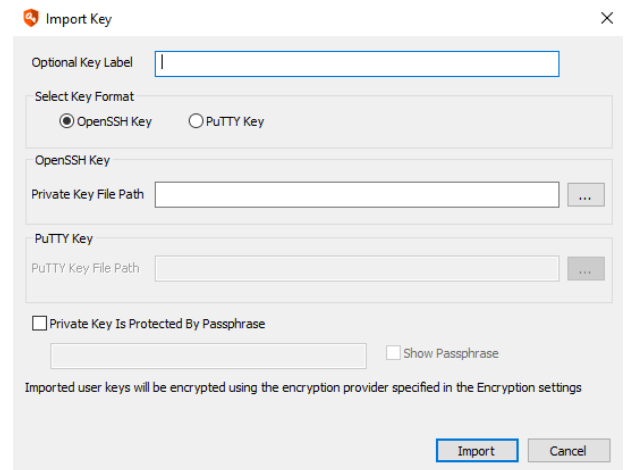
When clicked, a new window appears with dropdowns to select the desired **Management Set** and **Account store type**. Account stores can be filtered by name, and the number of returned records can be limited. Once the target is selected, enter the **Account store login user name**, then click **OK**.

The window closes, and the mapped target appears in the lower portion of the **Manage User Keys** window.

- **Delete Mapping:** Removes a key mapping.
- **Map to All Targets:** You can select one or more keys, then right-click and click **Map to All Targets**. A new window opens to enter the user login name. Click **OK**, and the list of new mappings appears in the lower portion of the **Manage User Keys** window.

## Import a Key Manually

1. Select the **File > Import Key**
2. Supply an optional **Key Label**. If no key label is provided, the fingerprint of the key is used for the key label.
3. Identify whether the imported key is an **OpenSSH Key** or a **PuTTY key**.
4. Provide the path to the private key in the **Private Key File Path** field. Privileged Identity also attempts to derive the public key from the private key file provided during this import process.
5. If a passphrase is required for the key, check the **Private key is protected by passphrase** box and provide the passphrase for the key.
6. Click **Import**.



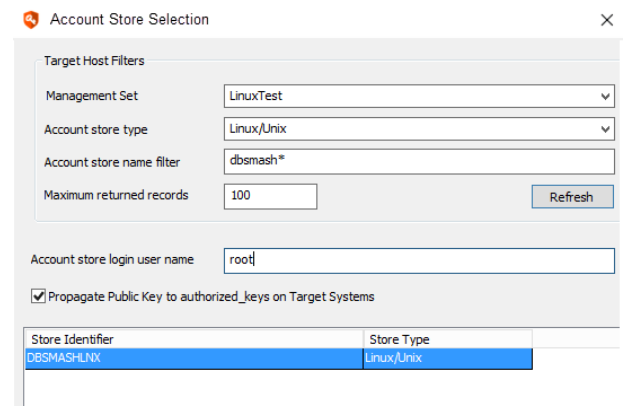
## Map a Key to a Store (System and Account)

Once an SSH key has been added to the store, you can then map the key to an account on one or more systems.



**Note:** EC and ED keys cannot be used for key mapping at this time. As a result, EC and ED keys cannot be used for subsequent operations such as system refresh or password change operations.

1. Select the key.
2. Click **Mappings > Map to Store**.
3. Set the **Account store type** to **Linux**. If necessary, to locate the target system, use the **Account store name filter** and/or the **Maximum returned records** value, then click **Refresh**.
4. Select the target system to map the key to, then click **OK**.

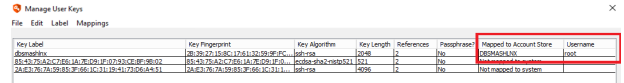


Store Identifier	Store Type
DBSMASH-LNX	Linux/Unix

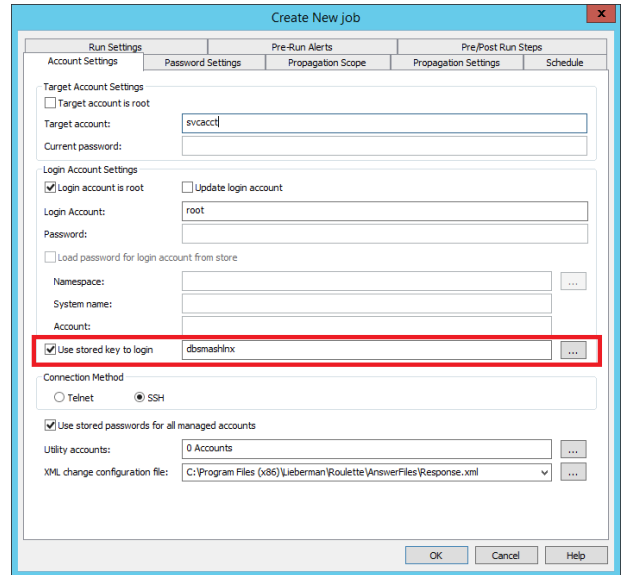
The mapped to account store is then populated with that system name.

Once a key mapping is added, it is automatically attempted for refresh operations. Additionally, jobs targeting systems on the **Linux/Unix** node can also select the mapped key to be used for the login to the target system during a password rotation job.

When configuring a Linux/Unix password management job that uses an SSH key to log in, check the **Use Stored Key to Login** option, then click the ellipses (...) to the right of the key name field. Select the key from the list and click **OK**. Configure the rest of the job as normal.



Key Label	Key Fingerprint	Key Algorithm	Key Length	References	Passwords	Mapped to Account Store	Username
dbsshinx	28:78:23:18:9C:17:61:13:79:4F:FC...	ssh-rsa	2048	1	Yes	dbsshinx	root
dbsshinx	85:45:75:83:C7:86:1A:7E:03:CE:8F:8E:62...	ssh-rsa	2048	2	Yes	dbsshinx	root
dbsshinx	28:78:23:18:9C:17:61:13:79:4F:FC...	ssh-rsa	2048	1	Yes	dbsshinx	root



**Create New job**

Run Settings | Password Settings | Pre-Run Alerts | Pre/Post Run Steps

Target Account Settings

Target account is root:

Target account: svcacct

Current password:

Login Account Settings

Login account is root:  Update login account:

Login Account: root

Password:

Load password for login account from store:

Namespace:  ...

System name:

Account:

Use stored key to login: dbsshinx ...

Connection Method

Telnet:  SSH:

Use stored passwords for all managed accounts:

Utility accounts: 0 Accounts ...

XML change configuration file: C:\Program Files (x86)\Lieberman\ Roulette\AnswerFiles\Response.xml ...

OK Cancel Help

## Remove a Key Mapping

1. Select the target SSH key mapping.
2. Click the **Mapping > Delete Mapping** button.

## Change Passwords and SSH Keys

A job starts when the system(s) are selected or a single account under a particular system is selected and you click Change Password. What each platform requires regarding login information or target account information varies. For example, a Windows host will use Integrated Windows Authentication (typically) while a Linux host will use a login account that may change its own password or another password or switch context to another account during that process using 'su' or 'sudo'.

This section describes how to configure a password change job for each of the various platforms (i.e. nodes) available in Privileged Identity. Specifically, each section describes configuration of the Account tab of a job.

Additional options, such as password constraints and schedule are described in "[Configure Scheduled Job Options](#)" on page 215.

Password change jobs may only target one account at a time. If you select multiple accounts and click the **Change Passwords** button, multiple jobs will be created; one for each account.

## Manage Passwords on Linux, Unix, and Related OSs

This section documents how to configure password and change jobs for Unix, Linux, Solaris, and OSX systems as well as anything that might be added under the Linux/Unix node or a custom account store using SSH or Telnet.

To change Unix, Linux, Solaris, or OSX accounts, supply the name of the account (for example, root), as well as the current password for that account. If the default logon process of the computer has not been modified, changing passwords for many distributions will work out of the box.

**i** If the logon procedure has been customized or the default settings do not work, please see *"About Response Files" on page 246*.

The following system types require extra setup to permit password management via SSH:

- **ESX:** ESX can be managed via SSH or through Native APIs. It is recommended to use the Native APIs. If the ESX host is to be managed via SSH, a stock installation is not ready for management via SSH without special setup. See *"ESX Considerations" on page 267* for more information when managing ESX via SSH. See *"Manage Passwords on VMware ESX" on page 292* to manage ESX using the Native APIs.
- **OSX:** When managing OSX systems, the OSX version becomes a factor. Additionally, the root account is not ready for management without special setup.

**i** For more information, please see *"Manage Passwords on OSX" on page 273*.

## Before You Begin

What you need to know before beginning is what account will login and what account will have its password changed, and which if those two accounts will perform that action. This affects what information should go into which field or check box and potentially, which answer file will be used.

## Example 1

A system will be managed via SSH. There are no stored passwords for any accounts on the system. A low powered account will login. The target account is root. The low powered account will 'su' to the root account (which requires root's current password). Root will then issue the passwd command on itself.

In Example 1, the job will be configured as follows:

- The default response file (response.xml) may be used.
- The login account field will be populated with the the low powered account name.
- The Current password for the login account will be placed in the Password field.
- No check boxes in the Login Account Setting section of the dialog will be enabled.
- The Target account is root check box will be enabled.
- The Target Account field will be populated with the root account name, root.
- The Current password for the root account will be placed in the Current Password field.
- The connection method will be set to SSH.



- The use stored passwords for all managed accounts will be enabled (for future job runs).
- The XML change configuration file may be left at default or set to Response.XML.

## Example 2

A system will be managed via SSH. There are stored for the login account currently. The login account is a low powered account that needs to use 'sudo' to manage the root account's password. The login account is required to enter its password to use sudo.

In Example 2, the job will be configured as follows:

- The response file needs to include the proper sequence for sudo. The job will be configured to use the default **SudoRequirePassword** answer file.
- The login account field will be populated with the low powered account name.
- The default password for the login account will be placed in the Password field. This will be used for any like-configured systems added to the job at a later time.
- Because there is already a stored password for the low powered account that we wish to use enable the check box to **Use stored passwords for all managed accounts**.
- The check box for Login account is root will be enabled, even though the account is not a root account in this case, based on how the answer file is configured.
- No other check boxes will be enabled.
- The connection method will be set to SSH.

## Defined Answer Files

There are many answer files defined for Linux/Unix password change jobs. The list below identifies what ships with Privileged Identity

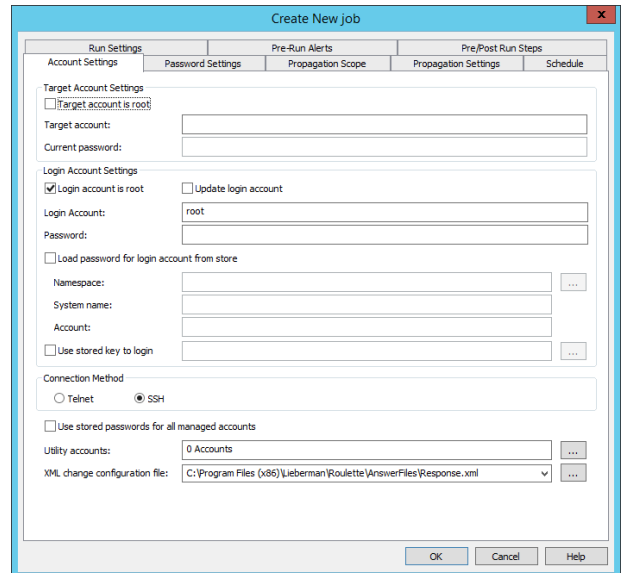
- **Response** - This answer file valid for most password change jobs. It addresses the following scenarios:
  - Low powered login account changing its own password.
  - Low powered login account that will su to another low powered account which will change its own password.
  - Low powered login account that will su to root which will change its own password.
  - Root logging in to change its own password.
  - Root logging in to change another account's password.
  - Also use this file for:
    - **AS400**: "AS400 and OS390 Considerations" on page 264
    - **OS390**: "AS400 and OS390 Considerations" on page 264
    - Solaris
    - Most Linux distributions
    - Most Unix distributions
    - Anything generally Linux-based like ESX via SSH (as opposed to Native APIs)
- **OSX**: The default answer file for OSX password change jobs.
- **AIX**: The default answer file for AIX password change jobs.
- **HPBladeChassis**: The default answer files for HP Blade Chassis where the login account will change its own password.
- **OpenVMS**: The default answer files for OpenVMS servers.

- **Sudo:** Account will login and issue 'sudo passwd' against a target account and will NOT be required to supply a password.
- **SudoRequirePassword:** Account will login and issue 'sudo passwd' against a target account and WILL be required to supply a password.
- **Tandem:** The default answer file for Tandem systems which will use SSH connectivity.

## Create the Password Change Job

In the Account Store view, expand the Linux/Unix node.

1. Select one or more systems or expand one system and select the target account. If selecting the system(s), the Login Account name field will be populated with 'root'. If selecting an individual account, the Target Account field will be populated with that account name and the Login Account field will be populated with 'root'.
2. **Define Login Account Settings:**
  - **Login account is root:** Enable if the login account is a root level account or can simply issue passwd against any account.
  - **Update login account:** Enable if the login account will change its own password.
  - Login Account...
    - **Login Account and Password:** fill in these fields if using an account local to the system.
    - **Load password for login for account from store:** Enable this check box when the system is joined to a directory or an account from another directory/system will be used to login to the system. Then select the Namespace, System name, and account that has already been stored/managed to perform the login. See "[Namespace Values](#)" on page 589 for more information on available namespaces.
    - **Use stored key to login:** Enable this check box to use an SSH key that has been previously imported into the solution and mapped to the target system.
3. **Define Target Account Settings:**
  - **Target account is root:** Enable this check box if one account will login and switch contexts to a different account and this account is root or root level.
  - **Target account:** Supply the name of the target account.
  - **Current Password:** Supply the current password of the target account. This is required when the target account is root and the login account will perform an 'su' to switch to this account which will then change its own password.
4. **Define Connection Method:** Choose between Telnet or SSH (default). If modifying an answer file, this corresponds to the commands in the SSH or Telnet portion of the answer file.
5. **Define Utility accounts:** Not typically used for this type of job. Configure utility accounts when a tertiary account is required to perform additional actions during the transaction. The typical use case is when the process logs in as account one to manage account two but requires a different set of credentials to perform an additional action. Utility accounts are machine specific which means jobs like this are typically also machine specific rather than containing multiple systems. Click the ellipses (...) to browse for utility accounts.



6. **Define XML change configuration file:** This is the answer file to use for the password change jobs. All systems in the job must use the same answer file and same section within the answer file which also implies they will use all the same commands. If two or more systems will require different syntax or different answer files, they must be in a different job (often a different management set), unless you have also configured "[Create Custom Communication Types](#)" on page 219.



*For more information on configuring the additional tabs, please see "[Configure Scheduled Job Options](#)" on page 215.*

## AS400 and OS390 Considerations

AS400 and OS390 systems require special consideration. The default response file has sections pertaining to both AS400 and OS390 that makes certain assumptions, based on provided test servers which may or may not be true in your configurations. For example, the default application name is set to TSO where in your configuration may be DSO. There is also not concept of a login banner which may or may not require additional input.

AS400s often leverage a TN5250 terminal while OS390 systems often leverage a TN3270 terminal

Moreover, you may require the use of a TN3270 or 5250 terminal with SSL or you may have upgraded your system to use SSH with a VT100 terminal. The change in terminal types requires completely different syntax be used in the answer file as well as additional options.

### IMPORTANT!

*Managing systems using a 3270 or 5250 terminal type also requires separate purchase and installation of Quick3270 or Quick3270 Secure, available from DN-Computing at [www.dn-computing.com](http://www.dn-computing.com).*

For example:

An SSH-based answer file will look similar to this:

```
<Command>
  <Name>OS390</Name>
  <TotalTimeout>10</TotalTimeout>
  <UnmatchTimeout>2</UnmatchTimeout>
  <Encryption>ALL</Encryption>
  <Parameters>
    <Parameter>
      <Name>UserName</Name>
      <Display>Username</Display>
      <Default></Default>
    </Parameter>
    <Parameter>
      <Name>CurrentPassword</Name>
      <Display>Current Password</Display>
      <Default></Default>
    </Parameter>
    <Parameter>
      <Name>Password</Name>
      <Display>Password</Display>
      <Default></Default>
    </Parameter>
  </Parameters>
  <Execute>
    <Step>
      <StdIn>\u000a</StdIn>
      <StdOut>Sign On|Username</StdOut>
    </Step>
    <Step>
      <StdIn>$UserName\u0009$CurrentPassword\u000d\u000a</StdIn>
      <StdOut>Press Enter to continue</StdOut>
    </Step>
  </Execute>
```

```

        <StdIn>\u000d\u000a</StdIn>
        <StdOut>Selection or Command</StdOut>
    </Step>
    <Step>
        <StdIn>chgpwd\u000d\u000a</StdIn>
        <StdOut>Change Password</StdOut>
    </Step>
    <Step Exclude="1">
        <StdIn>$CurrentPassword\u0009$Password\u0009$Password\u000d\u000a</StdIn>
        <StdOut>password and verify password not the same|Current password not
correct</StdOut>
    </Step>
</Execute>
</Command>

```

While a 3270 Telnet-based session using SSL on a custom port, performing the same operations will look like this (**note the additional parameters of code page, port, and SSL as well as the unicode characters being replaced with different commands**):

```

<Command>
  <Name>OS390</Name>
  <Port>993</Port>
  <TotalTimeout>10</TotalTimeout>
  <UnmatchTimeout>2</UnmatchTimeout>
  <TerminalType>3270</TerminalType>
  <HostCodePage>37</HostCodePage>
  <SSL>True</SSL>
  <Encryption>ALL</Encryption>
  <Parameters>
    <Parameter>
      <Name>UserName</Name>
      <Display>Username</Display>
      <Default></Default>
    </Parameter>
    <Parameter>
      <Name>CurrentPassword</Name>
      <Display>Current Password</Display>
      <Default></Default>
    </Parameter>
    <Parameter>
      <Name>Password</Name>
      <Display>Password</Display>
      <Default></Default>
    </Parameter>
  </Parameters>
  <Execute>
    <Step>
      <StdIn>&gt;ENTER&lt;</StdIn>
      <StdOut>Sign On|Username</StdOut>
    </Step>
    <Step>
      <StdIn>$UserName&gt;TAB&lt;;$CurrentPassword&gt;ENTER&lt;</StdIn>
      <StdOut>Press Enter to continue</StdOut>
    </Step>
    <Step>
      <StdIn>&gt;ENTER&lt;</StdIn>

```

```
        <StdOut>Selection or Command</StdOut>
    </Step>
    <Step>
        <StdIn>chgpwd&gt;ENTER&lt;</StdIn>
        <StdOut>Change Password</StdOut>
    </Step>
    <Step Exclude="1">
        <StdIn>$CurrentPassword&gt;TAB&lt;;$Password&gt;TAB&lt;;$Password&gt;ENTER&lt;</StdIn>
        <StdOut>password and verify password not the same|Current password not
correct</StdOut>
    </Step>
</Execute>
</Command>
```

## ESX Considerations

This section documents how to configure ESX or ESXi password changes using SSH. These steps are not necessary for managing an ESX host when using Native APIs. In order to change ESX/ESXi accounts, supply the name of the account (for example, root), as well as the current password for that account.

Do not enable Lockdown mode. Doing so blocks all connections with the exception of the vCenter it is joined to and will prevent management of the target host. Only the vCenter Server can manage the host when Lockdown mode is enabled. The port used to connect to these systems for a password change is configured in the response file that is chosen when setting up the password change job. If SSH is configured for a different port, please update your response file accordingly.

The management of the root account will have little to no impact regarding management from vCenter. The reason for this is that when a host is joined to vCenter, vCenter creates a local user account on the host called vpxuser. This user account is what is used to control and manage the host. The vpxuser password is managed automatically by vCenter on a 30 day cycle. The following two scenarios apply to management of ESXi when vCenter is involved:

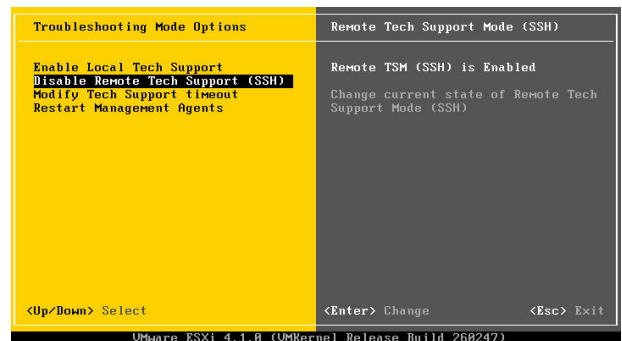
The ESX server is already joined to vCenter: There is no change to your processes. Management of the root account has no effect as it is not used after being joined to vSphere.

The ESX server is not yet joined to vCenter but needs to be managed right away. In this scenario, manage the passwords of the root account. When it is time to join the system to vCenter, simply recover the root password and join the system. Once the system is joined, the root account is no longer required for use by vCenter and can be safely re-randomized (checked in).

## To Enable SSH on an ESX Host

ESX is not configured to support remote SSH out of the box. To enable SSH on an ESX host perform the following steps:

1. For ESXi 4.1+, login to the console select **Troubleshooting Options**. For ESX 4.0 and earlier, go to step 3.
2. Set **Remote Tech Support** to **Enabled**. This configuration will survive a restart.



- For ESXi 3.5 and 4.0, at the console of the ESXi host, press **ALT+F1** to access the console window prior to login and a login prompt will be presented. Enter the password for the root login. Note that there is no prompt when you press **ALT+F1**.

```
Restoring UKernel TCP/IP routes
[2811-02-01 12:02:09 'RoutingInfo' warning] Unable to restore UKernel default g
ateway (192.168.20.1):Unable to set UKernel gateway address. Please verify yo
ur IP settings and try again
Restoring UKernel Resource group settings
Restoring UKernel storage settings
Restoring UKernel NAS settings
/scratch is /vms/volumes/4d47121b-dec845e8-5146-808c29ce123e
/locker is /store
Enabling swap
Starting inetd
Starting crond
Running hostd start
Running slpd start
[1281] Begin 'hostd ++min=0,swap,group=hostd -a /etc/vmware/hostd/config.xml', m
in-uptime = 60, max-quick-failures = 5, max-total-failures = 1000000
Starting slpd
Running sfcdb start
Starting sfcdb
No OEM Policy File exists.
CIM OEM Providers are disabled
Running sfcdb-watchdog start
Running wsmand start
Starting opensmand
_
```

- Enter unsupported in the console and then press Enter. The typed text will not be visible. Simply type **unsupported**, then hit enter. If unsupported was typed correctly, the root login prompt will appear. Enter the password for the root login.

```
Starting inetd
Starting crond
Running hostd start
Running slpd start
[1281] Begin 'hostd ++min=0,swap,group=hostd -a /etc/vmware/hostd/config.xml', m
in-uptime = 60, max-quick-failures = 5, max-total-failures = 1000000
Starting slpd
Running sfcdb start
Starting sfcdb
No OEM Policy File exists.
CIM OEM Providers are disabled
Running sfcdb-watchdog start
Running wsmand start
Starting opensmand
_
You have activated Tech Support Mode.
The time and date of this activation have been sent to the system logs.

WARNING - Tech Support Mode is not supported unless used in
consultation with VMware Tech Support. Tech Support Mode may be
disabled by an administrative user. Disabling requires a reboot of
the system. Please consult the ESX Server 3i Configuration Guide
for important additional information.

Password: _
```

- After entering the root password, the host will display the prompt of **~#**. Edit the file `inetd.conf`. (Enter the command `vi /etc/inetd.conf`.)

```
Starting sfcdb
No OEM Policy File exists.
CIM OEM Providers are disabled
Running sfcdb-watchdog start
Running wsmand start
Starting opensmand
_
You have activated Tech Support Mode.
The time and date of this activation have been sent to the system logs.

WARNING - Tech Support Mode is not supported unless used in
consultation with VMware Tech Support. Tech Support Mode may be
disabled by an administrative user. Disabling requires a reboot of
the system. Please consult the ESX Server 3i Configuration Guide
for important additional information.

Password:
Tech Support Mode successfully accessed.
The time and date of this access have been sent to the system logs.

WARNING - Tech Support Mode is not supported unless used in
consultation with VMware Tech Support.

# vi /etc/inetd.conf_
```

- Find the lines that begins with `#ssh` and remove the `#`. Then save the file. Move the cursor down to each `#ssh` line and then type `dw` to delete the `#` symbol. Then type `:wq` (colon + w + q) to save the file and exit vi. If you make a mistake, press the ESC key and then type `:q!` to quit vi without saving the file.

```
# send the inetd process a HUP signal:
# Do a "ps x" as root and look up the pid of inetd. Then do a
# kill -HUP <pid of inetd>
# inetd will re-read this file whenever it gets that signal.
# <service_name> <sock_type> <proto> <flags> <user> <server_path> <args>
#
#INTERNAL: Internal services
# It is generally considered safer to keep these off.
#echo stream tcp nowait root internal
#echo dgram udp wait root internal
#discard stream tcp nowait root internal
#discard dgram udp wait root internal
#daytime stream tcp nowait root internal
#daytime dgram udp wait root internal
#chargen stream tcp nowait root internal
#chargen dgram udp wait root internal
#time stream tcp nowait root internal
#time dgram udp wait root internal
#
# These are standard services.
#
#ftp stream tcp nowait root /usr/sbin/tcpd in.ftpd
#telnet stream tcp nowait root /sbin/telnetd dropbear ++min=0
#inetd stream tcp nowait root /bin/busybox telnetd
- /etc/inetd.conf 32/77 41%
```



**Note:** there are two lines for SSH with ESXi 4.0 - one for IPv4 (TCP) and the other for IPv6 (TCPv6).

- For ESXi 3.5 prior to update 2, either restart the host or restart the `inetd` process. Restart the management service by typing: `/sbin/services.sh restart`. For ESXi 3.5 update 2 or later and ESXi 4.0, run the command: `ps | grep inetd` to determine the process-ID of `inetd` and then run the command: `kill -HUP <process-id>`.

```
~ # ps | grep inetd
1266 1266 busybox inetd
~ # kill -HUP 1266
~ # _
```



Running this command tells inetd to re-read its configuration file, which was just modified.

8. Type **Exit** to leave unsupported mode.
9. Press **ALT+F2** to return to DCUI mode.

## Manage SSH Keys on Linux, Unix, and Related OSs

Privileged Identity can rotate SSH keys on Linux, Unix, and Unix-like systems, propagate the new SSH keys to other related target systems, and clean up references to old SSH keys. Presently there is no programmatic method to establish SSH Key rotation jobs.

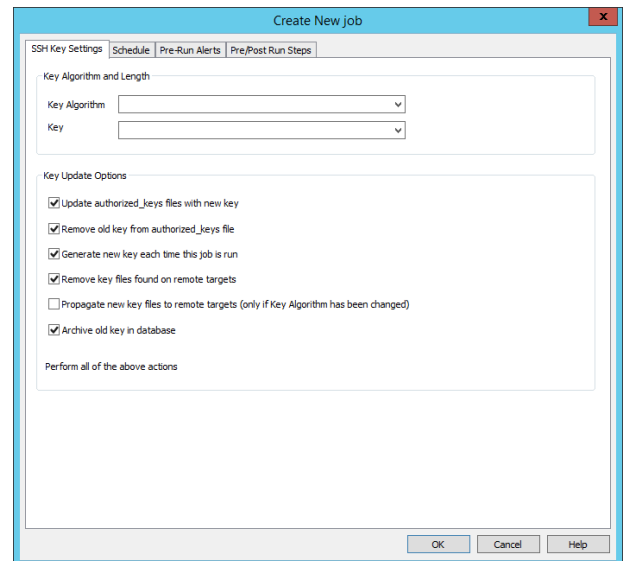
To be able to update a key, the private key must be present and imported into Privileged Identity. Private keys that are automatically discovered can only be imported automatically if they are not also password protected. If the SSH key requires a password, it must be manually imported and the password supplied at that time. Once a private key is imported, the key can be rotated.

The current scope of functionality is designed to take control of SSH keys in such a way as only Privileged Identity has access to the keys. These keys would then be used for application launching or future management connections to the target system.

### Create a Unix/Linux SSH Key Change Job

This topic describes how to complete the configuration tabs in the **Create New Job** dialog for SSH key change jobs.

1. Open the management console and choose **View > SSH Key View**.
2. To create an SSH key change job, select a key, right-click on the target key and choose **Update Key**.
3. Configure the **Key Type** and **Key Length**: Supported values are:
  - RSAv2
    - 2048
    - 3072
    - 4096
  - DSA
    - 1024
  - EC
    - 256
    - 384
    - 521
  - ED25519
    - 256



4. Configure the check box options...

- **Update authorized\_keys files with new key:** Any target that contains the key in an authorized\_keys file will have that file updated.
- **Remove old key from authorized\_keys file after update:** Any target that contains the key in an authorized\_keys file will have that file updated by removing the key from the file.
- **Generate new key each time this job:** Indicates that a new key should be created for that operation. The key generated is specified by the Key Algorithm combo box.
- **Remove key files found on remote targets:** If the algorithm of the new key is different from the key being updated, it indicates that the key files of the old key should be deleted.
- **Propagate new key files to remote targets:** This option is grayed out unless the key algorithm is changed. If the key algorithm is changed, it specifies that the new key files should be copied to the target. The target in this case, is any system

that had a copy of the public or private key file on it. The rules regarding how the new files are named are as such:

- If the standard OpenSSH file name for that algorithm does not exist in the target directory, it will be given that name. For instance, if a DSA key is upgraded to an ED25519 key, when the new key files are copied to the target, if the target does not currently have files named `id_ed25519` and `id_ed25519.pub`, then the new key files will be given those names.
- If there are already files there, as in the previous example, named `id_ed25519` or `id_ed25519.pub`, then the new files will be given the name `RED-IM_<Algorithm>_<Date>_<Time>.<MS>`, such as `RED-IM_ssh-ed25519_11 8 2016_13:22:10.714`.
- **Archive old key in database** will cause the old key to be archived in the database in the event it needs to be retrieved and reinstalled at a later date.
- If all options are unchecked, it represents an option itself, which is Sync existing key to target hosts. This will not generate a new key. It will update all references on target systems with the existing key. For instance, if the `id_rsa` key was deleted from a system, using this option will recreate the file on the target system. It will also verify that the key is contained in `authorized_keys` for any key that was noted as existing in that file for the target system.

The information line at the bottom of the dialog will change depending on what options are selected. As the options are selected and deselected, this line will change. The exception to this are that last two options. The explanations provided indicate what will happen with a variety of selected options in the case of the first 4 options.

The possible explanations that may display on that line and the boxes that are checked to receive that message (top to bottom) are:

- Sync existing key to target hosts. (none checked)
- Remove existing key from `authorized_keys`. (**Remove old key from authorized\_keys file after update**)
- Generate key and store it in the database. (**Generate new key each time this job**)
- Generate key and add to `authorized_keys` leaving existing key in place. (**Update authorized\_keys files with new key, Generate new key each time this job**)
- Generate and store a new key and remove existing key from `authorized_keys`. (**Remove old key from authorized\_keys file after update, Generate new key each time this job**)
- Generate new key, remove existing key from `authorized_keys`, and add new key to `authorized_keys`. (**Update authorized\_keys files with new key, Remove old key from authorized\_keys file after update, Generate new key each time this job**)
- Remove key files from target hosts. (**Remove key files found on remote targets**)
- Remove key files and remove existing key from `authorized_keys`. (**Remove old key from authorized\_keys file after update, Remove key files found on remote targets**)
- Remove key files and generate and store a new key. (**Generate new key each time this job, Remove key files found on remote targets**)
- Remove key files and generate new key and update `authorized_keys` with new key. (**Update authorized\_keys files with new key, Generate new key each time this job, Remove key files found on remote targets**)
- Remove key files and generate and store a new key and remove existing key from `authorized_keys`. (**Remove old key from authorized\_keys file after update, Generate new key each time this job, Remove key files found on remote targets**)
- Generate new key, remove existing key from `authorized_keys`, add new key to `authorized_keys` and remove key files. (**Update authorized\_keys files with new key, Remove old key from authorized\_keys file after update, Generate new key each time this job, Remove key files found on remote targets**)



**Note:** For those options that indicate that `authorized_keys` will be updated with the new key or the existing key will be removed, if the existing key was not contained in the `authorized_keys` file in the first place, it obviously will not be deleted, since it's not there, but it will also not be added to the `authorized_keys` file; this is an update operation only.

5. **Replace key references on systems in the management set:** This defines the list of systems that will be updated with references to the new key.



For more information on configuring the additional tabs, please see "[Configure Scheduled Job Options](#)" on page 215.



### IMPORTANT!

There is presently no support for `su` or `sudo` operations for these key rotation operations. The account configured for connectivity must have the rights to read/write and otherwise modify the target files.



**Note:** For any key that generated by the solution, there is no passphrase on the key. The reason the system does not define a passphrase is because the key is not directly accessible to anything other than the solution. To establish a passphrase on the key, the key must be exported, re-encoded with a passphrase, and then re-imported.

## Manage Passwords on OSX

OSX does not have its own node type and will be added to the Linux/Unix node.

This section outlines how to configure OSX for root-based password changes and account discovery.

### Configure OS X for Password Management

In order to perform administrative functions in OS X, use the **sudo command**. The way sudo is configured out of the box requires the logon user to re-enter their own password for authentication. This means that if someone guesses the password or steals it (and has access to it locally or via SSH), they can take over the computer just as if root had not been enabled. Worse, when executing **sudo -s** to start a root shell, the only thing that shows up in the system.log is this:

```
Mar 20 07:49:12 my-mac-mini sudo: username : TTY=ttyp3 ; PWD=/Users/username ; USER=root ;  
COMMAND=/bin/bash
```

Enabling the root account ensures that users cannot blindly issue sudo commands. The OS X root account, however, is not enabled by default. In order to manage this account, the account must first be enabled.

### Enable the Root Account in OS X From the Command Line

1. SSH into the computer or open a terminal as an administrator.
2. Issue the command **sudo passwd root** and set the root password.

### Enable the Root Account in OS X Using the GUI

Use whichever steps are appropriate for the installed version of OS X.

#### Mac OS X 10.5 or later

1. From the Finder's Go menu, choose Utilities.
2. Open Directory Utility.
3. Click the lock in the Directory Utility window.
4. Enter an administrator account name and password, then click **OK**.
5. Choose Enable Root User from the Edit menu
6. Enter the desired root password in both the Password and Verify fields, then click **OK**.

#### Mac OS X 10.4.x or earlier

1. Click the **Finder** icon in the dock.
2. From the **Go** menu, choose **Applications**.
3. Open the **Utilities** folder.
4. Open the NetInfo Manager utility.
5. Click the lock in the NetInfo Manager window.

6. Enter an administrator account name and password, then click **OK**.
7. For Mac OS X 10.2 and later, choose **Enable Root User** from the **Security** menu.
8. For Mac OS X 10.0 and 10.1, choose **Security** from the **Domain** menu, then **Enable Root User** from the submenu.
9. If a root password has not been previously set, an alert computer may appear that says "NetInfo Error," indicating that the password is blank. Click **OK**.
10. Enter the root password and click **Set**.
11. Enter the password again for verification and click **Verify**.
12. Click the lock again to prevent changes.

Finally, consider that there are three basic levels of user accounts within OS X: users, admins, and root. By default, users do not have the ability to issue sudo commands, but admins can issue sudo commands. If attempting to automate the changing of passwords for the root account using Privileged Identity, and the SSH login account is not a root user, the SSH login account will either need to be an admin (not root), or need to have sudo enabled for the users group on the OS X computer.



**Note:** To enable sudo to "users" of the system edit the `/private/etc/sudoers` file.

## Manage Passwords on Windows

This section discusses how to configure password changes and account discovery for Windows systems. If a domain account is being managed, a domain controller (or domain) must be present in the list and selected for the password change job. If a local account is to be managed, select the local system(s) where the account resides.

Managing Passwords from Windows NT4 through Windows Server 2016, both both workstation and server as well 32 and 64bit is fully supported by Privileged Identity. However, Microsoft Windows may not communicate very well with certain down-level client due to SMB and other security configurations. Please refer to your Microsoft documentation for more information.

It is recommended to run Privileged Identity from the latest operating system as their are interface limitations when running from older operating systems. For example, Windows Server 2008 R2 cannot manage scheduled tasks on Windows Server 2012 or later systems. There are also considerations for clustered services due to lack of forward or backward compatibility of the cluster APIs present in Windows that may necessitate the use of zone processors running on specific operating system versions.

You may manage the following Windows accounts:

- Any local account
- Any domain account
- Active Directory Directory Services Restore Mode Password (DSRM)
- Any service or process account (services, tasks, COM, embedded, hard coded, etc.)
- **Built-in Administrator (RID = 500)**
- **Built-in Guest (RID = 501)**

For Windows, password change jobs for may be initiated from any view. To create a password change job, select one or more systems in the current managed group and click the **Change Passwords** button on the left pane then identify the target Windows Account. If updating a domain account, select only a domain controller for the domain where the account resides; there is no need to select any other system.



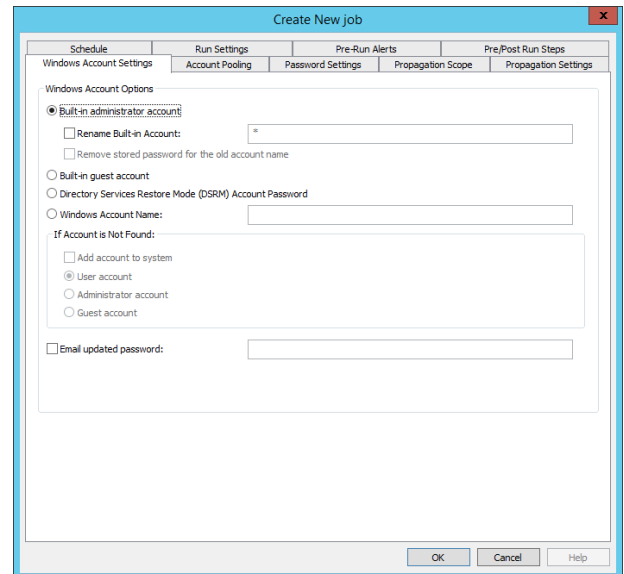
***Tip:** If managing domain accounts, it is recommended to create a separate management set that includes the "domain." Specifically, if the DNS name of the domain is demo.msft, then create a new management set for just this domain object. Using this method, the solution host will use DNS to locate the nearest domain controller to perform a password change job against. As such, accounts will always be associated with the domain rather than a domain controller when they are managed. This has the benefit that the domain controller systems may come and go, but the domain will always persist and the managed accounts will always be available for recovery. If this approach is not used, and a specific DC is targeted, when that DC is retired and removed, the accounts will be removed with it and jobs targeting that DC will simply fail.*

## Create the Password Change Job

1. Open the desired management set containing the target system(s).
2. Go to the preferred view, **Account Store** is recommended.
3. Select one or more systems or expand one system and select the target account. If multiple accounts under a system are selected, you will be prompted that multiple jobs (one for each account) will be created with the same settings.
4. Click **Change Passwords**.

5. Configure the target account on the **Windows Account Settings** tab:

- **Built-in administrator account:** On local systems, member servers, and domains, finds the built-in administrator account. This account's RID is 500. It does not matter if the account has been renamed.
    - If changing the built-in administrative account, the account can be renamed during the update process. To rename the account, select the Rename Built-in Account check box and supply the new name for the account. This feature is handy if the name of the administrator account has been changed on any of the selected systems but the name should be consistent across all systems. This feature is provided as an alternative to what is found in Active Directory Group Policy.
    - If the built-in administrator has been previously managed, and is now being renamed, the old name and old passwords will still be present in the password store. To remove the old password information associated with the old account name, select the option to Remove stored password for the old account name. This option will be unavailable if the account is not also being renamed.
  - **Built-in guest account:** On local systems, member servers, and domains, finds the built-in guest account. This account's RID is 501. It does not matter if the account has been renamed.
  - **Directory Services Restore Mode (DSRM) password:** This is a special password that is created on each and every Active Directory domain controller at install time. If working by hand, the update process would require the use of NTDSUTIL. This process cannot be scripted. Every domain controller's DSRM password is unique to that server and should be changed regularly like any other password. Select the Directory Services Restore Mode Account Password option to change this password.
  - **Windows Account Name:** specify any account by name to update.
    - When updating a specific account by name and that account is not found on one or more of the selected systems, the account can be added to those target systems. To add the missing account to the target systems, enable Add account to system and specify the target type of account (admin , guest, user).
6. As part of the password change job, the password can be emailed to a specified email address. There is no security such as a digital signature associated with this. Simply enable the check box and supply the email address the password should be sent to.



For more information on configuring the additional tabs, please see *"Configure Scheduled Job Options"* on page 215.



## Manage Database Passwords

This section documents how to manage password change jobs on supported databases. Before management can occur, all database types will require a specific database provider be installed. Refer to the installation guide for more information on the specific database target.

### Manage IBM DB2 Passwords

IBM DB2 does not use an internal/explicit account store the way SQL Server, Sybase, MySQL or Oracle do. Rather, DB2 databases leverage the local account store of the host system. This means that to change a password for an account associated with DB2, you need to determine if DB2 is hosted on a Windows, Linux, or Unix platform, and then choose that platform as the target platform when managing DB2 accounts.

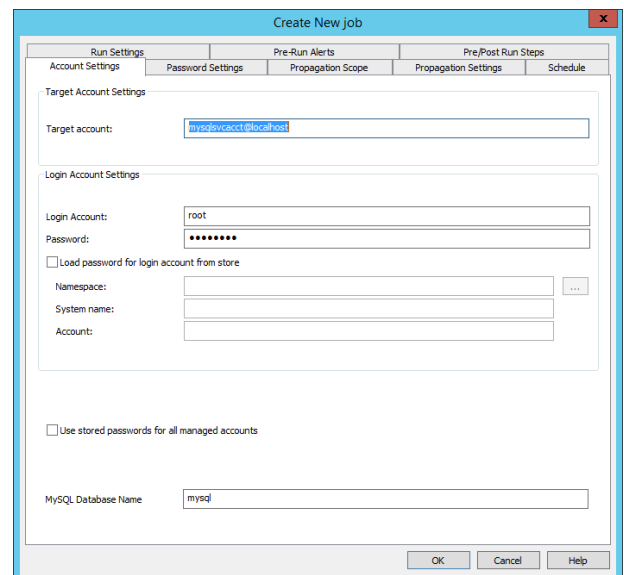
### Manage MySQL and MariaDB Passwords

This section describes how to change passwords for local accounts in a MySQL or MariaDB instance.

### Create the Password Change Job

1. Open the desired management set containing the target system(s).
2. Go to the **Account Store** view.
3. Select one or more systems or expand one system and select the target account.
4. Click **Change Passwords**.
5. Configure the **Account Settings** tab:

- **Target Account Settings:**
  - **Target account:** the name of the target account.
- **Login Account Settings:**
  - **Login Account and Login Password:** Supply the name and password of the login account. If the Login Password field is left blank, the solution will use the stored password for the account if one exists. If no password exists, the job will fail.
  - **Use Stored Password for Login Account:** Use this options instead of the Login Account field to specify a stored/managed account to login to the target database. For example, if a directory account can login to the database, specify the **Namespace**, **System Name** and **Account Name**, as stored in Privileged Identity.
- **Use stored passwords for all managed accounts:** Enable this check box to ensure that stored passwords are used when the job is run. If this check box is not selected and the Login Password field is filled in, the job will use this static password for every job run.
- **MySQL Database Name:** The name of the default database. This field will be pre-populated with the database name configured for the selected instance.



**i** For more information on available namespaces, please see ["Namespace Values" on page 589](#).

**i** For more information on configuring the additional tabs, please see ["Configure Scheduled Job Options" on page 215](#).

## Manage Oracle Database Passwords

This section describes how to change passwords for local accounts in an Oracle database instance.

### Create the Password Change Job

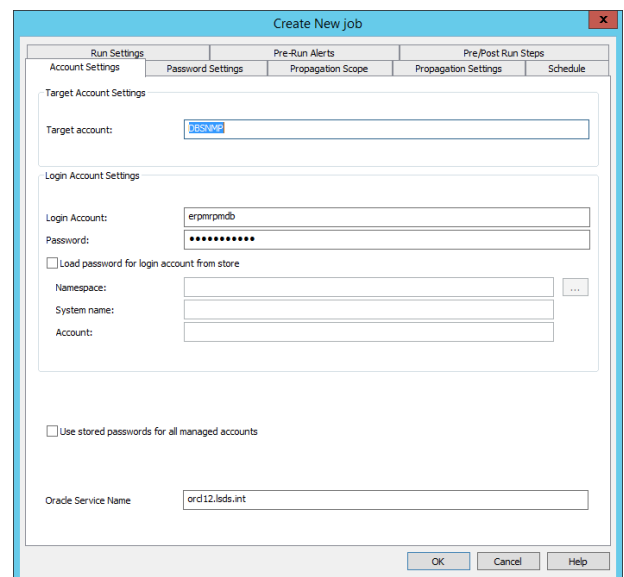
1. Open the desired management set containing the target system(s).
2. Go to the **Account Store** view.
3. Select one or more systems or expand one system and select the target account.
4. Click **Change Passwords**.
5. Configure the **Account Settings** tab:

- **Target Account Settings:**

- **Target account:** the name of the target account.

- **Login Account Settings:**

- **Login Account and Login Password:** Supply the name and password of the login account. If the Login Password field is left blank, the solution will use the stored password for the account if one exists. If no password exists, the job will fail.
- **Use Stored Password for Login Account:** Use this options instead of the Login Account field to specify a stored/managed account to login to the target database. For example, if a directory account can login to the database, specify the **Namespace**, **System Name** and **Account Name**, as stored in Privileged Identity. See ["Namespace Values" on page 589](#) for more information on available namespaces.



- **Use stored passwords for all managed accounts:** Enable this check box to ensure that stored passwords are used when the job is run. If this check box is not selected and the Login Password field is filled in, the job will use this static password for every job run.
- **Oracle Service Name:** The name of the default database. This field will be pre-populated with the database name configured for the selected instance.

**i** For more information on available namespaces, please see ["Namespace Values" on page 589](#).

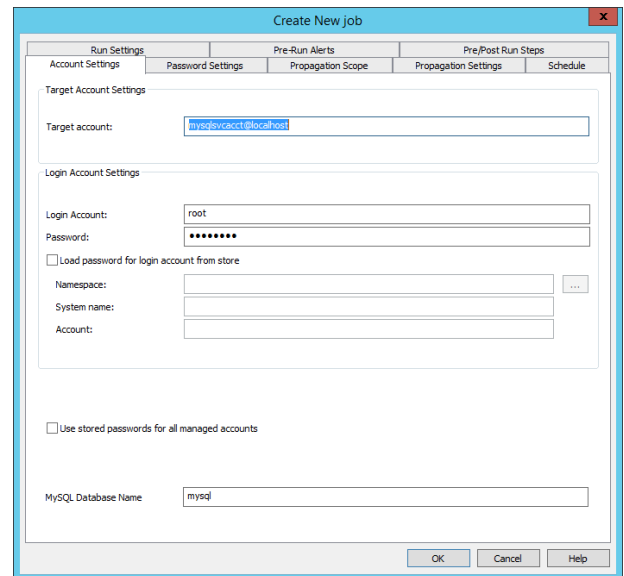
**i** For more information on configuring the additional tabs, please see *"Configure Scheduled Job Options"* on page 215.

## Manage Microsoft SQL Server Passwords

This section describes how to change passwords for local accounts in a Microsoft SQL Server instance.

### Create the Password Change Job

1. Open the desired management set containing the target system(s).
2. Go to the **Account Store** view.
3. Select one or more systems or expand one system and select the target account.
4. Click **Change Passwords**.
5. Configure the **Account Settings** tab:
  - **SQL Server Instance Details:** Specify Default Instance or Named Instance. If using a named instance, the name will be pre-populated with the selected instance. If multiple instances are selected, the field will be populated with "[multiple]".
  - **Use the same connection credentials and connection settings that you specified when you added the SQL Server database to the management set:** If this option is not enabled, you can specify the Login Account and Login Password for a local SQL Account (e.g. SA) that will be used to connect to and manage the SQL instance. If the option is enabled (default), the account store configuration will be used for authentication.
  - **Database Account Name:** The name of the target account.



**i** For more information on available namespaces, please see *"Namespace Values"* on page 589.

**i** For more information on configuring the additional tabs, please see *"Configure Scheduled Job Options"* on page 215.

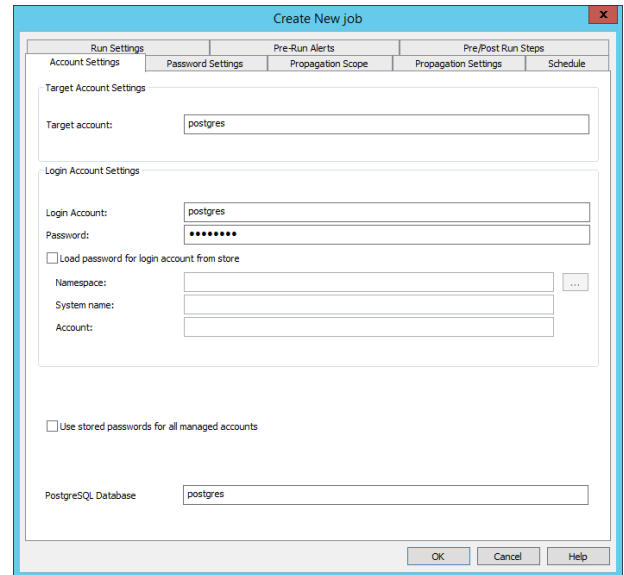
## Manage PostgreSQL Passwords

This section describes how to change passwords for local accounts in a PostgreSQL database instance.

## Create the Password Change Job

1. Open the desired management set containing the target system(s).
2. Go to the **Account Store** view.
3. Select one or more systems or expand one system and select the target account.
4. Click **Change Passwords**.
5. Configure the **Account Settings** tab:

- **Target Account Settings:**
  - **Target account:** the name of the target account.
- **Login Account Settings:**
  - **Login Account and Login Password:** Supply the name and password of the login account. If the Login Password field is left blank, the solution will use the stored password for the account if one exists. If no password exists, the job will fail.
  - **Use Stored Password for Login Account:** Use this options instead of the Login Account field to specify a stored/managed account to login to the target database. For example, if a directory account can login to the database, specify the **Namespace**, **System Name** and **Account Name**, as stored in Privileged Identity. See "[Namespace Values](#)" on [page 589](#) for more information on available namespaces.



- **Use stored passwords for all managed accounts:** Enable this check box to ensure that stored passwords are used when the job is run. If this check box is not selected and the Login Password field is filled in, the job will use this static password for every job run.
- **PostgreSQL Database Name:** The name of the default database. This field will be pre-populated with the database name configured for the selected instance.



For more information on available namespaces, please see "[Namespace Values](#)" on [page 589](#).



For more information on configuring the additional tabs, please see "[Configure Scheduled Job Options](#)" on [page 215](#).

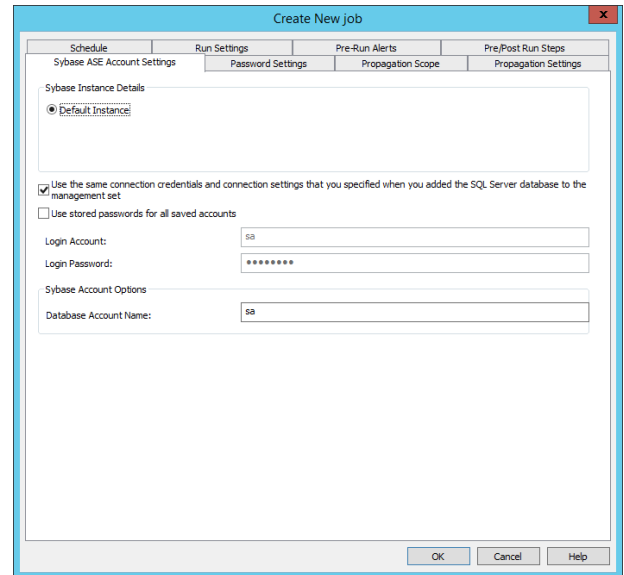
## Manage Sybase ASE Passwords

This section describes how to change passwords for local accounts in a Sybase ASE database instance.

## Create the Password Change Job

1. Open the desired management set containing the target system(s).
2. Go to the **Account Store** view.
3. Select one or more systems or expand one system and select the target account.
4. Click **Change Passwords**.
5. Configure the **Account Settings** tab:

- **Sybase Instance Details:** The default instance is selected.
- **Use the same connection credentials and connection settings that you specified when you added the SQL Server database to the management set:** If this option is not enabled, you can specify the Login Account and Login Password for a local SQL Account (e.g. SA) that will be used to connect to and manage the SQL instance. If the option is enabled (default), the account store configuration will be used for authentication.
- **Use stored passwords for all saved accounts:** Enable this check box to ensure that stored passwords are used when the job is run. If this check box is not selected and the Login Password field is filled in, the job will use this static password for every job run.
- **Login Account and Login Password:** Supply the name and password of the login account. If the Login Password field is left blank, the solution will use the stored password for the account if one exists. If no password exists, the job will fail.
- **Database Account Name:** The name of the target account.




For more information on available namespaces, please see "[Namespace Values](#)" on page 589.



For more information on configuring the additional tabs, please see "[Configure Scheduled Job Options](#)" on page 215.

## Manage Teradata Database Passwords

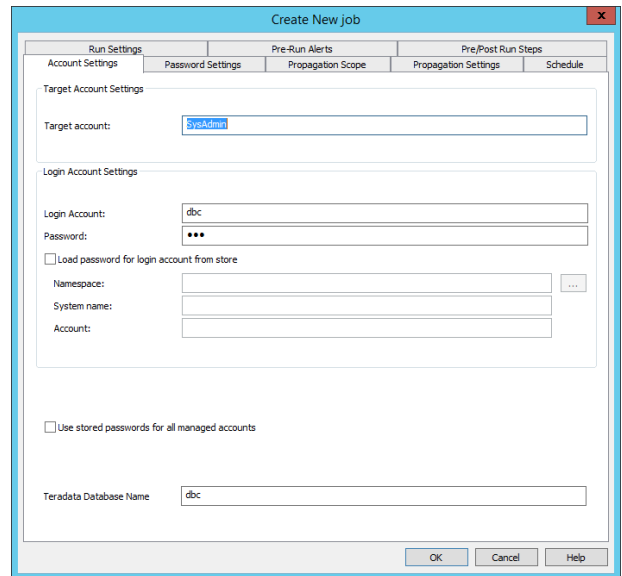
This section describes how to change passwords for local accounts in a Teradata database instance.

### Create the Password Change Job

1. Open the desired management set containing the target system(s).
2. Go to the **Account Store** view.
3. Select one or more systems or expand one system and select the target account.
4. Click **Change Passwords**.

5. Configure the **Account Settings** tab:

- **Target Account Settings:**
  - **Target account:** the name of the target account.
- **Login Account Settings:**
  - **Login Account and Login Password:** Supply the name and password of the login account. If the Login Password field is left blank, the solution will use the stored password for the account if one exists. If no password exists, the job will fail.
  - **Use Stored Password for Login Account:** Use this options instead of the Login Account field to specify a stored/managed account to login to the target database. For example, if a directory account can login to the database, specify the **Namespace**, **System Name** and **Account Name**, as stored in Privileged Identity. See "[Namespace Values](#)" on [page 589](#) for more information on available namespaces.



- **Use stored passwords for all managed accounts:** Enable this check box to ensure that stored passwords are used when the job is run. If this check box is not selected and the Login Password field is filled in, the job will use this static password for every job run.
- **Teradata Database Name:** The name of the default database. This field will be pre-populated with the database name configured for the selected instance.



For more information on available namespaces, please see "[Namespace Values](#)" on [page 589](#).



For more information on configuring the additional tabs, please see "[Configure Scheduled Job Options](#)" on [page 215](#).

## Manage Passwords on LDAP Directories

Privileged Identity can manage passwords in any LDAP-compliant directory. Use the **Account Store View** to expand the appropriate directory node, and then select one or more LDAP instances to begin the process.

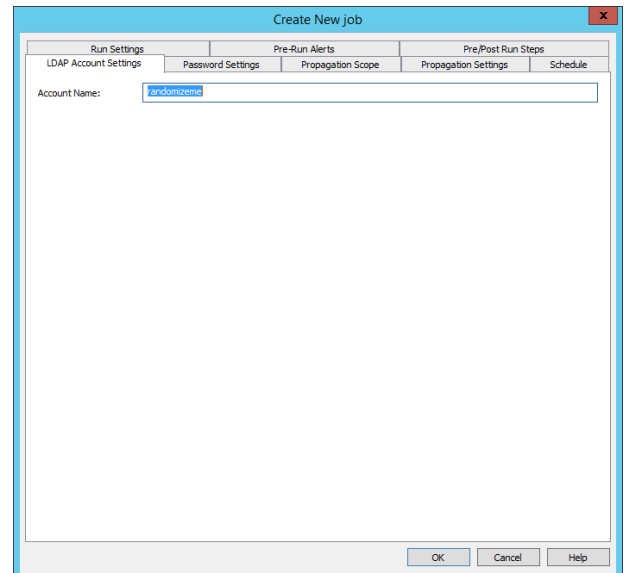
There are nodes for:

- Oracle Internet Directory (OID)
- IBM Tivoli Directory
- Novell eDirectory
- ViewDS Directory

Though the nodes have specific names, you may use these nodes to manage any LDAP-compliant directories.

### Create the Password Change Job

1. Open the desired management set containing the target system(s).
2. Go to the **Account Store View**.
3. Select one or more directories or expand one directory and select the target account.
4. Right-click, and then select **Change Passwords**.
5. Select the **LDAP Account Settings** tab, specify the target account name, and then click **Ok**.



For more information on configuring the additional tabs, please see *"Configure Scheduled Job Options" on page 215.*

## Manage Passwords on McAfee ePO, PeopleSoft, and SAP NetWeaver

Privileged Identity can manage passwords in McAfee ePO, PeopleSoft, and SAP. The target accounts are those that are local to the application rather than anything that might be embedded within a extension or add-on. Choose Account Store View, expand the appropriate directory node, and select one or more instances to begin the process.

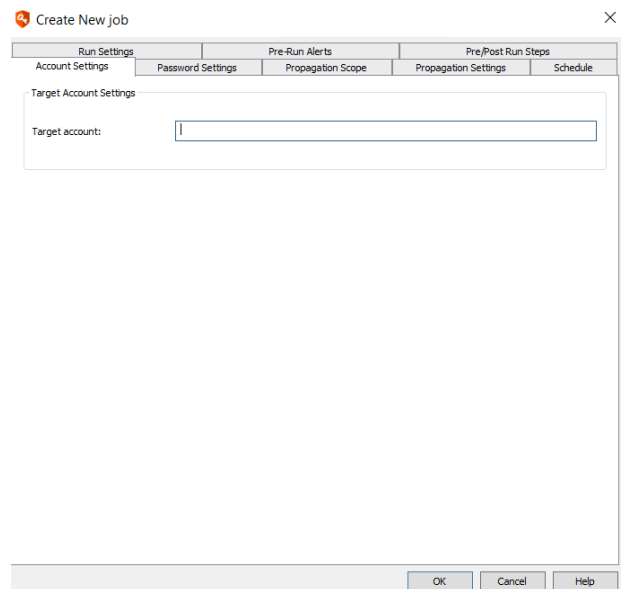
There are nodes for:

- McAfee ePO
- PeopleSoft
- SAP

The process for creating passwords on each of these nodes is identical once the account store has been enrolled.

### Create the Password Change Job

1. Open the desired management set containing the target system(s).
2. Go to the **Account Store View**.
3. Select one or more directories or expand one directory, and then select the target account.
4. Right-click, and then select **Change Password**.
5. Select the **Account Settings** tab, specify the target account name, and then click **OK**.



**i** For more information on configuring the additional tabs, please see *"Configure Scheduled Job Options"* on page 215.



## Manage Passwords on Network Devices

This section covers managing passwords for network devices such as Cisco, IPMI, Xerox, and various SSH/Telnet targets.

### Manage Cisco Node Passwords

This section discusses how to configure password changes for Cisco device accounts and similar devices, switches, and routers.



**Note:** Privileged Identity version 5.4 introduced changes that could affect Cisco IOS and ASA devices. New Cisco job settings are not compatible with legacy jobs. Legacy Cisco jobs that target the enable account will continue to work as they did previously. Do not, however, edit your existing Cisco jobs because the jobs will be overwritten with new settings and break. (Even opening a legacy job and clicking OK without making changes will corrupt the job.) To work around this issue, recreate jobs using the Cisco node.

Privileged Identity can manage the enable password, local VTY accounts, and other local Cisco accounts.

If the default logon process of the device has not been modified, changing passwords will work out of the box for most Cisco based devices and scenarios. However, be sure to review the specific response file before running your job to ensure all required steps are taken such as 'copy run start' and 'wr mem' or the 'password' vs 'secret' parameter when setting a password as not every device requires the same commands.

### Create Jobs That Target the enable Account

Jobs that target the enable account on a Cisco device must use the name EnableAccount in the **Target account** field. (This name is case sensitive.) Using this name correctly also ensures that when your answer file is configured to use **\$(EnableAccount)** to pass the EnableAccount password (to enter config mode), the password is available. In the **XML change configuration file** field, we recommend using the CiscoEnable response file.

- To get to the enable account when the login account is not a priv-15 account, the EnableAccount name/password must be previously managed or imported into the password store before attempting to run these jobs or must be correctly entered into the job.
- If the login account is a priv-15 account, then continue to use your existing [custom] password files or the **CiscoPriv15** response file. We suggest that you select the **Use stored passwords for all saved accounts** option.

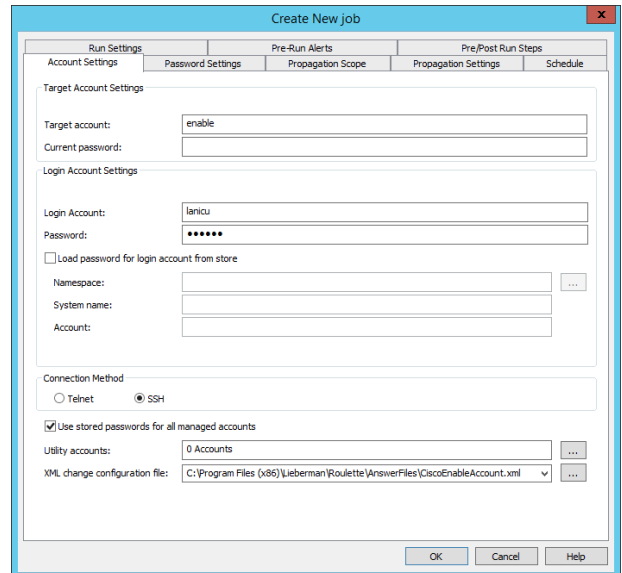
### Create Jobs That Target Local VTY Accounts

Jobs that target the local accounts on a Cisco device should supply the target name of the account in the **Target account** field and ignore the value in the Current Password field because it is not used. In the **XML change configuration file** field, we recommend using the CiscoTTY response file.

- In most cases, the login account will not be a priv-15 level account. As such the job will require the EnableAccount password to actually function. This name/password must be previously managed or imported into the password store before attempting to run these jobs.
- If the login account is a priv-15 account, then continue to use your existing [custom] password files or the **CiscoPriv15** response file. We suggest that you select the **Use stored passwords for all saved accounts** option.

## Create the Password Change Job

1. Open the desired management set containing the target system(s).
2. Go to the **Account Store** view.
3. Select one or more systems or expand one system and select the target account.
4. Click **Change Passwords**.
5. Configure the **Account Settings** tab:
6. Define **Login Account Settings**:
  - **Login Account and Password**: fill in these fields if using an account local to the system.
  - **Load password for login for account from store**: Enable this check box when the system is joined to a directory or an account from another directory/system will be used to login to the system. Then select the Namespace, System name, and account that has already been stored/managed to perform the login.
7. Define **Connection Method Settings**:
  - Choose either **Telnet** or **SSH**.
8. **Use stored passwords for all managed accounts**: Enable this check box to ensure that stored passwords are used when the job is run. If this check box is not selected and the Login Password field is filled in, the job will use this static password for every job run.



**i** For more information on available namespaces, please see "[Namespace Values](#)" on page 589.

**i** For more information on configuring the additional tabs, please see "[Configure Scheduled Job Options](#)" on page 215.

## Managing Passwords for SSH/Telnet Devices Not Under the Cisco Node

While almost any SSH/Telnet target can be added to almost any node, certain devices may be added to specific nodes. Recommendations for these nodes are made earlier in this manual when enrolling devices if you will attempt to use default answer files. Refer to the enrollment sections under "[Enroll New Systems and Devices](#)" on page 62 for more information.

For devices NOT added under the Cisco node, you will follow the steps outlined for devices added to the Linux/Unix node, OS390 node, or Custom Account Store node.

For devices added under the Cisco node, see "[Manage Cisco Node Passwords](#)" on page 285 for more information.

For SSH/Telnet devices added to the Linux/Unix, DRAC or OS390 nodes, see "[Manage SSH Keys on Linux, Unix, and Related OSs](#)" on page 270 for more information.

For DRAC and other Lights Out devices added to the IPMI node (including those mentioned above), see "[Manage Passwords for IPMI Devices](#)" on page 287 for more information.

These devices include SSH Managed Devices:

- Anything added to the Linux/Unix node, including:
  - CheckPoint
  - Cisco ACE
  - Cisco Nexus
  - F5
  - Fortigate
  - Foundry
  - Juniper
  - NetApp
  - PaloAlto
- Anything added to a user create custom account store node.
- Anything added to the DRAC node (as opposed to the IPMI node), including:
  - iDRAC3 via SSH
  - iDRAC4 via SSH
  - iDRAC5 via SSH
  - iDRAC6 via SSH
  - iDRAC7 via SSH
  - Dell CMC via SSH
- Anything added to the OS390 Node, including:
  - HP ProCurve

## Manage Passwords for IPMI Devices

This section describes how to change passwords for local accounts in an IPMI device. IPMI compatible devices include, but are not limited to:

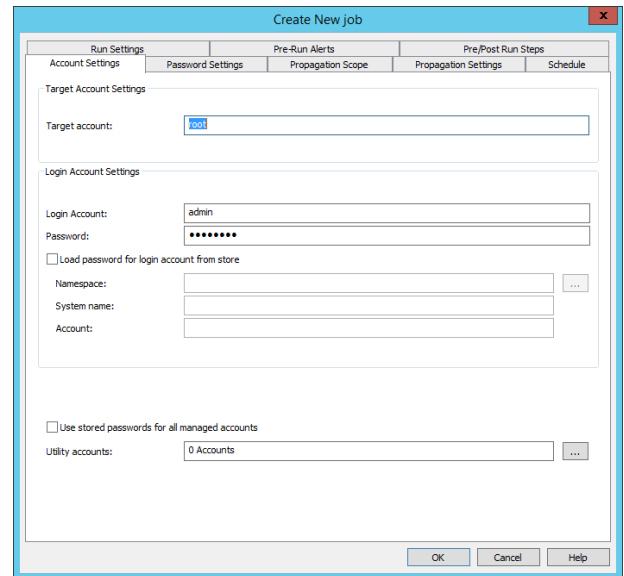
- HP iLO
- Dell DRAC
- SuperMicro IPMI

## Create the Password Change Job

1. Open the desired management set containing the target system(s).
2. Go to the **Account Store** view.
3. Select one or more systems or expand one system and select the target account.
4. Click **Change Passwords**.

5. Configure the **Account Settings** tab:

- **Target Account:** The name of the target account.
- **Login Account...**
  - **Login Account and Password:** fill in these fields if using an account local to the system.
  - **Load password for login for account from store:** Enable this check box when the system is joined to a directory or an account from another directory/system will be used to login to the system. Then select the Namespace, System name, and account that has already been stored/managed to perform the login.
- **Use stored passwords for all managed accounts:** Enable this check box to ensure that stored passwords are used when the job is run. If this check box is not selected and the Login Password field is filled in, the job will use this static password for every job run.
- **Utility accounts:** Utility accounts will not be used for IPMI operations.



6. ID Mgr



For more information on available namespaces, please see "[Namespace Values](#)" on page 589.



For more information on configuring the additional tabs, please see "[Configure Scheduled Job Options](#)" on page 215.

## Manage Password on Xerox Phaser Printers

Privileged Identity can manage passwords the Admin account of a Xerox Phaser printer (note there are no other accounts). Select one or more instances to change the password for.

The process for creating passwords on each of these nodes is identical once the account store has been enrolled.

### Create the Password Change Job

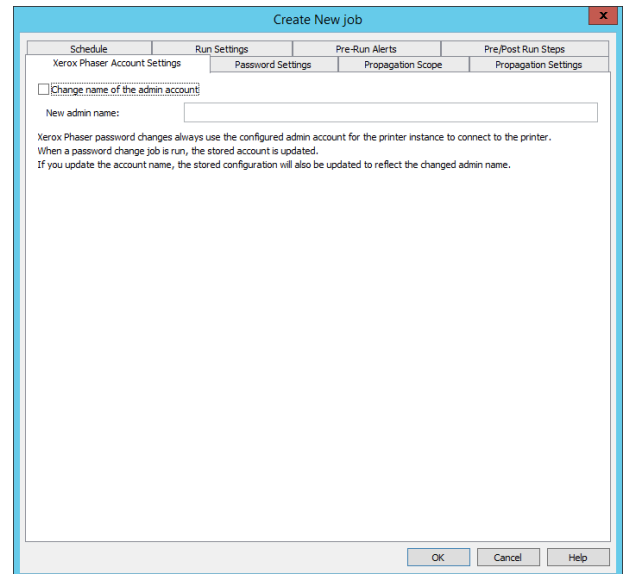
1. Open the desired management set containing the target system(s).
2. Go to the **Account Store** view.
3. Select one or more directories or expand one directory and select the target account.
4. Click **Change Passwords**.

5. Configure the **Xerox Phaser Account Settings** tab:

- **Change name of the admin account:** When enabled will allow specifying of a new name for the admin account in the **New admin name** field. Note there are no other accounts on a Xerox Phaser printer.



For more information on configuring the additional tabs, please see ["Configure Scheduled Job Options" on page 215](#).



The screenshot shows the 'Create New job' dialog box with the 'Xerox Phaser Account Settings' tab selected. The dialog has a title bar with 'Create New job' and a close button. Below the title bar are four tabs: 'Schedule', 'Run Settings', 'Pre-Run Alerts', and 'Pre/Post Run Steps'. Under the 'Run Settings' tab, there are four sub-tabs: 'Xerox Phaser Account Settings', 'Password Settings', 'Propagation Scope', and 'Propagation Settings'. The 'Xerox Phaser Account Settings' sub-tab is active and contains a checkbox labeled 'Change name of the admin account'. Below the checkbox is a text input field labeled 'New admin name:'. Below the input field is a paragraph of text: 'Xerox Phaser password changes always use the configured admin account for the printer instance to connect to the printer. When a password change job is run, the stored account is updated. If you update the account name, the stored configuration will also be updated to reflect the changed admin name.' At the bottom right of the dialog are three buttons: 'OK', 'Cancel', and 'Help'.

## Manage Passwords on WebLogic and WebSphere

Privileged Identity can manage passwords in WebLogic and WebSphere. The target accounts are those that are local to the middleware tier rather than anything that might be embedded within a particular web application. Choose **Account Store View**, expand the appropriate directory node, and select one or more instances to begin the process.

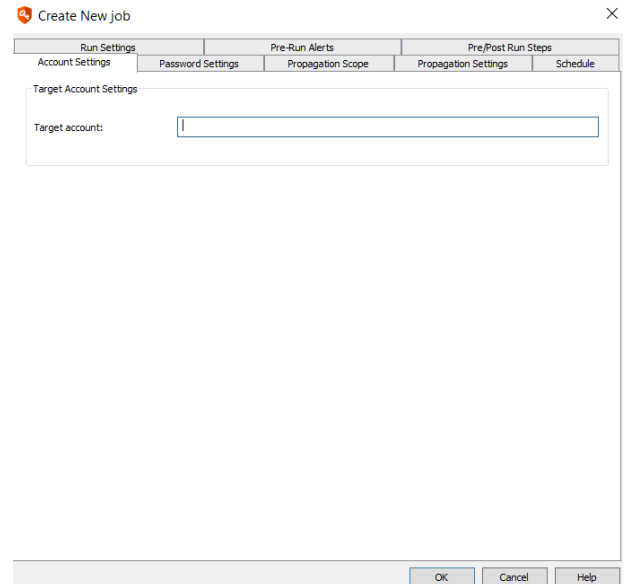
There are nodes for:

- Oracle WebLogic
- IBM WebSphere

The process for creating passwords on each of these nodes is identical once the account store has been enrolled.

### Create the Password Change Job

1. Open the desired management set containing the target system(s).
2. Go to the **Account Store View**.
3. Select one or more directories or expand one directory, and then select the target account.
4. Right-click, and then select **Change Password**.
5. Select the **Account Settings** tab, specify the target account name, and then click **OK**.



**i** For more information on configuring the additional tabs, please see ["Configure Scheduled Job Options"](#) on page 215.

## Manage Cloud Service Provider Passwords

Privileged Identity can manage passwords in various cloud service providers. The target accounts are those that are local to the cloud instance rather than anything that might be embedded within a particular object present in the cloud instance. Choose **Account Store View**, expand the appropriate cloud service node, and select one or more instance to begin the process.

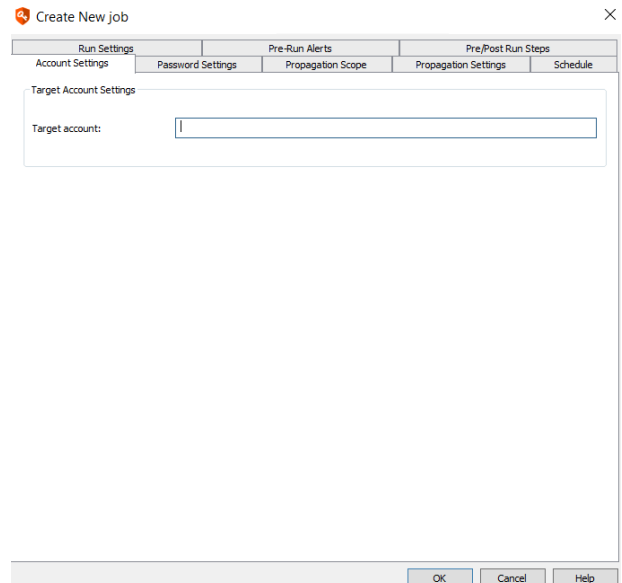
There are nodes for:

- Azure Active Directory
- Amazon Web Services
- RackSpace Public Cloud
- Force.com or SalesForce
- Softlayer

The process for creating passwords and secrets on each of these nodes is identical once the account store has been enrolled.

### Create the Password Change Job

1. Open the desired management set containing the target system(s).
2. Go to the **Account Store View**.
3. Select one or more directories or expand one directory, and then select the target account.
4. Right-click, and then select **Change Password**.
5. Select the **Account Settings** tab, specify the target account name, and then click **OK**.



The screenshot shows a dialog box titled "Create New job" with a close button (X) in the top right corner. The dialog has several tabs: "Run Settings", "Pre-Run Alerts", "Pre/Post Run Steps", "Account Settings", "Password Settings", "Propagation Scope", "Propagation Settings", and "Schedule". The "Account Settings" tab is selected. Below the tabs, there is a section labeled "Target Account Settings" containing a text input field labeled "Target account:". The input field is currently empty. At the bottom of the dialog, there are three buttons: "OK", "Cancel", and "Help".

**i** For more information on configuring the additional tabs, please see *"Configure Scheduled Job Options"* on page 215.

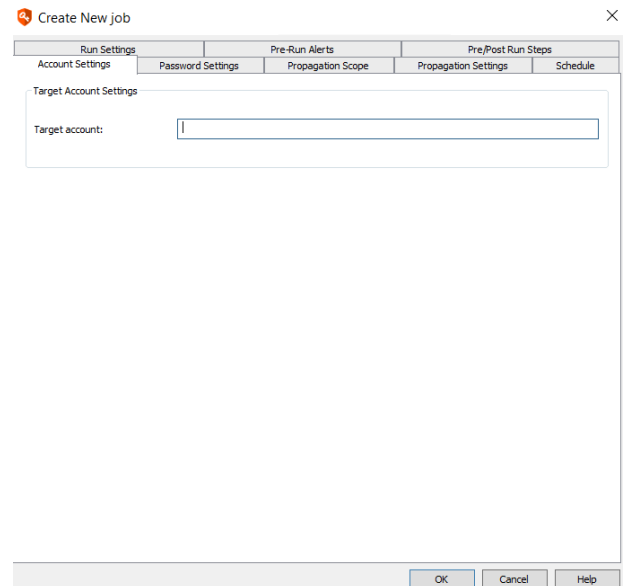
## Manage Passwords on VMware ESX

Privileged Identity can manage passwords in VMware ESX. The target accounts are those that are local to the VM host. Choose **Account Store View** and select one or more instances to begin the process.

### Create the Password Change Job

1. Open the desired management set containing the target system(s).
2. Go to the **Account Store View**.
3. Select one or more directories or expand one directory, and then select the target account.
4. Right-click, and then select **Change Password**.
5. Select the **Account Settings** tab, specify the target account name, and then click **OK**.

**i** For more information on configuring the additional tabs, please see *"Configure Scheduled Job Options"* on page 215.





## Manage Secrets in Key Vaults and Secrets Managers

Privileged Identity can manage secrets stored Azure and HashiCorp key vaults, and AWS Secrets Manager, in much the same way as it manages passwords for other types of account stores. From the Accounts Store View, expand the appropriate key vault or secrets manager node, and then select one or more instance to begin the process.

Nodes may exist for:

- Azure Key Vault
- AWS Secrets Manager
- HashiCorp Vault Enterprise

The process for creating secrets on each of these nodes is similar to creating passwords for any other type of directory or service. The process is also identical for each type of key vault or secrets manager once the providers and directories have been enrolled and the secrets have been refreshed from the directory instances.

### Create the Password Change Job

1. Open the desired management set containing the target system(s).
2. Go to the **Account Store View**.
3. Expand the desired key vault or secrets manager node.
4. Right-click the target secret, and then select **Change Password**.
5. Configure the options on the tabs as required.



*For more information on configuring the additional tabs, please see "Configure Scheduled Job Options" on page 215.*

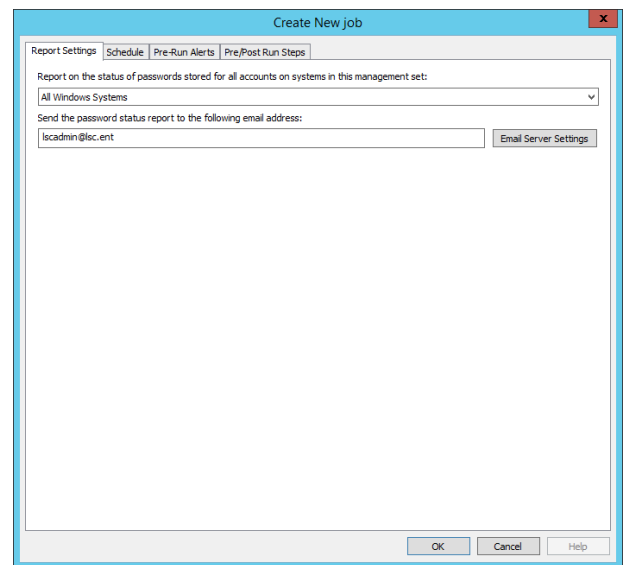
## Verify Stored Passwords

Privileged Identity can verify stored managed passwords for the following platforms:

- Linux/Unix (via direct login or su)
- IPMI
- Microsoft SQL Server
- Microsoft Windows
- Oracle database
- Xerox Phaser

To verify stored passwords, go to **Manage > Verify Stored Managed Passwords**.

Verification reports run against a management set and are sent to an email address with the scan results of that verification report.



For more information on configuring the additional tabs, please see "[Configure Scheduled Job Options](#)" on page 215.

## Create Passwords Lists and Pre-Import Managed Passwords

In addition to managing passwords, Privileged Identity can also store passwords that will not be automatically managed, or that may be used later for managed accounts.

This chapter describes how to import passwords that:

- Will be imported for use with real systems that may be managed or used for management operations.
- Will never be managed and may refer to external locations or non-managed system but must shared among multiple administrators. These are added to Shared Credential Lists or SCLs.
- Are stored for personal use like a personal password wallet. These are added to the personal password store.

## Pre-Import Managed Passwords

Passwords are used in a variety of locations and for a variety of operations in Privileged Identity. This section specifically addresses passwords used for systems and account stores added to a management set. Systems and account stores added to a management set are accessed in the web application on the managed passwords page and leverage the following delegation systems, which are discussed later in this guide:

- Global
- Per Management Set
- Per System
- Per Account
- Compartmentalized Passwords

You can import managed passwords using the management console, PowerShell, Web Service, or web application.



**Note:** When using any of these options, if the targeted account, system, or namespace already exists, it is updated with the newly imported information and the previous password is added to password history, per password history options.

## Import a Managed Password from the Management Console

1. Open the management console and go to **Manage > Import Password Information > Import Password into Password Store**.

2. Fill out the required information:

- **Account type:** The type of account being added: Windows or Linux or Custom. Depending on your choice the **Namespace** field populates with a specific value.
- **System Name:** The name of the system to associate with the password. If the system does not currently appear in any management set with that exact name (or IP), only all access users will be able to see or retrieve the credential.
- **Namespace:** You must supply a value for **Namespace** only if you have choose **OS\_TYPE\_WINDOWS** or **OS\_TYPE\_CUSTOM** as the account type. Expected values for the namespace (when user provided) are:
  - NetBIOS name of the system for a member server, member workstation, or standalone system.
  - NetBIOS name of the domain for a domain object or domain controller.
  - Name of the custom account store surrounded by square brackets. For example, **[VMware (ESX)]**.
- **Account Name:** Name of the target account being imported.
- **Instance Name:** The instance name (named instance, default database, etc.) of the target database. This option is unavailable for anything other than database types.
- **Password:** The password to store and associate with the target account.
- **Password Comment:** (Optional) Any comment for this account that is to be visible in the web site.
- **System Asset Tag:** (Optional) Any asset tag information that is to be stored with this system.

3. Click **Import Account**.

**i** For more information on available namespaces, please see "[Namespace Values](#)" on page 589.

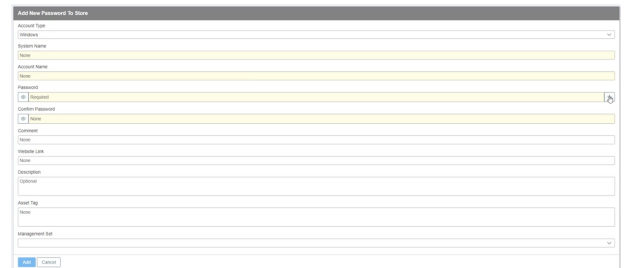
## Add a Managed Password Using the Web Application

You can add passwords in the web application if the logged in identity has any of the following permissions:

- **All Access:** Can do anything in the web application.
- **Add/Edit/Delete Passwords:** Can add, edit, and delete any password from the web site.
- **Add/Edit/Delete Passwords for only Managed Systems:** Can add, edit, and delete passwords for any system appearing in a management set they are already granted access to via global delegations.

To add a managed password:

1. In the web application, go to **Passwords > Managed**.
2. Click the **Add Password (+)** button to the right of the **Filter** field at the top of the page.
3. Fill out the required information:
  - **Account Type:** Select the target account type from the dropdown list of custom account stores and built-in account types.
  - **System Name:** The name of the system as it appears in the solution. Add databases using the format `<systemName\instanceName>`.
  - **Account Name:** The name of the target account.
  - **Password:** The password for the account name.
    - Click the **+** button next to the **Password** field to activate a password generator. The purpose of the password generator is to help you choose a complex password for the account, if desired.
    - Set the desired settings, click **+** to see a **Sample password**, if desired, and then click **Generate**. A new password is generated and placed into the **Password** and **Confirm Password** fields. You may then copy the new password to use in your other web forms.
  - **Comment:** (Optional) Any comment for this account that is to be visible in the web site.
  - **Website Link:** (optional) Any associated URL that is to be shown to the user retrieving the password for this account.
  - **Description:** (optional) Any description for the system that is to be visible in the web site.
  - **Asset Tag:** (optional) Any asset tag information that is to be stored with this system.
  - **Management Set:** (Optional) If you use this option, a list of management sets you can add the system to appear when you click in this field. If the system does not already exist in the target management set, it is added at this time.
4. Click **Add**.



## Import a Managed Password Programmatically

- From PowerShell, call **Set-LSPassword**.
- From SOAP, call **AccountStoreOps\_StoredCredential\_SetPassword**.
- From REST, call **/REST/StoredCredential**.

## Shared Credential Lists

Shared Credential Lists or SCLs, are lists of passwords that will not be managed (i.e. automatically changed) by Privileged Identity. SCLs represent a secure way to store and access credentials that used to be put in spread sheets or on shares.

Using SCLs allows customers to bridge the gap between using an excel spreadsheet or managing passwords by providing a secured and audited portal to access the credentials. When using SCLs, credentials are protected in the same database as managed credentials, and use the same encryption method as managed credentials. Later, when you are ready, you may manage the credential and remove it from the SCL.

When working with Shared Credential Lists, there are three important rules to keep in mind:

- SCLs do not share the same delegation system as managed passwords. This means you must be granted access to a specific list in order to glean any information from it.
- SCLs are mutually exclusive. This means even though an entry for a given system/account may appear in more than one list, updating/editing/deleting of an entry in one list has no effect on that system/account in another list.
- You cannot grant permissions to a single specific account in an SCL as could be done with a managed password. To grant access to a specific account on a specific SCL, the SCL must contain only the single account.

Password lists can be created from either the management console or web interface or PowerShell/web service. They all provide the same options.

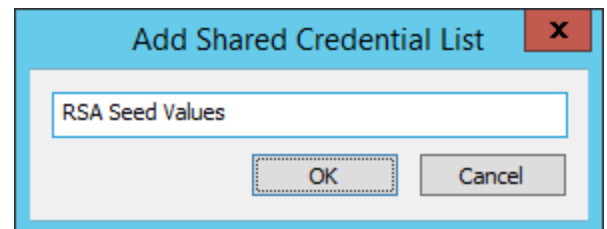
The user who will be able to grant initial permissions is any user who has **All Access** or has the **Web Application Global Delegation** permission for **Manage External Lists**. Permissions may be applied in the web site interface or via the management console. To edit permissions in the management console go to **Delegations > Web Application Shared Credential List Rules**.

To add a new permission to the list, please see "[Manage Shared Credential List Permissions](#)" on page 423.

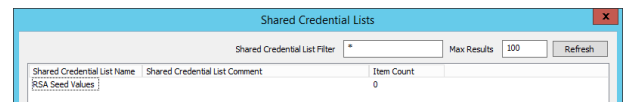
To import or add passwords into an SCL, please see "[Add Credentials to a Shared Credential List](#)" on page 300.

### Create a Shared Credential List from the Management Console

1. From the management console, go to **Manage > Edit Shared Credentials List**.
2. Click the **New** button.
3. Provide a name for the list and click **OK**.



4. Once the list is added, click **Edit** to rename it or add a comment.

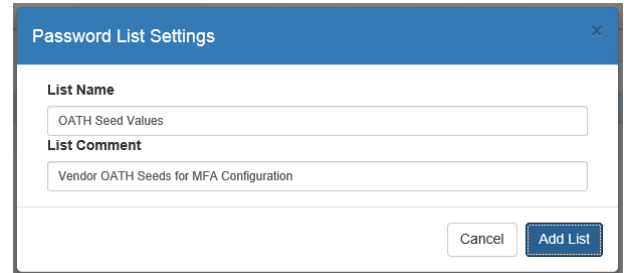


### Create a Shared Credential List from the Web Application

Passwords Lists may be created via the web application provided the identity in question has any of the following permissions:

- **All Access:** Can do anything in the web application.
- **Manage External Lists:** Effectively All Access to all Shared Credential Lists.

1. Go to **Passwords > Shared Lists**.
2. Click the **Add password list (+)** button to the right of the **Filter** field at the top of the page.
3. Provide a name and optional comment for the list and click **Add List**.



4. The list is added to list of SCLs available to you.

## Create a Shared Credential List Programmatically

- From PowerShell, call **New-LSSharedCredentialList**.
- From SOAP, call **AccountStoreOps\_CreateSharedCredentialsList**.
- From REST, call **/REST/SharedCredentialList**.

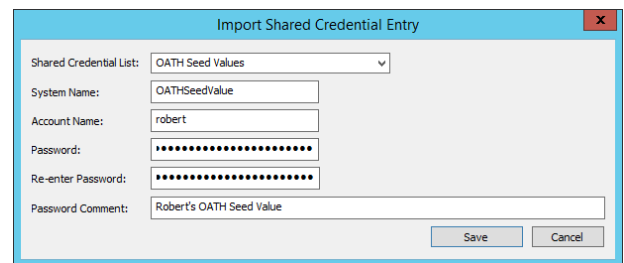
## Add Credentials to a Shared Credential List

Identities with any of the following permissions may add, edit, and delete entries from a shared credential list:

- **All Access:** Provides access to all features in the web application.
- **Manage External Lists:** Provides full access to all shared lists.
- **Add, Edit or Delete Password** on a specific list.

### Import a Shared Credential Using the Management Console

- From the management console, go to **Manage > Import Shared Credential Entry**.
- Select the target list from the **Shared Credential List** dropdown.
- Supply the remaining information:
  - **System Name:** The name as it will appear in system.
  - **Account Name:** The account name (name, email, etc.).
  - **Password:** The password or relevant data for the account name.
  - **Password Comment: Optional:** A comment that will appear next to the account in the SCL.
- Click **Save** to add the entry.



### Add a Shared Credential Using the Web Application

You can add credentials to passwords lists via the web application provided your identity has any of the following permissions:

- **All Access:** Can do anything in the web application.
- **Manage External Lists:** Effectively has All Access to all Shared Credential Lists.
- **Add Password** on a specific list.

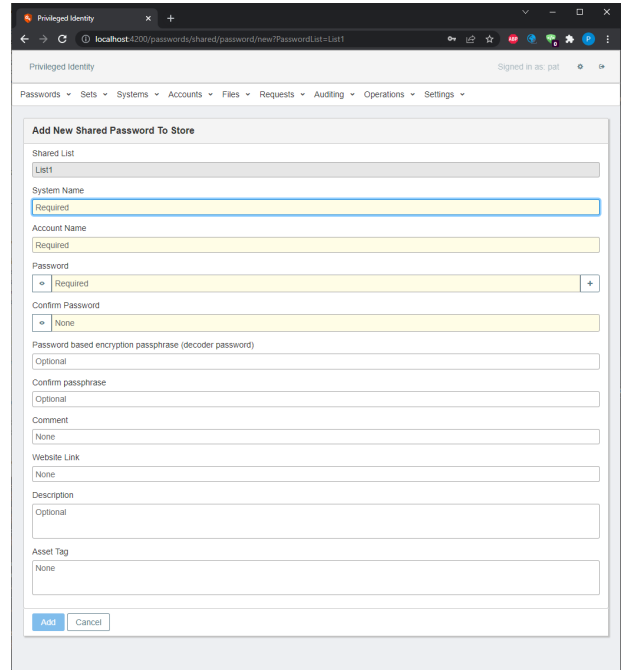
To add a shared password:

1. In the web application, go to **Passwords > Shared Lists**.
2. Set the filter to use the SCL you wish to add the credential to then click the magnifying glass to apply the filter.
3. Click the **Add password** button (+) to the right of the filter at the top of the page.



#### 4. Provide a the following information:

- **System Name:** The name as it will appear in system.
- **Account Name:** The account name (name, email, etc.).
- **Password:** The password for the account name.
  - Click the **+** button next to the **Password** field to activate a password generator. The purpose of the password generator is to help you choose a complex password for the account, if desired.
  - Set the desired settings, click **+** to see a **Sample password**, if desired, and then click **Generate**. A new password is generated and placed into the **Password** and **Confirm Password** fields. You can then copy the new password to put into your other web forms.



• **Password based encryption passphrase (decoder password):** (Optional) The decoder password is a separate password that, if used, must be entered by any user who attempts to retrieve the account password. Providing a decoder passphrase derives a client AES encryption key from that passphrase and then encrypts the password on the client side before sending the data to the server. The data is encrypted on the server again using the configured encryption settings (if enabled). When a password that is protected with a client-provided passphrase is recovered, the password is returned from the server still encrypted with the AES key that was generated on the client side. The same passphrase needs to be provided to derive the same AES key to decrypt the password after recovery. This is done so the encryption key is never transmitted, stored, or known on any system other than the client system, protecting the password from being accessed by system and database administrators who have access to the application's encryption key and settings.

- **Confirm passphrase:** (optional) Confirm the decoder password entered above.
- **Comment:** (Optional) A comment that appears next to the account in the SCL.
- **Website Link:** (Optional) Any URL associated with the system or account. This is visible when the password is retrieved.
- **Description:** (Optional) Any additional description for the account.
- **Asset Tag:** (Optional) Add Asset Tag for passwords, systems, and/or shared credentials.

#### 5. Click **Add**.

## Import a Shared Credential Programmatically

- From PowerShell, call **New-LSSharedCredential**.
- From SOAP, call **AccountStoreOps\_SharedCredential\_SetPassword**.
- From REST, call **/REST/SharedCredential**.

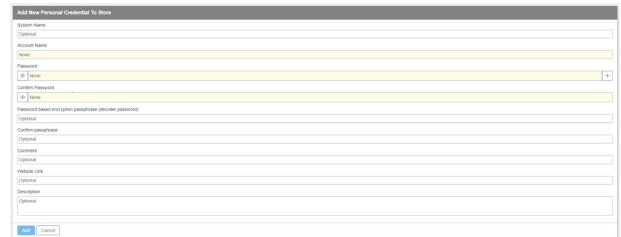
## Personal Password Stores

When the personal password store is enabled in the web application configuration, any user that has the ability to log in to the web site can add their own personal passwords to the password store. Users can share passwords with other users.

### Add a Password to Your Personal Password Store

1. In the web application, go to **Passwords > Personal**.
2. Click the **Add Password (+)** button to the right of the **Filter** field at the top of the page.
3. Fill in the required information:

- **System Name:** (Optional) A system name for the personal password.
- **Account Name:** The login account name.
- **Password:** The password for the account name.
  - Click the **+** button next to the **Password** field to activate a password generator. The purpose of the password generator is to help you choose a complex password for the account, if desired.
  - Set the desired settings, click **+** to see a **Sample password**, if desired, and then click **Generate**. A new password is generated and placed into the **Password** and **Confirm Password** fields. You can then copy the new password to put into your other web forms.
- **Password based encryption passphrase (decoder password):** (Optional) The decoder password is a separate password that, if used, must be entered by any user who attempts to retrieve the account password. Providing a decoder passphrase derives a client AES encryption key from that passphrase and then encrypts the password on the client side before sending the data to the server. The data is encrypted on the server again using the configured encryption settings (if enabled). When a password that is protected with a client-provided passphrase is recovered, the password is returned from the server still encrypted with the AES key that was generated on the client side. The same passphrase needs to be provided to derive the same AES key to decrypt the password after recovery. This is done so the encryption key is never transmitted, stored, or known on any system other than the client system, protecting the password from being accessed by system and database administrators who have access to the application's encryption key and settings.
- **Confirm passphrase:** (Optional) Confirm the decoder password entered above.
- **Comment:** (Optional) Any comment to help you identify what the account is used for.
- **Website Link:** (Optional) A URL associated with this account, visible next to the account in the Personal Password Store.
- **Description:** (Optional) Any other relevant information about this account.



4. Click **Add**.

### Configure Additional Options Within the Personal Password Store

In addition to storing passwords, you can recover the password, edit the password attributes, edit details, view the account history, and delete the password entry. You can access these options by clicking the ellipsis button for the specific credential.

You can sort the columns in your personal password vault by clicking on the column headers. The **System Name** and **Account Name** headings sort alphabetically (A-Z or Z-A). **Last Change Time** sorts from oldest to newest and vice versa.

- **Recover Password:** In the **Password** field in the **Password Recovery** dialog, click the eye icon to make the current password visible.



**Note:** *If a decoder password has been set for these credentials, you must provide the decoder password before you can recover the credential password.*

- **Edit Password Attributes:** Click the pencil icon to edit the password, as well as the **Decoder Password** and **Description** fields.
- **Edit Details:** Click the icon of the pencil inside a box to modify the **System Name**, **Account Name**, **Comment**, **Website Link**, or **Description** fields.
- **View the Account History:** Click the bullet list icon to view the password history for the credential. For each item in the history list, click the eye icon to reveal the password.



**Note:** *The Account History option does not appear on an account entry until the password for that account has been changed at least once.*

- **Delete the Password Entry:** You receive a prompt to inform you that the change will be permanent, and to click **Yes** to continue, or **Cancel**, to back out.

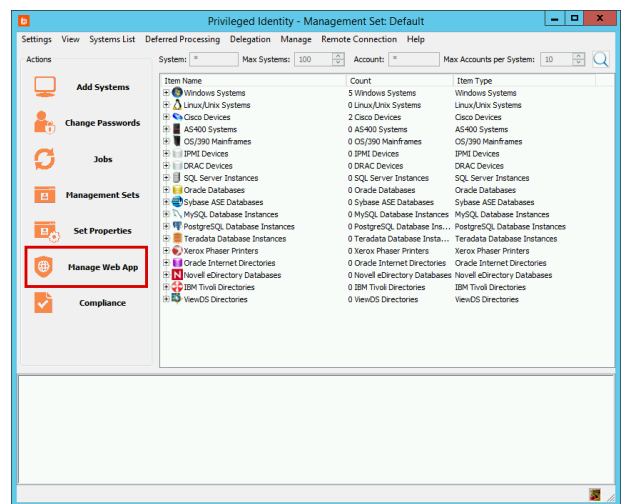
# App-to-App Password Management

With app-to-app password management, you can install a host-based agent on Windows endpoints to enable embedded application authentication. The app-to-app feature also lets you enforce attributes of the calling application, such as its full path, a matching SHA256 hash, and the authorization of the user executing the application.

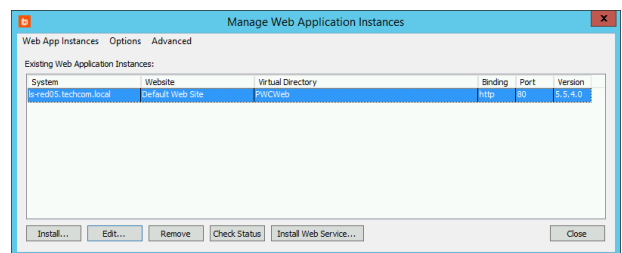
Developers can securely embed credentials into compiled applications subject to compliance mandates for rotation. Privileged Identity administrators can further lock down these applications, leveraging one or all of the attributes listed above.

## Enable App-to-App Password Management

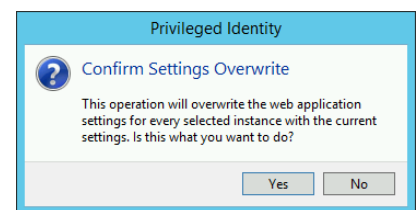
1. In the management console, click **Manage Web App** from the left action pane.



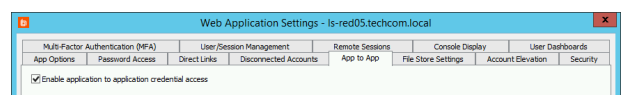
2. Select the desired web application instance from the list, then click **Edit**.



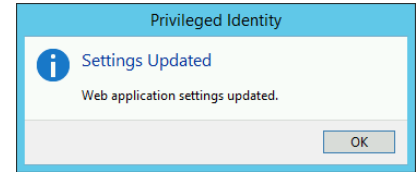
3. When prompted to **Confirm Settings Overwrite**, click **Yes**.



4. Select the **App to App** tab.
5. Make sure **Enable application to application credential access** is checked.
6. Click **OK**.

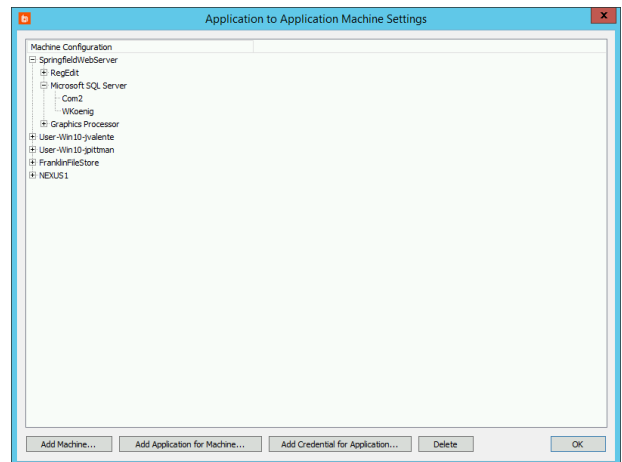


- You will see a notification that the settings were updated. Click **OK**, then close the **Manage Web Application Instances** window.

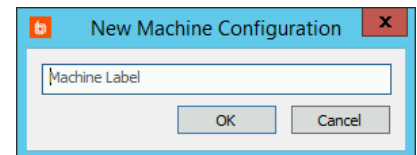


## Configure App-to-App Password Management

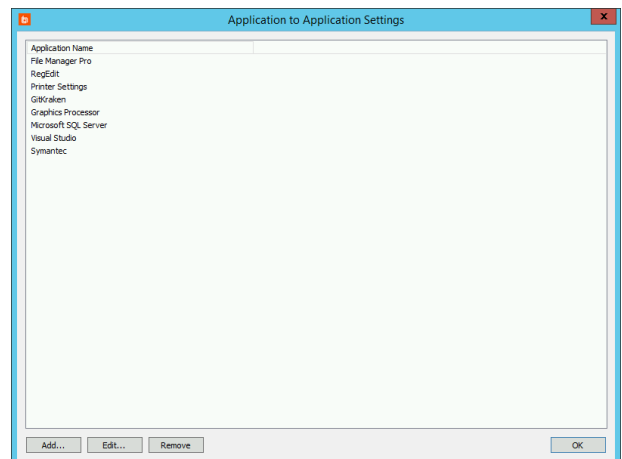
- In the management console, select **Settings > App to App Configuration**.



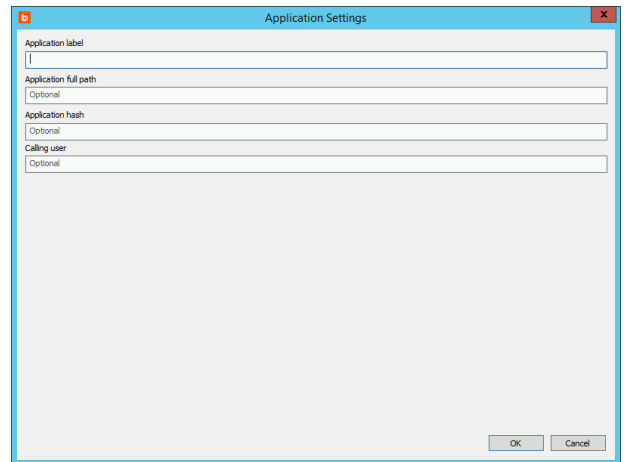
- To add a new machine configuration, click **Add Machine**, then add a descriptive name.



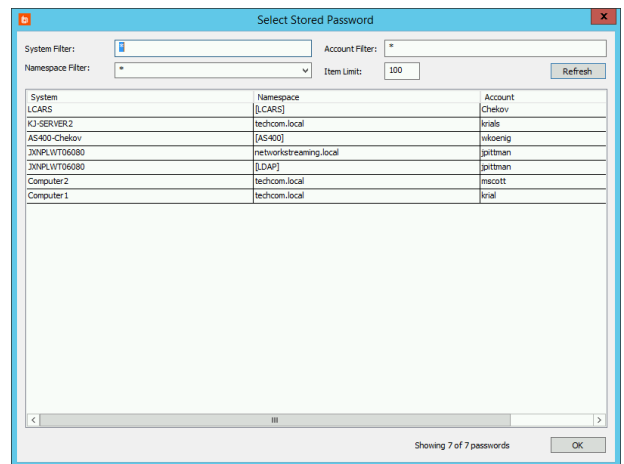
- Select a machine configuration from the list, then click **Add Application for Machine**. This adds an application that the endpoint can execute and get credentials for.



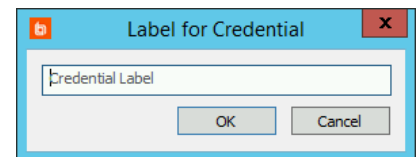
4. To add a new application, click **Add**.
  - **Application label:** Add a descriptive name.
  - **Application full path:** (Optional) Enter a string used to verify the calling process's full application path.
  - **Application hash:** (Optional) Enter a string used to verify the application hash.
  - **Calling user:** (Optional) Enter a string used to verify the fully qualified login name for the Windows user who is calling the process.
5. Select an application to associate with this machine, then click **OK**.




6. Select a credential from the credential store to associate with this application, then click **OK**.



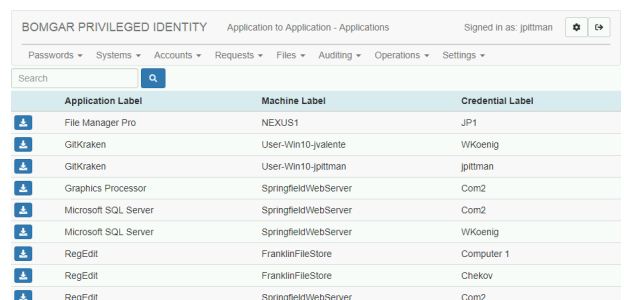
7. Enter a descriptive name to identify this credential when using it on the endpoint, then click **OK**.



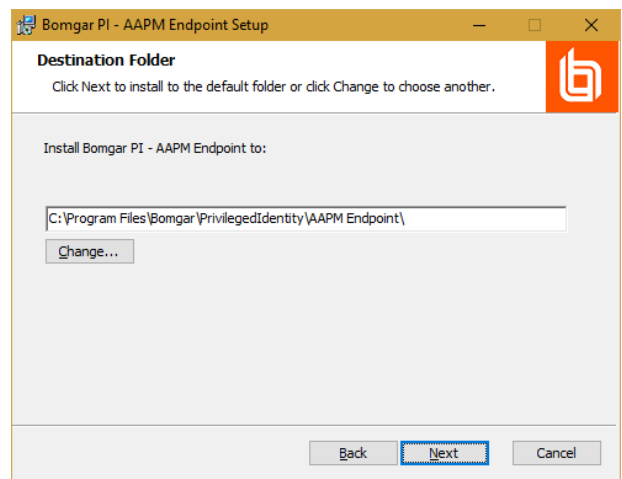
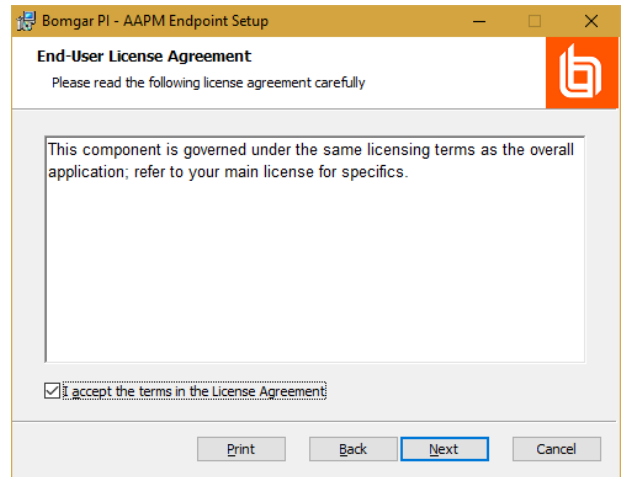
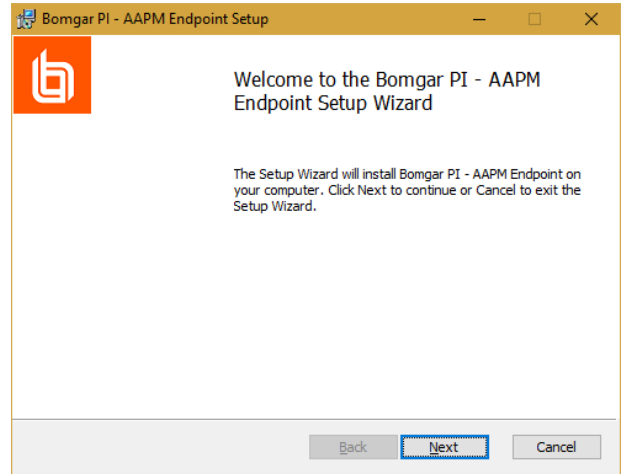
 **Note:** You can add multiple applications to a machine, and you can add multiple credentials to an application.

## Download and Install the App-to-App Endpoint Client

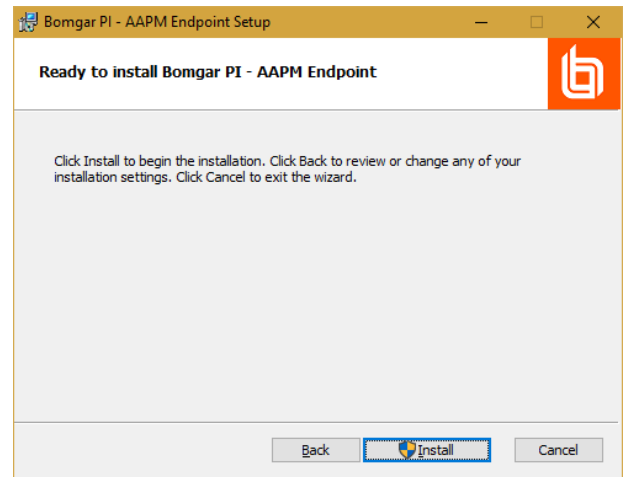
1. Using an **All Access** account, log into the Privileged Identity web app.
2. Select **Passwords > Application to Application**.
3. App-to-app configurations are listed by **Application Label**. You can search on any field.
4. Locate the configuration you want and click the download button. Download the settings file and the Windows service. Transfer both files to the endpoint you want to manage.



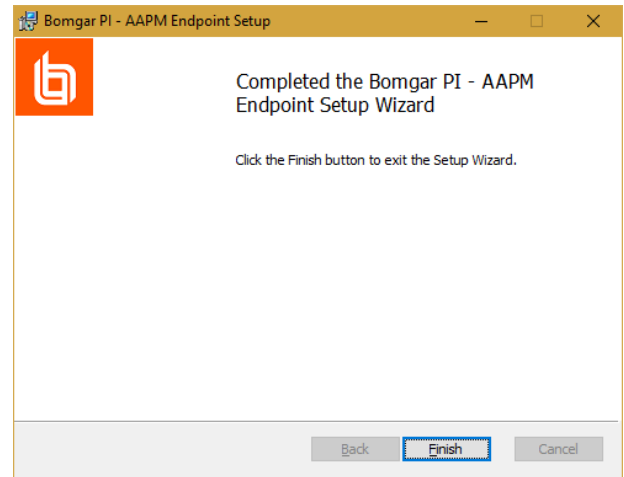
- On the target endpoint, locate **App2AppSetup.msi**. Make sure that **ExplicitConfig.settings.json** is in the same folder. Run the installer.
- On the welcome screen, click **Next**.
- Read and accept the license agreement, then click **Next**.
- Choose a location to install the app-to-app endpoint client software. The default location is **C:\Program Files\Bomgar\PrivilegedIdentity\AAPM Endpoint**. Click **Next**.



- Click **Install**. Installation requires an account with administrative rights.



- When installation completes, click **Finish**.



- Copy the **ExplicitConfig.settings.json** file and paste it into the directory you specified as the destination folder. The default location is **C:\Program Files\Bomgar\PrivilegedIdentity\AAPM Endpoint**.

## Use the App-to-App Endpoint Client

To call the app-to-app client directly from the command line, call **PasswordRetrievalClient\_Console.exe**, followed by the application label. If you have more than one credential configured for this app, specify it here. You may also add a comment. For example, to call a configuration called "Microsoft SQL Server" using the "WKoenig" credential, the command line would be:

```
PasswordRetrievalClient_Console.exe --ApplicationID "Microsoft SQL Server" --CredentialID WKoenig --
Comment "Test Run"
```

When run, this command retrieves the credential and prints it to the command line.

If the app configuration specifies a calling user, only the named user can successfully run the command. Anyone else who runs the command will see a blank output.

Similarly, if the application path and/or hash were specified in the app configuration, the command must be executed from within an app matching those requirements.

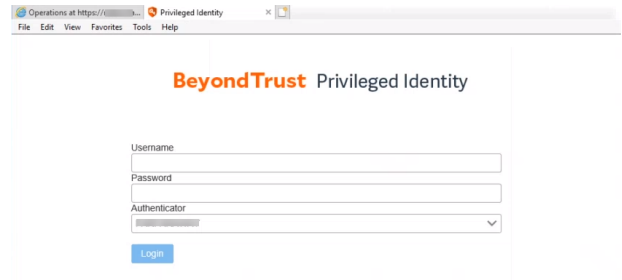


## Web Application Access

This chapter describes the technical configurations required for web application (and web service) access.

The basic flow for a user is as follows:

1. Go to the web application (or web service) URL.
2. Select an authenticator. On the web application, this is the last option on the login page, though it is mentioned first here as SAML and OAuth users will select this option before they can provide their username and password.
3. Supply a username and password. Scenarios using Integrated Windows Authentication and/or certificates will likely not need provide an additional username and password.
4. Supply any MFA information.
5. Get authenticated. This happens by the authentication server, such as Active Directory or the SAML provider.
6. Get authorized. Delegations are evaluated by Privileged Identity based on the identity information provided by the user, authentication server, and delegations granted the identity.
7. Log into the web application or service.



**i** For more information, please see "[Configure Authentication Servers](#)" on page 310.

**i** For more information, please see "[Configure MFA](#)" on page 335.

## Configure Authentication Servers

Authentication Servers are used by Privileged Identity to help perform user authentication for web site and web service logons. When Privileged Identity is first installed, authentication server entries for any trusted Windows domains (relative the management console host) will be added automatically. If you delete them, they will be re-added on the next console restart. If you change the authenticator label, new entries will be created with the original domain names.

Beyond the automatically created entries for trusted domains, you may need to add new authentication server entries if you will be leveraging RADIUS, OAuth, SAML, LDAP directories or untrusted Windows domains.

## LDAP Authentication Servers

To authenticate users using an LDAP server, an LDAP Authentication Server entry must be created. These are created automatically when you add an entry to one of the LDAP nodes (e.g. Oracle Internet Directory or ViewDS directory) or you may add one to the authentication server entries list.

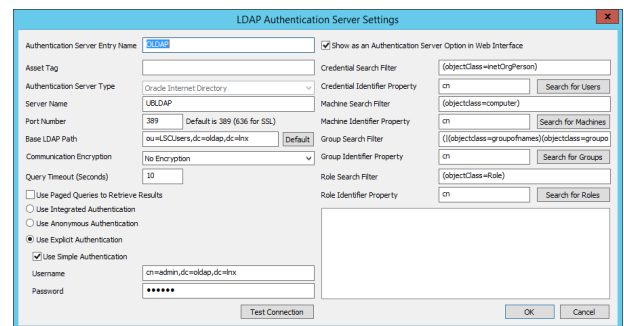
To add the authentication server, you will need the following information:

- Server name
- Port and SSL requirements
- LDAP Authentication account
- User query and identifier attributes
- Base LDAP path

### Adding an LDAP Authentication Server

1. Go to **Delegation | Authentication Servers**.
2. Click **Add LDAP**.
3. Supply the following information:

- **Authentication Server Entry Name** - the name friendly directory name. This value will be appended to the server name when the directory is added to the list. When also used as an authentication server entry, this name will appear in the authenticators list on the web site's login page.
- **Asset Tag** - optional - add an asset tag for this directory.
- **Authentication Server Type** - this entry will be filled out depending on what node you are adding your directory service to. The sole purpose of this selection is to define the default search filters for the right side of the dialog. It does not actually matter what value is supplied here.
- **Server Name** - the name of the server Privileged Identity should bind to. This can be a short name, FQDN, or IP address.
- **Port Number** - the port your directory is listening on. The default port is the default non-secured LDAP port, 389. If you know your directory requires SSL/TLS, then set the port to 636. Otherwise, supply an alternate port if the directory is not listening on the default port.
- **Base LDAP Path** - the path from which all object searches will start. When used as an authentication server, this value will be appended to the user's name automatically when logging into the web site.
- **Communication Encryption** - Choose from No Encryption (default), Use Start TLS, or Use SSL. If using TLS or SSL, the Privileged Identity host making the connection must trust the cert and there can be no certificate errors.
- **Query Timeout (Seconds)** - The time that any query can take before the call times out. The default is 10 seconds. Set this to a higher value if the target LDAP server is slow to respond.
- **Use Pages Queries to Retrieve Results** - The use of this option is directory specific. If the login is successful and search filters are valid, enable this option if queries fail. Not all directories support paged queries.
- Authentication:
  - **Use Integrated Authentication** - this option is set by default for Windows Domains and will use the credentials of the calling user during web site/service logins when a username and password is provided or will use the COM application identity credentials when Integrated Windows Authentication is performed if no username and



password is provided. For management operations, the interactive user or deferred/zone processor account will be used to perform lookups and management. This option is typically not supported by anything other than Windows Active Directory domains.

- **Use Anonymous authentication** - if the directory supports lookups using anonymous authentication, this option may be used. Management operations (password resets) will typically fail with this configuration.
  - **Use Explicit Authentication** - lookup and password reset functions will use the account name and password specified in the username and password field. Enable Use Simple Authentication to pass the exact name shown in the username field.
- **Show as an Authentication Server Option in the Web Interface** - enable this option to make this LDAP server available to the web site for user logins. The Authentication Server Entry Name will be shown in the Authenticators list.
  - **{Option} Search Filter** - the filter that will be used to locate user, computer or groups from the Base LDAP Path.
  - **{Option} Identifier Property** - The attribute that, when present, will be used to identify the object. For web site logins, this value will be pre-pended to the username. Use the appropriate search button to test the search filter and identifier property query. Results will be shown in the results pane in the lower right corner.
4. Enable the option to **Show as an authentication server option in web interface** to allow users to connect with this LDAP server.
  5. Click **OK**.

## OAuth Authentication Servers

To authenticate users using an OAuth server, an OAuth Authentication Server entry must be created. These entries are reliant on an OAuth capable account store already existing in the solution.

There are two OAuth capable account stores in Privileged Identity:

- Azure Active Directory
- Salesforce (Force.com)

There are also two additional OAuth capable Authentication Servers available to Privileged Identity:

- Facebook
- Google

## Adding an OAuth Authentication Server

1. In the management console go to **Delegation | Authentication Servers**.
2. Click **Add OAuth**.
3. Follow the instructions for the specific OAuth Provider:
  - "OAuth - Azure AD" on page 314
  - "OAuth - Salesforce" on page 315
  - "OAuth - Facebook" on page 316
  - "OAuth - Google" on page 318

Regarding multi-factor authentication, MFA options provided within Privileged Identity are not supported for OAuth-based users. Rather, MFA must be provided by the OAuth provider.

## OAuth - Azure AD

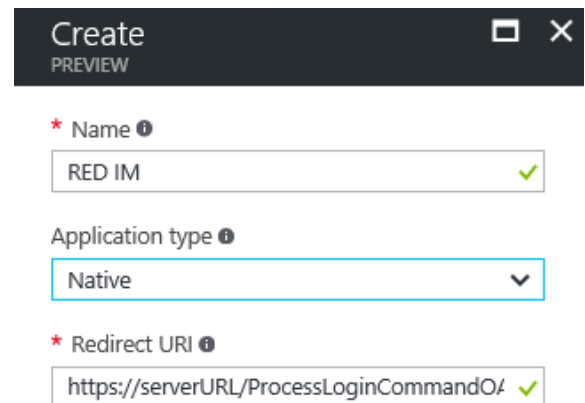
To add the authentication server, you will need the following information:

Which target Azure Active Directory account store to target.

Client ID.

### Adding an Oauth Application to Azure AD

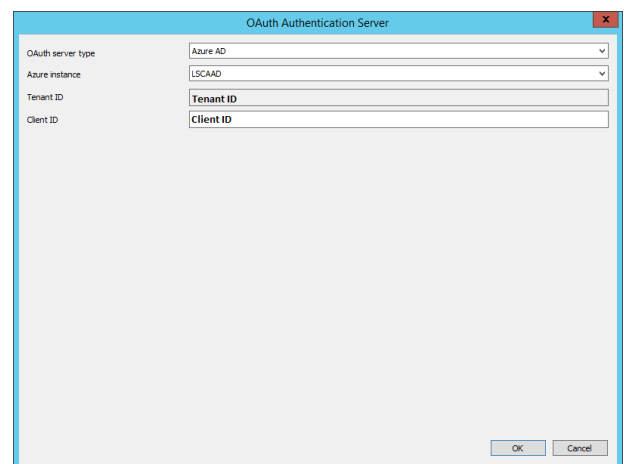
1. Login to your Azure AD administrative web site.
2. In the left-hand navigation pane, choose **More Services**, click **App Registrations**, and click **Add**.
3. Provide the following information:
  - **Name** - A friendly name for the application that will be visible in the Azure portal.
  - **Application Type** - Native.
  - **Redirect URI** - This is the endpoint name on the Privileged Identity web application host. This is URL is typically **https://{serverURL}/PWCWeb/auth/oauth2**. Replace the {serverURL} with the full and correct URL to the web application that a user would type into their browser. The address put here should not contain any redirects.
4. Click **Create**. The application will be added to the list of applications.
5. Click the new application to edit it.
6. Make note of the **Application ID**. This must be entered into the Client ID field in the authentication server entry.



### Adding an OAuth Authentication Server

Once the AzureAD OAuth application has been created, and an Azure AD instance has been added correctly to a management set, it may then be used as an authentication server.

1. Open the management set that contains the Azure AD account store previously added.
2. Right-click on the Azure AD instance and click **Add instance as Authentication Server**. Go to **Delegation | Authentication Servers**.
3. Click **Add OAuth**.
4. Supply the following information:
  - Client ID. The client ID is derived from directly from the target application in Active Directory.
5. Click **OK**.



## OAuth - Salesforce

To add the authentication server, you need the following information:

- Which target Salesforce account store to target.
- Client ID.

### Adding an OAuth Application to Salesforce

1. Go to the Force.com REST API Developer Guide and follow the steps in the "Defining Connected Apps" topic, located here: [https://developer.salesforce.com/docs/atlas.en-us.api\\_rest.meta/api\\_rest/intro\\_defining\\_remote\\_access\\_applications.htm](https://developer.salesforce.com/docs/atlas.en-us.api_rest.meta/api_rest/intro_defining_remote_access_applications.htm)

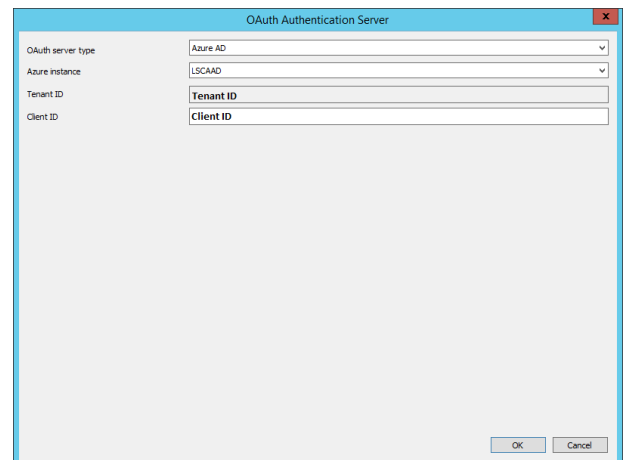
Note the following:

- a. When entering a name for the application, choose a name that indicates that Privileged Identity is the application that is authenticating to Salesforce
  - b. When entering the Callback URL, enter the web application URL and specify **PWCWeb/auth/oauth2** as the resource - for example, **https://serverURL/PWCWeb/auth/oauth2**.
2. Copy and save the Consumer Key and Consumer Secret values. You will need these values when you add the Salesforce directory instance to Privileged Identity.

### Adding an OAuth Authentication Server

Once the Salesforce OAuth application has been created, and a Salesforce instance has been added correctly to a management set, it may then be used as an authentication server.

1. Open the management set that contains the Salesforce account store previously added.
2. Right-click on the Salesforce instance and click **Add instance as Authentication Server**.
3. Go to **Delegation | Authentication Servers**.
4. Click **Add OAuth**.
5. Supply the following information:
  - Client ID. The client ID is derived from directly from the connected app in Salesforce.
6. Click **OK**.

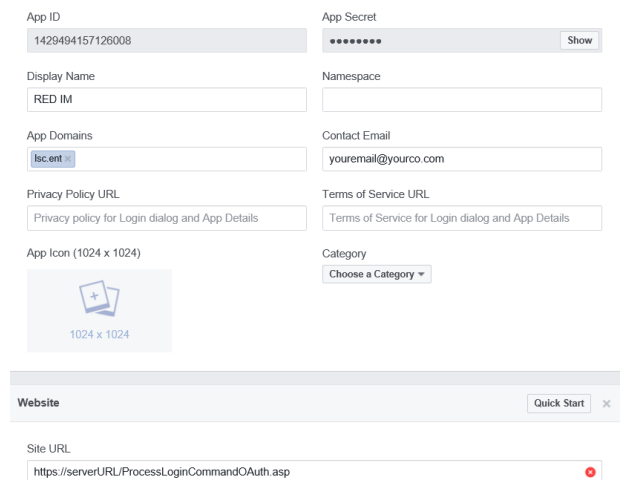
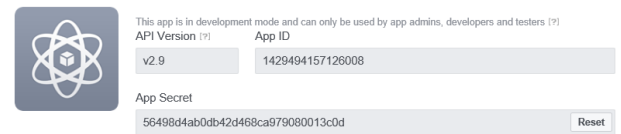


## OAuth - Facebook

To use Facebook OAuth authentication, you and the users who will use your application, must have a Facebook account. This section documents adding an OAuth application to Facebook and adding Facebook as an OAuth Server to Privileged Identity.

### Adding an Oauth Application to Facebook

1. Navigate to <https://developers.facebook.com>.
2. Login with your Facebook account.
3. If you have not previously configured any applications, click Get Started and accept the agreements to continue. Otherwise click **My Apps | Add New App**.
4. In the application, note the following items:
  - **App ID** - This will be entered into the App ID field.
  - **App Secret** - This will be entered into the App Secret field.
5. Next click **Settings**.
6. Supply the App Domains. This is the fully qualified domain name of server that will be contacting the Facebook application.
7. Click **Add Platform**.
8. Select **Web Site**.
9. Add the **Site URL** - This is the endpoint name on the Privileged Identity web application host. This is URL is typically **https://{serverURL}/PWCWeb/auth/oauth2**. Replace the {serverURL} with the full and correct URL to the web application that a user would type into their browser. The address put here should not contain any redirects.
10. Click **Save Changes**.
11. If desired, go to **Settings | Advanced** to configure additional MFA requirements for the application.



### Adding the OAuth Authentication Server Entry

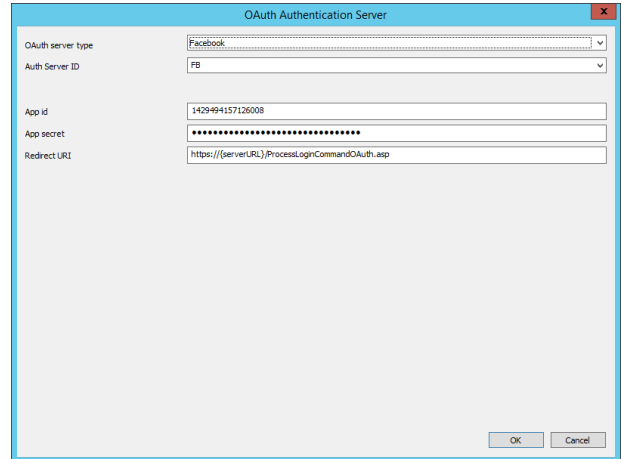
After the Facebook application has been created, it can be added as an authentication server entry via the management console.

1. In the management console, go to **Delegation | Authentication Servers**.
2. Click **Add OAuth**.



3. Supply the following information:

- **OAuth Server Type** - Set to Facebook.
- **Auth Server ID** - This is the friendly name for this authentication server that will be visible in the authenticators field of the web application.
- **App id** - The application ID as noted from step 4 above.
- **App secret** - The application secret from step 4 above.
- **Redirect URL** - This is the endpoint name on the Privileged Identity web application host. This is URL is typically **https://{serverURL}/PWCWeb/auth/oauth2**. Replace the {serverURL} with the full and correct URL to the web application that a user would type into their browser. The address put here should not contain any redirects.



The screenshot shows a dialog box titled "OAuth Authentication Server" with the following fields:

OAuth server type	Facebook
Auth Server ID	FB
App id	1429494157126008
App secret	*****
Redirect URL	https://{serverURL}/ProcessLoginCommandOAuth.asp

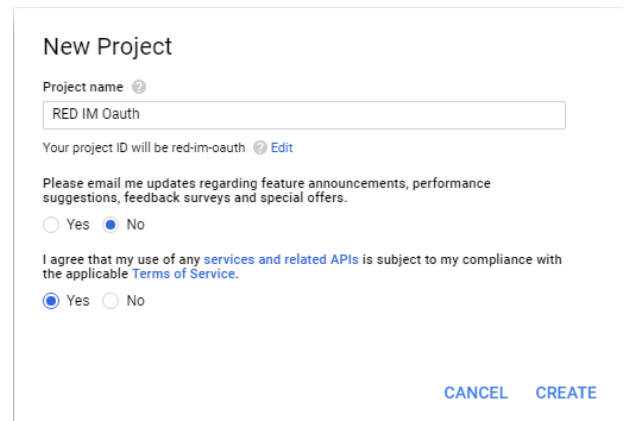
Buttons: OK, Cancel

## OAuth - Google

To use Google OAuth authentication, you and the users who will use your application, must have a Google account. This section documents adding an OAuth application to Google and adding Google as an OAuth Server to Privileged Identity.

### Adding an Oauth Application to Google

1. Log into the Google API Manager at <https://console.developers.google.com> using your google account.
2. Select **Credentials** from the left pane.
3. If you don't have a project, you must create a project. Click **Create a project**. If you do have a project, you can go straight to step 9.
4. Supply the following information:
  - **Project name** - This is a friendly name for the google application.
  - **Email updates** - Select yes or no to opt in our out of Google's advertising machine.
  - **Terms of Service** - You must select Yes to continue. Be sure to read the terms and conditions before agreeing!
5. Click **Create**.
6. On the Credentials page, click **Create credentials**.
7. Select **OAuth Client ID**.
8. If it hasn't been previously configured, Click **Configure consent screen**.
  - Configure the OAuth consent screen settings then click **Save**. You must at least provide a product name on this screen.
9. On the **Create Client ID** page, supply the following information:
  - **Application Type** - Web Application.
  - **Name** - A friendly name for this application.
  - **Authorized redirect URIs** - Supply a name as follows: **https://serverurl.example.com/PWCWeb/auth/oauth2**. Note here that the redirection URI is limited to top level public domains (such as .com or .org). That means your Privileged Identity web server must be published on the internet so it can be verified by Google.
10. Click **Create**.
11. Google will then display the OAuth client page which lists the client ID and client secret. Configuring the OAuth Authentication Server entry for Google will require both of those elements. You can edit the application later to re-obtain these items if needed.
12. Click **OK**.



**New Project**

Project name <sup>?</sup>

RED IM OAuth

Your project ID will be red-im-oauth <sup>?</sup> [Edit](#)

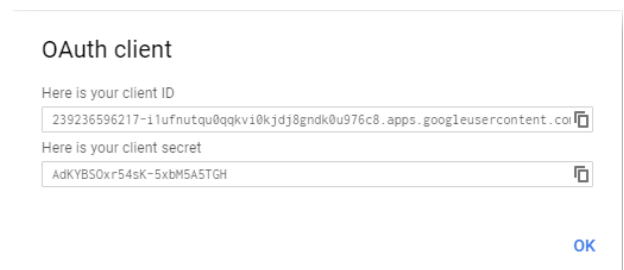
Please email me updates regarding feature announcements, performance suggestions, feedback surveys and special offers.

Yes  No

I agree that my use of any [services and related APIs](#) is subject to my compliance with the applicable [Terms of Service](#).

Yes  No

**CANCEL** **CREATE**



**OAuth client**

Here is your client ID

239236596217-11ufnutqu0qqkv10kj8gndk0u976c8.apps.googleusercontent.com

Here is your client secret

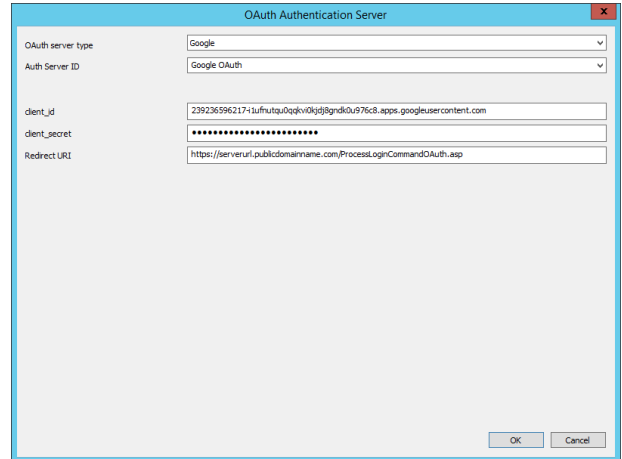
AdKYB50xr54sK-5xbM5A5TGH

**OK**

### Adding the OAuth Authentication Server Entry

After the Google application has been created, it can be added as an authentication server entry via the management console.

1. In the management console, go to **Delegation | Authentication Servers**.
2. Click **Add OAuth**.
3. Supply the following information:
  - **OAuth Server Type** - Set to Google.
  - **Auth Server ID** - This is the friendly name for this authentication server that will be visible in the authenticators field of the web application.
  - **client\_id** - The google client ID as noted from step 14 above.
  - **client\_secret** - The google client secret as noted from step 14 above.
  - **Redirect URL** - Enter the same URL entered as in the Authorized redirect URIs in step 12 above. Supply a name as follows:  
**https://serverurl.example.com/PWCWeb/auth/oauth2.**
4. Click **OK**.



The screenshot shows a dialog box titled "OAuth Authentication Server" with the following fields:

- OAuth server type: Google
- Auth Server ID: Google OAuth
- client\_id: 2392365962171ufrnuiqu0qkiv0iqd@gnrdk0u976c8.apps.googleusercontent.com
- client\_secret: [Redacted]
- Redirect URL: https://serverurl.publicdomainname.com/ProcessLoginCommandOAuth.asp

Buttons for "OK" and "Cancel" are located at the bottom right of the dialog.

## SAML Authentication Servers

To authenticate users using an application must be created in the SAML provider and a SAML Authentication Server entry must be created.

Regarding multi-factor authentication, MFA options provided within Privileged Identity are not supported for SAML-based users. Rather, MFA must be provided by the SAML provider.

Privileged Identity has been tested with the following SAML providers:

- "SAML - ADFS" on page 322
- "SAML - Okta" on page 328
- "SAML - OneLogin" on page 330
- "SAML - Ping" on page 332

Privileged Identity also provides a generic SAML provider configuration as well.

To add the authentication server, you will need the following information:

- Audience Application
- SAML login redirection page
- SAML issuer



**Note:** An option is available to override the domain for SAML claims. Contact BeyondTrust Support if this is required.

## Adding an SAML Authentication Server

1. Go to **Delegation | Authentication Servers**.
2. Click **Add SAML**.
3. Supply the following information:
  - **SAML server type** - Choose the best matching provider from the dropdown list.
  - **Authenticator name** - This is a friendly name that will appear in the Authenticator list of the web application.
  - **Audience application** - The name of the application as found in the SAML provider application.
  - **SAML login redirection page** - The URL that SAML users will be directed to to perform the SAML login before being redirected back to the Privileged Identity login page. This page is supplied by the SAML provider's application.
  - **SAML Issuer** - The issuer URL provided by the SAML provider's application.
  - **X.509 Certificate** - The certificate generated for use with the application created in the SAML provider's application.

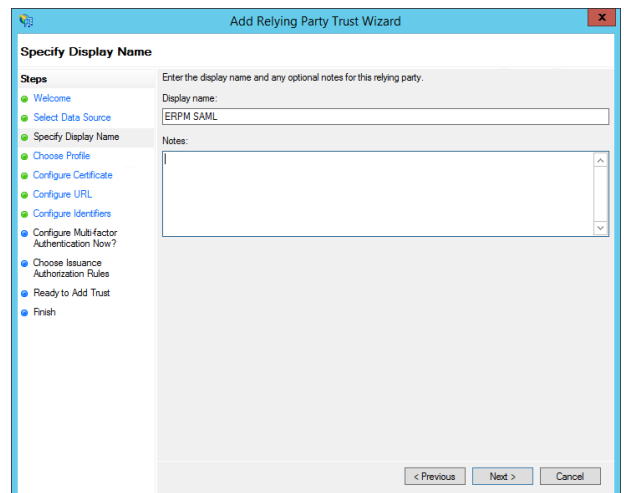
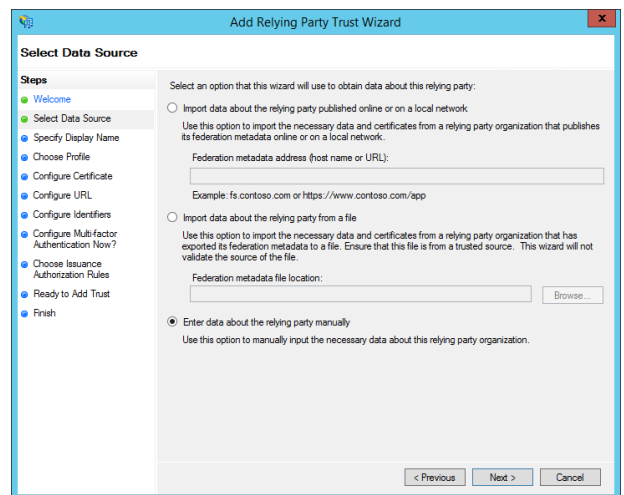
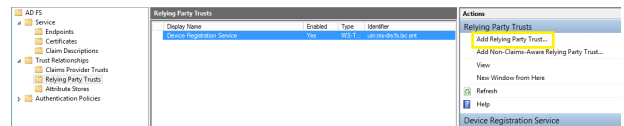
4. **Map SAML assertion attributes to permissions in ERPM** - When enabled, Privileged Identity will accept group and role assertions made by the SAML provider. To limit the accepted assertions made by the SAML provider, click Restrictions to define a white list of allowed assertions. This means that even if the SAML provider asserts the user is in group1, group2, and group 3, the restrictions can limit the allowed assertion to "group 3" only thus granting permissions to only "group 3" and any roles the user is explicitly added to
5. Click **OK**.

## SAML - ADFS

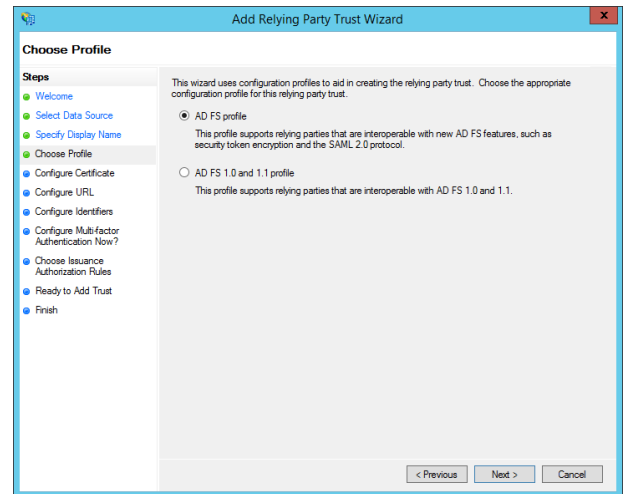
This topic shows how to configure an application in ADFS to provide SAML authentication services.

### Adding an Application in ADFS

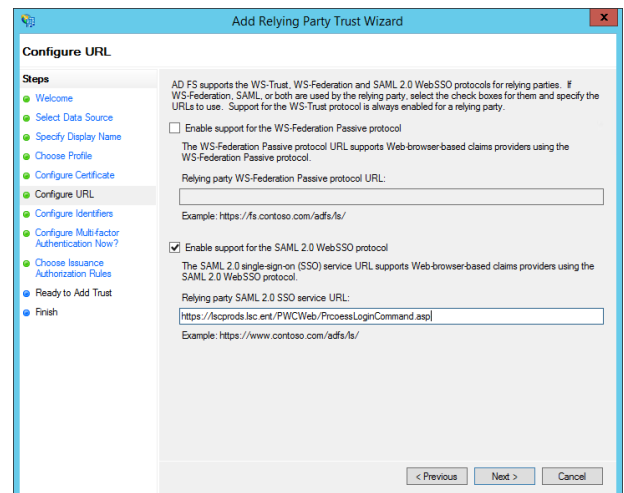
1. Open the ADFS administrative console
2. Expand **ADFS | Trust Relationships | Relying Party Trusts**.
3. Click **Add Relying Party Trust**.
4. On the **Add Relying Part Trust Wizard** welcome screen, click **Start**.
5. On the **Select Data Source** page, select **Enter data about the relying party manually**.
6. Click **Next**.
7. On the **Specify Display Name**, specify a friendly name and add any helpful notes for this application.
8. Click **Next**.



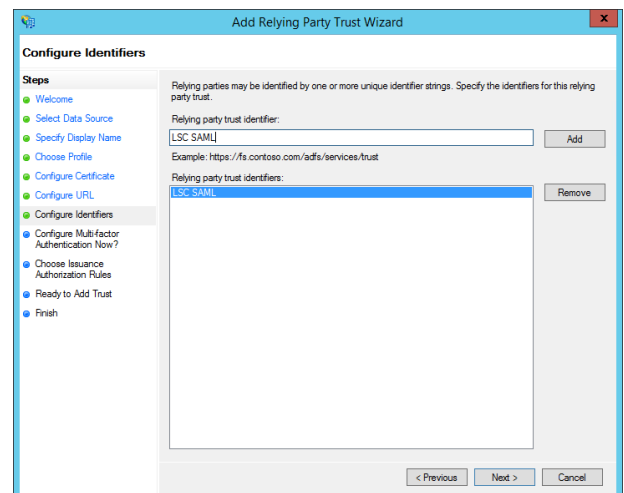
9. On the Choose Profile page, select **AD FS profile**. This enables the use of SAML 2.0.
10. Click **Next**.
11. On the **Configure Certificate** page, click **Next**. Encrypting of assertions is not currently supported.



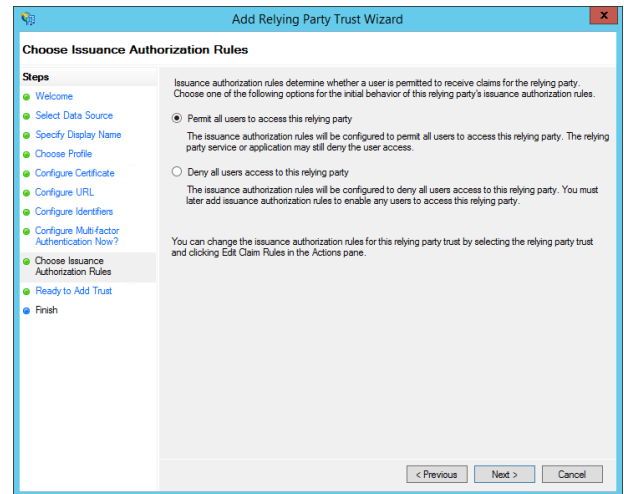
12. On the **Configure URL** page, make the following configurations:
13. Enable **Enable support for the SAML 2.0 WebSSO protocol**.
14. **Relaying party SAML 2.0 SSO service URL** - This is the endpoint name on the Privileged Identity web application host. This URL is typically **https://{serverFQDN}/SAML**. Replace the {serverFQDN} with the full and correct FQDN to the web application that a user would type into their browser. The address put here should not contain any redirects.
15. Click **Next**.



16. On the Configure Identifiers page, specify the **Relaying party trust identifier** and click **Add**. It is recommended to use a simple name. This name will be also be entered into the the Audience application field in the authentication server entry.
17. Click **Next**.
18. On the **Configure Multifactor Authentication Now?** page, configure any MFA requirements for this application. MFA for SAML providers is the exclusive domain of the SAML provider and not covered in this document.
19. Click **Next**.



20. On the Choose Issuance Authorization Rules page, choose the appropriate option to determine which ADFS users will be able to use this application. If you are unsure, select **Permit all users to access this relying party**. Selecting the second option, Deny all users access to this relying party, will require further configurations by you to allow each specific user access to this application and is not covered by this document.
21. Click **Next**.
22. On the **Ready to Add Trust** page, verify the Identifiers and endpoints you previously supplied and click **Next**.
23. On the **Finish** page, click **Close**.
24. After clicking Finish, a Claims Rule dialog will pop up. You may configure this now or later. This application will not work with Privileged Identity until claims rules are configured.



## Adding Claims for ADFS

Now that the application has been created, to make the application usable requires the creation of claims rules. There are many ways to setup claims (also known as assertions).

Privileged Identity looks for up to four attributes from a SAML response:

- **NameID - required** - The name of the user.
- **DomainUser - optional** - specify additional usernames for the user.
- **DomainGroup - optional** - specify domain groups for the user.
- **Role - optional** - Specify roles for the user.

The following are the expected SAML XML response values:

- **DomainUser** = <NameID>user@example.com</NameID>
- **DomainGroup** = <Attribute Name="DomainGroup">
- **Role** = <Attribute Name="Role">



**Note:** You must enable the Privileged Identity authentication server option **Map SAML assertion attributes to permissions in ERP** (**DomainGroup, DomainUser, Role**).

When creating a custom claims rule, the returned values for DomainGroup and Role should be assigned as AttributeValue. The returned attribute value for DomainGroup or Role is what will need to be added to the list of delegations in Privileged Identity.

For example, if you add a group called SAML\_Admin, the returned SAML response for the DomainGroup AttributeValue will need to look like this:

```
<AttributeValue>SAML_Admin</AttributeValue>
```



The delegation group or role added to the identity list in Privileged Identity needs to match exactly what is returned from the claim if the claim will be processed.

If you add a group called "LSC\SAML\_Admin", the returned SAML response for the DomainGroup AttributeValue will need to look like this:

Request User	Delegation Role
SAML All Access	Delegation Role
SAML Request	Delegation Role
SAML_Admin	Domain Group
SAML_ROLE	Domain Group
Shield.Ops	Delegation Role

```
<AttributeValue>LSC\SAML_Admin</AttributeValue>
```

Depending on how you have your custom Claims Rule written, you may see multiple values for AttributeValue. All values are valid, but only the one(s) listed in the Privileged Identity delegated identities list will be processed.

An example of a custom rule looks like the following.

```
c: [Type = = "http://schemas.microsoft.com/ws/2008/06/identity/claims/windowsaccountname", Issuer == "AD AUTHORITY"]
=> issue(store = "Active Directory", types = ("DomainGroup"), query = ";tokenGroups;{0}", param = c.Value);
```

An example login will flow like this:

User Fred opens the web application and selects the ADFS logon provider.

The Web Application redirects to the ADFS SAML provider web page to authenticate the user.

Fred will log in using his UPN: fred@adfsexample.int, ADFS, will authenticate Fred.

ADFS will then send a response back to Privileged Identity in the form of an XML response and redirect the user back to the web site. If all of the above attributes in the response are as expected, Fred will be logged into the web site as, Fred@adfsexample.int.

The SAML response will look similar to this.

```
<Subject>
  <NameID>Fred@example.com</NameID>
  <SubjectConfirmation Method="urn:oasis:names:tc:SAML:2.0:cm:bearer"><SubjectConfirmationData
NotOnOrAfter="2017-01-05T16:25:08.275Z"
Recipient="https://domain.example.com/SAML"/></SubjectConfirmation>
</Subject>
<Conditions NotBefore="2017-01-05T16:20:08.275Z" NotOnOrAfter="2017-01-05T17:20:08.275Z">
  <AudienceRestriction>
    <Audience>LSC SAML</Audience>
  </AudienceRestriction>
</Conditions>
<AttributeStatement>
  <Attribute Name="DomainGroup">
    <AttributeValue>SAML_ROLE</AttributeValue>
    <AttributeValue>SAML_Admin</AttributeValue>
  </Attribute>
</AttributeStatement>
```

## Custom Claims Example

When the above Claim Rules do not meet your requirements, you can create customized Claim Rules that would. Privileged Identity is looking for the two attributes: NameID, and AttributeValue. The Attribute Value will be one of three different types: DomainGroup, DomainUsers, DomainRole. These values are defined in Claim Rules that will then be passed in a SAML Response in XML format. Here is an examples of building customized Claim Rules to pass the individual user as a qualified user (Domain\Username). The rule names below, e.g. "Domain User 1", are example names for the Claim Rules, and can be modified to whatever you want.

You will also need to change the "Domain\" to pertain to your domain, or a domain you want users to qualify from.

### Domain User 1

Queries the AD sAmAccountName for user and assigns a attribute type as "DomainUser"

```
c: [Type = = "http://schemas.microsoft.com/ws/2008/06/identity/claims/windowsaccountname", Issuer == "AD AUTHORITY"]
=> add(store = "Active Directory", types = ("DomainUser"), query = ";samaccountname;{0}", param = c.Value);
```

### Domain User 2

Assigns an AttributeValue as "DomainUser"

```
c: [Type = = "DomainUser", Value =~ "^(?i)"]
=> issue(Type = "DomainUser", Value = "Domain\" + c.Value);
```

### NameID1

Queries the AD for qualified user and assigns it to a c.Value

```
c: [Type = = "http://schemas.microsoft.com/ws/2008/06/identity/claims/windowsaccountname", Issuer == "AD AUTHORITY"]
=> add(store = "Active Directory", types = ("User"), query = ";samaccountname;{0}", param = c.Value);
```

### NameID2

Passes the NameID as a qualified name.

```
c: [Type = = "User"]
=> issue(Type = "http://schemas.xmlsoap.org/ws/2005/05/identity/claims/nameidentifier", Value = "Domain\" + c.Value);
```

## Domain Group Membership

Passes an AttributeValue of DomainGroup and all of the groups the user is a member of.

```
c: [Type = = "http://schemas.microsoft.com/ws/2008/06/identity/claims/windowsaccountname", Issuer
== "AD AUTHORITY"]
=> issue(store = "Active Directory", types = ("DomainGroup"), query = ";tokenGroups
(domainQualifiedName);{0}", param = c.Value);
```

The above Claim Rules should be created in the order they are listed. They will query the Active Directory's MemberOf, and sAMAccountName, and assign them to attribute values of DomainUser and DomainGroup, then passes all pertinent information, which will allow you to add either the single Identity, or a Group that the user is a member of, to the Delegation List.

The expected SAML Response will look something like this (omitting the first, authentication portion of the response). Also, I have replaced the above "Domain\" with "LSC\" and therefore, you will see an expected "LSC\" output.

```
<Subject>
  <NameID>LSC\lscadmin</NameID>
  <SubjectConfirmation Method="urn:oasis:names:tc:SAML:2.0:cm:bearer"><SubjectConfirmationData
NotOnOrAfter="2017-03-31T15:29:17.539Z"
Recipient="https://server.lsc.ent/SAML"/></SubjectConfirmation>
</Subject>
<Conditions NotBefore="2017-03-31T15:24:17.539Z" NotOnOrAfter="2017-03-31T16:24:17.539Z">
  <AudienceRestriction>
    <Audience>ADFS SAML</Audience>
  </AudienceRestriction>
</Conditions>
<AttributeStatement>
  <Attribute Name="DomainUser">
    <AttributeValue>LSC\lscadmin</AttributeValue>
  </Attribute>
  <Attribute Name="DomainGroup">
    <AttributeValue>LSC\Domain Admins</AttributeValue>
    <AttributeValue>LSC\Domain Users</AttributeValue>
    <AttributeValue>LSC\Schema Admins</AttributeValue>
    <AttributeValue>LSC\Enterprise Admins</AttributeValue>
    <AttributeValue>LSC\SvcAccts</AttributeValue>
    <AttributeValue>LSC\SQLAdmins</AttributeValue>
    <AttributeValue>LSC\ERPM Admins</AttributeValue>
    <AttributeValue>LSC\ADFSSAML</AttributeValue>
  </Attribute>
</AttributeStatement>
<AuthnStatement AuthnInstant="2017-03-31T15:18:09.520Z" SessionIndex="_0a376dd1-8cc7-46df-a5ba-
705e2539940d">
  <AuthnContext>
    <AuthnContextClassRef>urn:oasis:names:tc:SAML:2.0:ac:classes>PasswordProtectedTransport</AuthnCon
textClassRef>
  </AuthnContext>
</AuthnStatement>
</Assertion>
</samlp:Response>
```

## SAML - Okta

This topic shows how to configure an application in Okta to provide SAML authentication services.

### Adding an Application in Okta

1. Login to your Okta admin dashboard.
2. Click **Add Applications**.
3. Click **Create New App**.
4. On the Create a New Application Integration dialog, select **SAML 2.0** and click **Create**.
5. On the General Settings tab, supply the following information:

- **App Name** - This is a friendly name for the application and is visible in the Okta portal.
- **App Logo - Optional** - Supply a graphic to help identify the application.


6. Click **Next**.

7. On the **Configure SAML Settings** page, supply the following information:

- **Single sign on URL** - This is the endpoint name on the Privileged Identity web application host. This is URL is typically **https://{serverFQDN}/SAML**. Replace the {serverFQDN} with the full and correct FQDN to the web application that a user would type into their browser. The address put here should not contain any redirects.
  - Ensure the option **Use this for Recipient URL and Destination URL** is enabled.
- **Audience URI (SP Entity ID)** - Supply a simple name such as RedIM. This name will also be entered in the Audience application field on the authentication server entry.
- **Name ID Format** - Leave as Unspecified.
- **Application username** - Leave as Okta username.

1 General Settings

**App name**

**App logo (optional)** 

**App visibility**


Do not display application icon to users


Do not display application icon in the Okta Mobile app


8. The Attribute Statements and Group Attribute Statements are both optional and will allow configuration of additional assertions, which is not required, though configuring additional assertions will allow you to assert the user (NameID) belongs to other groups or roles. Please refer to your Okta documentation for more information on configuring assertions. These can also be configured later.
  - If using mappings, the expected mapping attributes are DomainUser for users, DomainGroup for groups, or Role for role. The returned mappings would need to exactly match what is added as an identity in the solution already.
9. Click **Next**.
10. Complete the Feedback form if desired, then click **Finish**. You will be brought to the **SignOn** tab for the application.
11. In the **Settings** section, locate the box that says SAML 2.0 is not configured until you complete the setup instructions.
12. Click **View Setup Instructions**.
13. In the **How to Configure SAML 2.0 for {APPNAME} Application** window, make note of the following information:
  - **Identity Provider Single Sign-On URL** - This URL will be placed into the SAML login redirection page field in the authentication server entry.
  - **Identity Provider Issuer** - This URL will be placed into the **SAML issuer** field in the authentication server entry.
  - **X.509 Certificate** - Copy this whole field. It will be placed into the X509 certificate field in the authentication server entry.
14. Finally, assign users to use the application by opening the application from the applications page and clicking **Assign to People**.


**A** SAML Settings


**GENERAL**

Single sign on URL    
 Use this for Recipient URL and Destination URL

Audience URI (SP Entity ID) 

Default RelayState    
If no value is set, a blank RelayState is sent

Name ID format 

Application username 

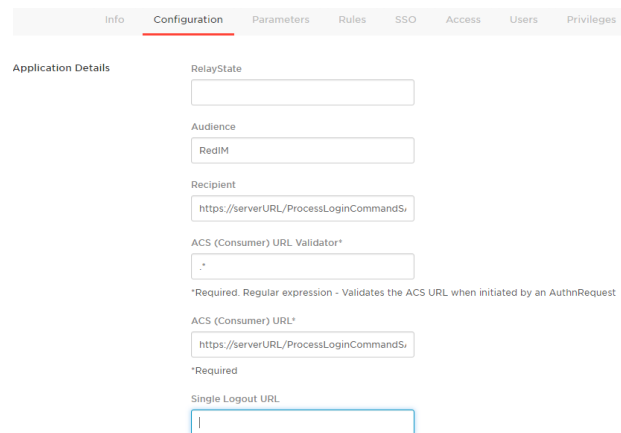
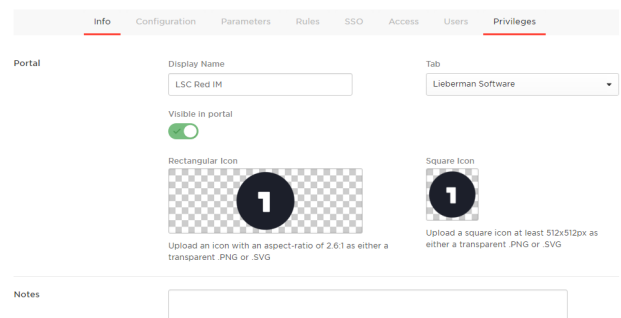
[Show Advanced Settings](#)

# SAML - OneLogin

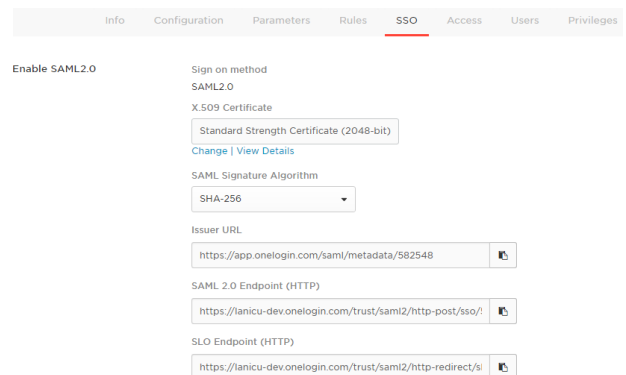
This topic shows how to configure an application in OneLogin to provide SAML authentication services.

## Adding an Application in OneLogin

1. Login to your OneLogin admin web site.
2. Go to **Apps | Add Apps**.
3. Search for SAML Test Connector.
4. Select **SAML Test Connector (IdP w/ attr w/ sign response)**.
5. On the **Info** tab, supply the following information:
  - **Display name** - This is the name of the app as it will appear in the one login web site.
  - **Tab** - Select the appropriate tab. If you have only one tab configured, its name will already be populated.
  - **Visible in portal** - Defaults to enabled. This identifies if users will be able to see the app when they login to the OneLogin portal.
  - Supply visual images to help further identify the app.
  - **Notes** - add any notes to describe what this application is for.
6. Click the **Configuration** tab.
7. On the **Configuration** tab, supply the following information:
  - **Audience** - Supply a simple name such as RedIM. This name will also be entered in the Audience application field on the authentication server entry.
  - **Recipient** - This is the endpoint name on the Privileged Identity web application host. This is URL is typically **https://{serverFQDN}/SAML**. Replace the {serverFQDN} with the full and correct FQDN to the web application that a user would type into their browser. The address put here should not contain any redirects.
  - **ACS (Consumer) URL Validator** - Supply a value of ".\*" without the double quotes.
  - **ACS Consumer URL** - This is the endpoint name on the Privileged Identity web application host. This is URL is typically **https://{serverFQDN}/SAML**. Replace the {serverFQDN} with the full and correct FQDN to the web application that a user would type into their browser. The address put here should not contain any redirects.
8. The Parameters page will allow configuration of additional assertions, which is not required, though configuring additional assertions will allow you to assert the user (NameID) belongs to other groups or roles. Please refer to your OneLogin documentation for more information on configuring assertions.
  - If using mappings, the expected mapping attributes are DomainUser for users, DomainGroup for groups, or Role for role. The returned mappings would need to exactly match what is added as an identity in the solution already.
9. Click the **SSO** tab.



10. On the SSO tab, take the following actions:
11. Ensure the Sign on method is set to **SAML 2.0**.
12. The default certificate strength is 2048 bits. One login allows you set a strength of only 1024 bits. This can cause problems with modern systems. Leave the value at 2048 bits or set it higher if the option is provided in the future.
13. Under the x.509 certificate field, right-click on the **View Details** link and select **Open link in new tab** or **Open link in new window**. **Failure to open the link this way will ultimately cause all of your work to this point to be lost.**
14. Copy the data from the X.509 Certificate field. This data will be pasted into the X509 certificate field in the Authentication Server entry.
15. Set the SAML Signature algorithm to **SHA-256**.
16. Make note of the Issuer URL. This value will be placed into the SAML issuer field in the authentication server entry.
17. Make note of the SAML 2.0 Endpoint (HTTP). This value will be placed into the SAML login redirection field in the authentication server entry.
18. Click **Save** in the top right corner of the window.
19. The finally, allow your OneLogin users to use this application. Go to the Users tab and add your desired users.



Info Configuration Parameters Rules **SSO** Access Users Privileges

Enable SAML2.0

Sign on method  
SAML2.0

X.509 Certificate  
Standard Strength Certificate (2048-bit)  
[Change](#) | [View Details](#)

SAML Signature Algorithm  
SHA-256

Issuer URL

SAML 2.0 Endpoint (HTTP)

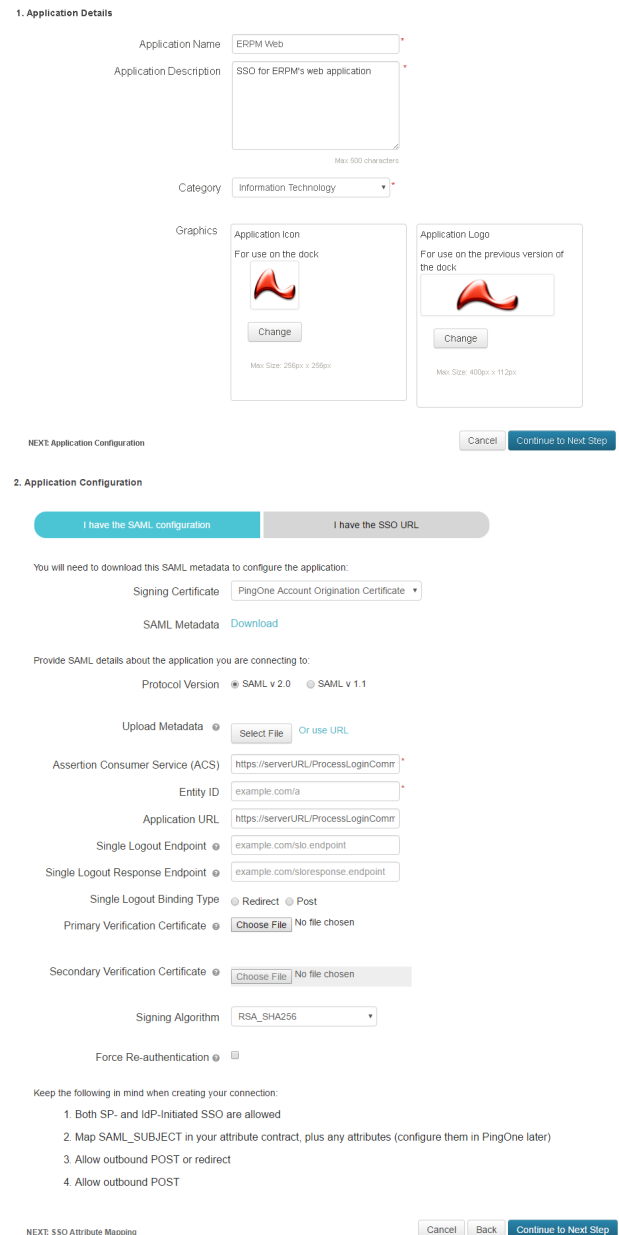
SLO Endpoint (HTTP)

# SAML - Ping

This topic shows how to configure an application in Ping to provide SAML authentication services.

## Adding an Application in Ping

1. Login to your ping admin web site.
2. Go to **Applications > My Applications**.
3. Click **Add Application** and select **New SAML Application**.
4. Configure **Application Details**:
  - **Application Name** - Add in a friendly name for the application. This name will be visible to users who login using SAML authentication.
  - **Application Description** - Provide a description for the application.
  - **Category** - Select a category for the application such as "Information Technology".
  - **Graphics - Optional** - Provide an image to associate with the application.
5. Click **Continue to Next Step**.
6. Configure **Application Configuration** (only the following information is required):
  - **Signing Certificate** - Select PingOne Account Origination Certificate
    - **SAML Metadata** - Click Download. This will download an XML file which will contain the entity information and certificate information required to be entered into the the authentication server entry.
    - **X.509 Certificate** - This information will be located between the **ds:x509** data tags. This
  - **SAML issuer** - This information is the data following the **entityID**.
  - **Protocol Version** - Select **SAML v2.0**.
  - **Assertion Consumer Service** - This is the endpoint name on the Privileged Identity web application host. This is URL is typically **https://{serverFQDN}/SAML**. Replace the {serverFQDN} with the full and correct FQDN to the web application that a user would type into their browser. The address put here should not contain any redirects.
  - **Entity ID** - This maps to the Audience application field on the authentication server entry dialog.



**1. Application Details**

Application Name: ERPM Web

Application Description: SSO for ERPM's web application (Max 500 characters)

Category: Information Technology

Graphics: Application icon (Max Size: 256px x 256px) and Application Logo (Max Size: 400px x 112px)

**2. Application Configuration**

I have the SAML configuration | I have the SSO URL

You will need to download this SAML metadata to configure the application:

Signing Certificate: PingOne Account Origination Certificate

SAML Metadata: Download

Provide SAML details about the application you are connecting to:

Protocol Version:  SAML v 2.0  SAML v 1.1

Upload Metadata:  [Or use URL](#)

Assertion Consumer Service (ACS): https://serverURL/ProcessLoginComin

Entity ID: example.com/a

Application URL: https://serverURL/ProcessLoginComin

Single Logout Endpoint: example.com/slo.endpoint

Single Logout Response Endpoint: example.com/sloresponse.endpoint

Single Logout Binding Type:  Redirect  Post

Primary Verification Certificate:  No file chosen

Secondary Verification Certificate:  No file chosen

Signing Algorithm: RSA\_SHA256

Force Re-authentication:

Keep the following in mind when creating your connection:

1. Both SP- and IdP-Initiated SSO are allowed
2. Map SAML\_SUBJECT in your attribute contract, plus any attributes (configure them in PingOne later)
3. Allow outbound POST or redirect
4. Allow outbound POST

NEXT: SSO Attribute Mapping



- **Application URL** - This is the endpoint name on the Privileged Identity web application host. This is URL is typically **`https://{serverFQDN}/SAML`**. Replace the {serverFQDN} with the full and correct FQDN to the web application that a user would type into their browser. The address put here should not contain any redirects.
  - **Signing Algorithm** - Set to **RSA\_SHA256**.
7. Click **Continue to Next Step**.
  8. The final page will allow configuration of additional assertions, which is not required, though configuring additional assertions will allow you to assert the user (NameID) belongs to other groups or roles. For more information on attribute mapping, see <https://documentation.pingidentity.com/pingone/employeeSsoAdminGuide/index.shtml#useAdvAttributeMapping.html>.
    - If using mappings, the expected mapping attributes are DomainUser for users, DomainGroup for groups, or Role for role. The returned mappings would need to exactly match what is added as an identity in the solution already.
  9. Click **Save & Publish**.
  10. A summary page will appear. On this page is the Initiate Single Sign-On (SSO) URL. Copy this URL. It will be added to the SAML login redirection page field on the authentication server entry.
  11. Ensure application Status will be set to Active and Enabled will be set to Yes.

## RADIUS Authentication Servers

To authenticate users using a RADIUS server, ensure that the web application and web service hosts are listed as clients of the RADIUS server. This process varies by RADIUS server.

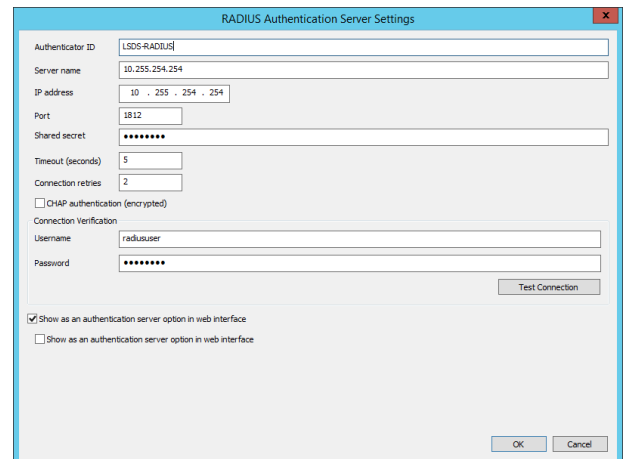
To add the authentication server, you will need the following information:

- RADIUS Port
- RADIUS "security" protocol
- Shared Secret

### Adding a RADIUS Authentication Server

1. Go to **Delegation | Authentication Servers**.
2. Click **Add RADIUS**.
3. Supply the following information:

- **Authenticator ID** - This is the name that will be shown in the authenticators list during a web application logon. Valid characters are 0-9, a-z, and A-Z.
- **Server name** - The name of the RADIUS server as identified in DNS.
- **IP Address** - IP address of the RADIUS server.
- **PORT** - The port of the RADIUS server.
- **Shared Secret** - The shared secret configured for your client in RADIUS.
- **Timeout (seconds)** - Default value is 5. This is the number of seconds to wait for a connection or reply to or from the RADIUS host.
- **Connection retries** - Default value is 2. The number of connection attempts to make to the RADIUS server before returning a connection error.
- **CHAP Authentication** - Enable this value to enable CHAP authentication encryption. The Server must be configured to allow CHAP authentication as well.



4. Verify connectivity by supplying a valid RADIUS username and password, then click **Test Connection**.
5. Enable the option to **Show as an authentication server option in web interface** to allow users to connect with RADIUS.
6. Click **OK**.

## Configure MFA

Privileged Identity supports 2-factor authentication for access to both the management application console and the delegated web interface.

Presently supported two factor implementations include:

- OATH built-in
- OATH compliant
- InfoCrypt
- RADIUS Compliant (based on available options in Privileged Identity)
- RSA
- SafeNet
- YubiKey

## OATH 2-Factor

OATH token authentication is available in both the management console and the web application. No further infrastructure is required for support of this two-factor authentication method when using TOTP tokens; HOTP may require additional elements.



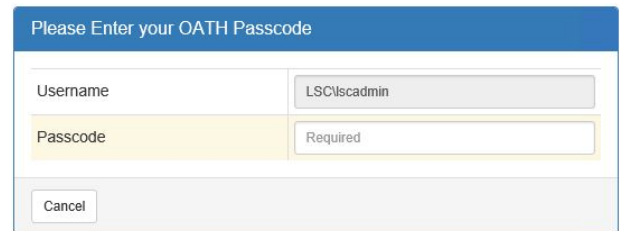
**Note:** Also see the "YubiKey" on page 368 section for more information when using the Yubikey.

## OATH 2-Factor Overview

OATH token authentication is available in both the management console and the web application. No further infrastructure is required for support of this two-factor authentication method when using TOTP tokens; HOTP may require additional elements.

If OATH Tokens are required for web access, then after a user enters their login credentials they will see a login prompt which asks for their OATH Token passcode.

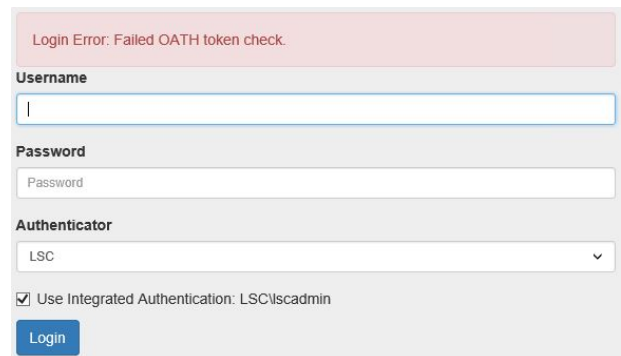
OATH logins can be setup in multiple ways with support for TOTP and/or HOTP tokens. TOTP will send the token key via email or SMS. HOTP will require a physical or soft device that is kept in sync with Privileged Identity. Below is a sample TOTP email message:



A screenshot of a web form titled "Please Enter your OATH Passcode". It contains two input fields: "Username" with the value "LSC\lscadmin" and "Passcode" with the value "Required". A "Cancel" button is located at the bottom left of the form.

```
Message From Privileged Identity (Version: 170417)
Message from 2K8R2-2
Your login requires token authentication
Token Code: 81607010
Token code is valid for 15 minutes
```

Input the **Token Code** and click **Login**. If the login is successful, there will be no further prompts from the OATH system. If the OATH login is unsuccessful, the user will be dropped back to the initial login page with a message stating: **Login Error: Failed OATH token check.**

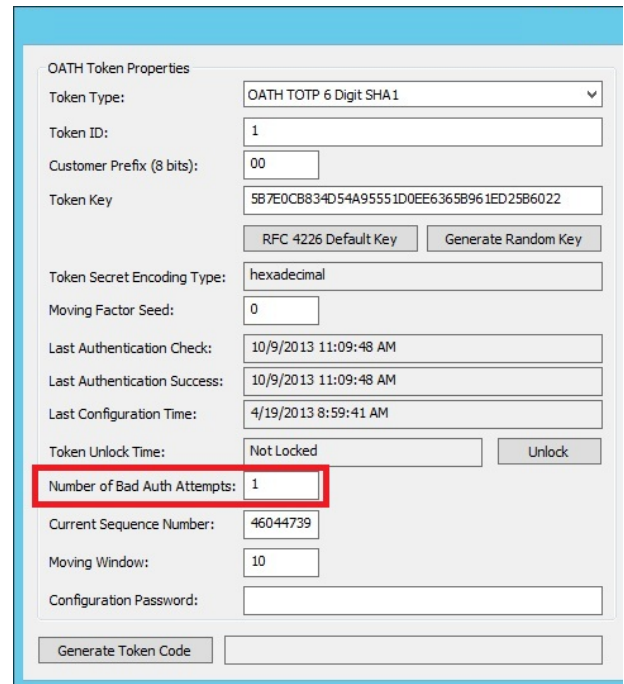


A screenshot of a login form with a red error message at the top: "Login Error: Failed OATH token check." The form includes fields for "Username", "Password", and "Authenticator" (set to "LSC"). A checkbox for "Use Integrated Authentication: LSC\lscadmin" is checked. A "Login" button is at the bottom.

The attempt will be logged in the program's audit logs:

```
Logon - Token authentication check failed - Failed token authentication
LSC\lscadmin check for user LSC\lscadmin - wrong number of passcode digits for assigned token 8/25/2015 12:57:16 PM
```

The user's bad login count will be incremented in the OATH Token Configuration dialog (**Delegation | OATH/Yubico Token Configuration**). Be careful! Too many unsuccessful logins will automatically lock out the user for 15 minutes or require admin intervention to unlock the account.



OATH Token Properties

Token Type:	OATH TOTP 6 Digit SHA1
Token ID:	1
Customer Prefix (8 bits):	00
Token Key:	5B7E0CB834D54A95551D0EE6365B961ED25B6022
	<input type="button" value="RFC 4226 Default Key"/> <input type="button" value="Generate Random Key"/>
Token Secret Encoding Type:	hexadecimal
Moving Factor Seed:	0
Last Authentication Check:	10/9/2013 11:09:48 AM
Last Authentication Success:	10/9/2013 11:09:48 AM
Last Configuration Time:	4/19/2013 8:59:41 AM
Token Unlock Time:	Not Locked <input type="button" value="Unlock"/>
Number of Bad Auth Attempts:	1
Current Sequence Number:	46044739
Moving Window:	10
Configuration Password:	<input type="password"/>
<input type="button" value="Generate Token Code"/>	<input type="text"/>

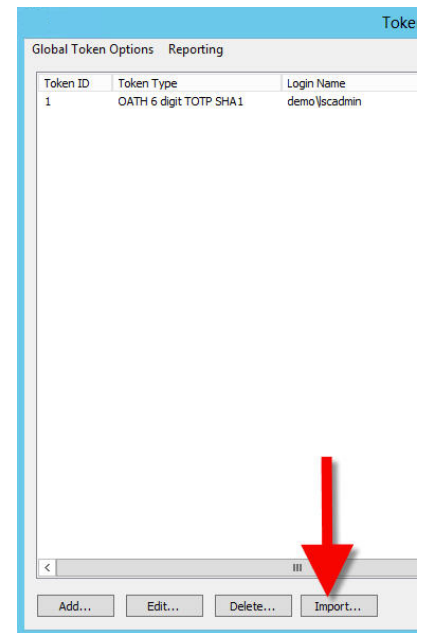
## OATH With Existing Tokens

When tokens are purchased in bulk that are already configured (loaded with a cryptographically random seed), the vendor can provide a CSV file that contains records of token identifies, seeds and other useful information.

Note that the same token seed can be used for most common types of OATH tokens, so the first step is to select the target type of token for the provided seed file.

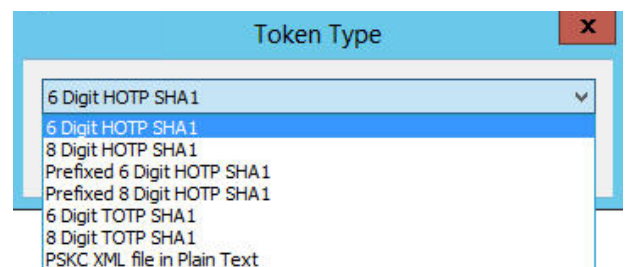
### Importing OATH/Yubico Tokens

1. Select **Delegation | OATH/Yubico Token Configuration**.
2. Click **Import**.

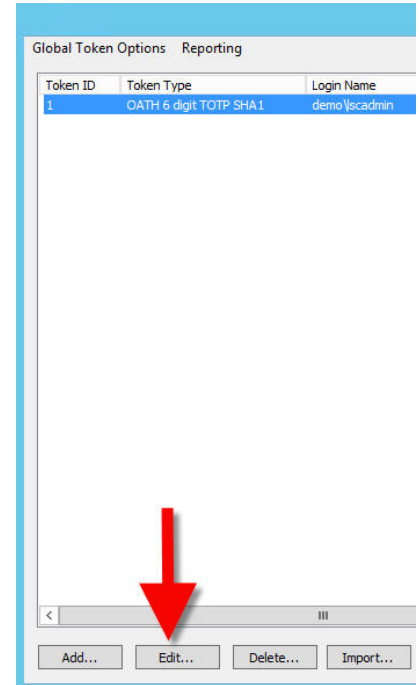


3. Choose the token type being imported. The same token seed can be used for most common types of OATH tokens, so the first step is to select the target type of token for the provided seed file.
4. Click the **OK** button, then select the file to import. Notice the **Seed Signature** field does not contain the actual seed value itself. Instead the field contains a hash or signature of each seed. The value is useful to determine if each seed is different (signatures will vary).

Once the file is imported the token list will be populated and the tokens awaiting assignment to specific users.



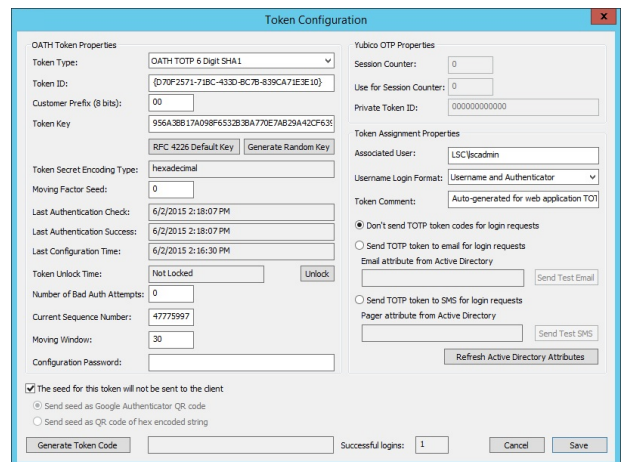
5. Select an entry and click **Edit**.



6. Specify the **Associated User**. The associated user is the user name (login name) that will be used to enter the web application. If using an explicit account, enter the name as **UserName**. If entering a domain or directory user, enter the name as **DirectoryNameUserName**.

7. Click **Save**.

For a description of each field, see the parent section, "[OATH Token Configuration](#)" on page 344.



## OATH Without Existing Tokens

If it is desired to leverage only the infrastructure built into Privileged Identity, then just begin by adding tokens.

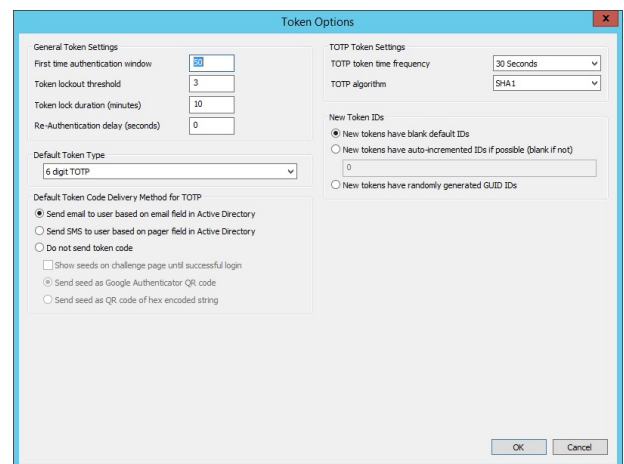
Select **Delegation | OATH/Yubico Token Configuration**.

Default token options can be configured by opening **Global Token Options | Options**. This dialog allows configuration of global options for both event (HOTP) and time-based (TOTP) tokens. These configurations only affect OATH and Yubico type tokens. Most of these settings can generally be left in their default state unless there are specific policies that require changes. Read the functions of each global token parameter to evaluate if these values need to change.

For a description of each field, see below.

Privileged Identity can make the process for user enrollment even easier by also generating QR codes that can be scanned on their smart device or delivering them the authenticator string on screen the first time they try to use OATH enrollment. To enable this functionality, set the Default Token Code Delivery Method for TOTP to **Do not send token code** then select the from following options:

- **Show seeds on challenge page until successful login** - This will show the QR code on the login page until they have successfully logged in at least once with OATH.
- **Send seed as Google Authenticator QR code** - Will display a QR code on the challenge page which may then be captured by a camera device on the user's mobile device in their authenticator program.
- **Send seed as QR code of hex encoded string** - Will provide a hex string which may be copied or typed into the user's authenticator app.



Google Authenticator QR Code:



If the QR code must be resent to the user at a later time, simply edit their token in the console and clear the check box next to **The seed for this token will not be sent to the client**.

## General Token Settings

- **First time authentication window (default: 50)** - We calculate the expected token value based on either time of events (token button presses). Time drift in tokens as well as inadvertent button presses on tokens may change the presented token value. This setting allows the program to search for the token value within a range of calculated values to determine how far the token is from the expected value and then set an internal offset correction going forward. For time-based tokens, and using the default of "50",



the value above is split in half and the token will be searched 25 x 30 seconds before the ideal time, and 25 x 30 seconds ahead of the current time. This is roughly to allow for a 15 minute drift test the first time the token is used. For event tokens, we will search up to 50 key presses from the starting point of a token (offset of zero). Once the token has been synced up with the program (drift/offset determined), the window for the token is narrowed to the specific window set individually for each token. See token configuration for the details of how much drift is allowed after token synchronization.

- **Token lockout threshold (default: 3)** - This value determines how many times a bad token can be submitted before a forced lockout goes into effect. Forced lockouts are generally not permanent, but are time-based and designed to protect against brute force attacks of the token. Locked out tokens may be reset manually via the token management screen.
- **Token lock duration (minutes) (default: 10)** - After the token lockout threshold has been reached, no more tokens for the specific user will be accepted (good or bad) until the lockout duration has passed. Locked out tokens may be reset manually via the token management screen.
- **Re-Authentication delay (seconds) (default: 0)** - To protect against reverse engineering the token seed value (very hard to do under any circumstances), the Re-Authentication delay is designed to slow down a potential attacker trying out a sequence of correct keys in rapid success to determine the value of the token seed used. This feature is primarily to protect against a theoretical attack vector, but this value may be used if this threat is considered to be probable (i.e. attacker has an actual list of seed files to test).

## TOTP Token Settings

Time-based tokens known as TOTP tokens, have a starting point in time (configured on a token basis), but tokens usually start at the Unix Epoch of Jan 1, 1970 UTC. The token is also initialized with a unique random number seed, an algorithm is chosen (generally SHA1), number of digits in the display decided, and the frequency how often the value changes in the token (generally 30 seconds). This dialog allows defining the global values for TOTP type tokens.

- **TOTP token time frequency (default: 30 seconds)** - For time-based tokens, this is the amount of time the token stays on a specific value before moving to the next value. This value is critical to determine the correct token value for a time-based token. 30 seconds is the most common value, but 60 seconds is also a normal value. The manufacturer of tokens and their customer may decide what value to use for the tokens they purchase.
- **TOTP Algorithm (default: SHA1)** - The common algorithm for both time and event-based tokens is known as SHA1. Time-based tokens may also use algorithms such as SHA256 and SHA512. Given the rarity of use for anything other than the SHA1 algorithm, we currently only supports SHA1. If it is desired to use or are currently using SHA256 or SHA512, please contact our support department and we will provide an updated version to support these other hashing algorithms.

## New Token IDs

- **New tokens have blank default IDs** - No token ID will be assigned to new tokens. This is the best option if the desire is to control every aspect of the token creation process.
- **New tokens have auto-incremented IDs if possible (blank if not)** - Starting from the integer provided in the field, token IDs will be generated in one-step increments from that point such that every token will have a unique ID. When a user logs into the web site for the first time and is required to use OATH authentication because a group or role that the user belongs to is requiring OATH authentication, this options will auto-generation and assign a token to that user using the next available ID. Also, when tokens are simply added one by one, the Token ID will be assigned the next available integer.
- **New tokens have randomly generated GUIDs** - Starting with a randomly generated GUID, token IDs will be generated randomly such that every token will have a unique ID. When a user logs into the web site for the first time and is required to use OATH authentication because a group or role that the user belongs to is requiring OATH authentication, this options will auto-generation and assign a token to that user using the next available ID. Also, when tokens are simply added one by one, the Token ID will be assigned another randomly generated GUID.

## Default Token Type

- **Default Token Type** - Select from the available token types for the default token type that will be assigned when new tokens are created. Token type for the assigned user may be changed at any time post-creation.

## Default Token Code Delivery Method for TOTP

- **Default Token Code Delivery Method for TOTP** - When a token of type TOTP is selected as the default token type, these settings define which method of delivery will be used to send the identity their login token when they attempt to access the web site.

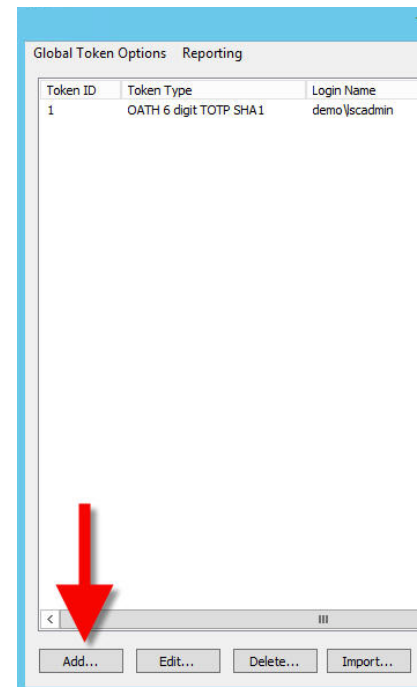
## Automatic Token Enrollment

Privileged Identity will automatically enroll users if OATH authentication is required for the user and no current token exists for the user. Global Token Options, noted above, must be configured and the **New Token IDs Settings** must be set to **New tokens have randomly generated GUID IDs**.

When a user who is set to require OATH authentication attempts to login to the web application for the first time, they will be enrolled at that time with no further interaction required.

## Manually Adding a New Token

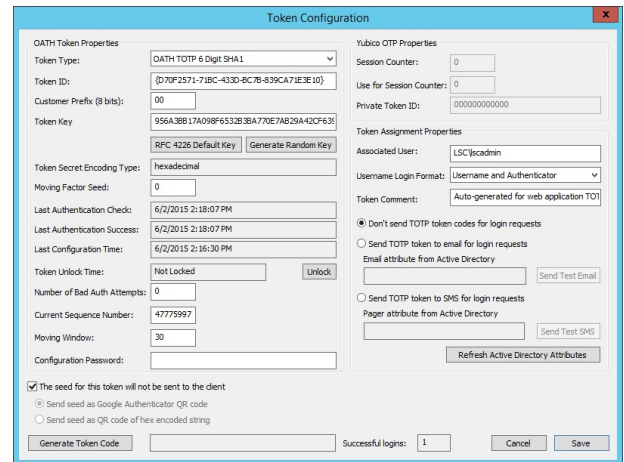
1. To add a new token, Click **Add**.



2. Fill in the required fields:

- Select the appropriate token type to use.
- Supply a token ID. You may use online GUID generators or simply supply a unique hexadecimal number.
- For **Token Key**, click **Generate Random Key**.
- If the token key will not be sent to the user directly (e.g. email), clear the check box for **The seed for this token will not be sent to the client** and then select the appropriate way to make the seed available to the client via web browser. Google Authenticator QR code is the most used option.
- Assign the associated user.
- Make any other edit required.

3. Click **Save**.



For a description of each field, see "[OATH Token Configuration](#)" on page 344.

# OATH Token Configuration

Support for OATH token authentication is available for the web application and management console. No further infrastructure is required for support of this two-factor authentication method when using TOTP tokens; HOTP may require additional elements.

Support for the web application and web service requires the following:

- Web application support for OATH/Yubico MFA is enabled. See the Installation Guide for more information.
- The user is enrolled with an OATH token configuration in the management console at **Delegation | OATH/Yubico Token Configuration**.
- The identity is configured to **Require OATH/Yubico** in the global delegations at **Delegation | Web Application Global Delegation Rules**.

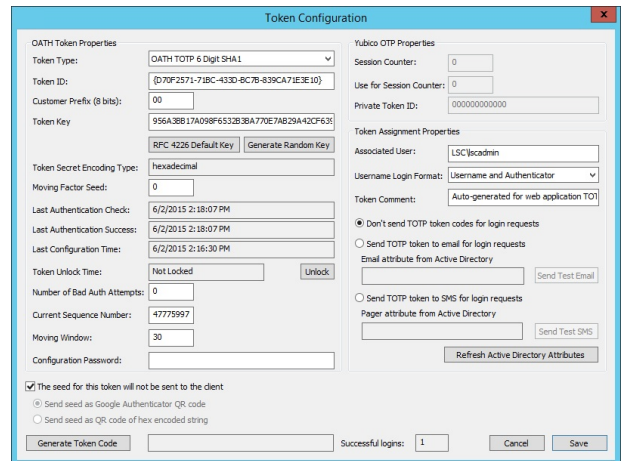
Privileged Identity provides support for seven different OATH token types:

- OATH HOTP 6 Digit SHA1
- OATH HOTP 8 Digit SHA1
- Prefixed OATH HOTP 6 Digit SHA1
- Prefixed OATH HOTP 8 Digit SHA1
- OATH TOTP 6 Digit SHA1
- OATH TOTP 8 Digit SHA1
- Yubico OTP

The one-time-password, or OTP, authentication method can be divided into two sub-types. Time-based methods rely on the transformation of a shared secret and a time value that is synchronized between the server and the client. Event-based methods rely on the transformation of a shared secret and an event count that is synchronized between the server and the client. Typically, the event that is counted is the pressing of a button on the token. HOTP Tokens rely on the event driven model. For example, a user with a key-fob or soft-ement on their smart-phone presses a button. That device or software is in sync with the OATH server (Privileged Identity) such that when they press the button, the code generated is the same code that Privileged Identity comes up with. TOTP tokens rely on a time driven model.

Following is a description of the token assignment dialog and its fields.

Select the appropriate token type to use; see the top of this section, OATH Token, for descriptions of each token type. The token type that is selected will determine which options are available. The following text will describe each token option. Once a token is configured and assigned, click **Save**.



## OATH Token Properties

- **Token Type** - Select the token type and number of digits for the token. Both OATH and Yubico formats are supported. OATH tokens come in an event-based version known as HOTP and a time-based version known as a TOTP. Yubico tokens only come in an HOTP version.
- **Token ID** - This is an 8 digit code that must be unique between tokens. This value is frequently auto-generated by token manufacturers to identify one token from another.
- **Customer prefix (8 bits)** - This is a single byte value that allows token vendors to further qualify tokens by customer. Note that some tokens have the ability to transmit their Token ID and Customer Prefix as well as the token code itself as a long string of digits (i.e. Yubico). In this case, the program will attempt to verify not only the token code (6/8 digits), but also the Token ID and Customer Prefix value. If using a token that only presents 6 or 8 digits (non USB device), then the preceding two fields are not verified and are used for internal accounting only.
- **Token Key** - The Token Key field is used to provide either a 160-bit key for OATH tokens or 128-bit key for Yubico tokens. The field uses decimal encoding where each two digits represent a byte (8-bits) as two hexadecimal digits (00-ff). For an OATH token of 160 bit, this is represented by 40 digits (20 bytes x 8 bits). The OATH token is a random number seed feed into the SHA1 algorithm defined by OATH. Yubico tokens place an AES 128-bit key in this field. This key is unique to each key and is used to decrypt the payload of the Yubico token. Yubico does not present a token value per se, instead it encrypts a token USB insertion count (Session Count) and a key press count during the current session. Note that Yubico has additional user fields for verification. Since the token cannot be decrypted with the wrong AES key, the token is secure. Note that Yubico is an event token only (HOTP).
- **RFC4226 Button** - Pressing this button creates a special test Token Key that is used to confirm the OATH compatibility of HOTP tokens:

```
"3132333435363738393031323334353637383930"
```

The seed created is nothing more than the byte equivalence of the ASCII character sequence of "12345678901234567890". In practice, the first few token values for this test token are well known and can be used to make sure that hardware or software is creating the correct token code sequences.

Example: 6 digit RFC4226 sequences:

```
755224  
287082  
359152
```

Example: 8 digit RFC4226 sequences:

```
84755224  
94287082  
37359152
```

- **Generate Random Key button** - This button generates a series of random numbers to create either a 160-bit seed for OATH format tokens, or a 128-bit crypto key for Yubico tokens. When creating tokens for production use, always use the Generate Random Key option or use import seed files generated by a reliable source. Do not create seeds manually by hand as it is essential that the values be completely random.
- **Token Secret Encoding Type** - Encoding type used for the seed. Valid values are decimal or base64.
- **Moving Factor Seed** - When an OATH HOTP is first programmed, the first token code generally assumed start from zero offset in the SHA1 cryptographic sequence. Some vendors of tokens will start the token off with a random number of initial key press clicks. The thought is that if someone were to get the seed for a token, they would not know where the sequence began (if other than zero). Some token customers feel more comfortable starting tokens at random points in the random sequence generator, so we provide support for the function even it does not provide any significant improvement in security. Generally it is assumed that OATH seeds are secure and there is no need to start the token off at a random starting point in its key press history (the first key press is offset or moving factor = 0).

- **Last Authentication Check** - The last time the user attempted authentication using their token.
- **Last Authentication Success** - The last time the user successfully authenticated using their token.
- **Last Configuration Time** - Last time the token configuration was updated.
- **Token Unlock Time** - If the user has locked themselves out due to failed login times, this is the time at which time they will be unlocked. To unlock the user now, click the **Unlock** button.
- **Number of Bad Auth Attempts** - The number of times a user can fail authentication before being locked out. A value of 0 means lockout immediately.
- **Current Sequence Number** - This is the current number of key presses recorded against this event token (HOTP). This number starts at zero with a new token and is updated to reflect the last matched key press sequence number. As an example: a brand new token will start with a sequence number of zero. If a user presses the button four (4) times to play around with the token, if they then log in with the fifth (5th) key press on the token, the program will detect that the fifth token code was detected and to expect and accept only token code number six (6) and later. The Current Sequence Number is required to make sure that a user does not reuse an old or previous token code.
- **Moving Window** - Given that a user may inadvertently press the token code, the software needs to account for this and look forward from the last token code. In the default case, the program will accept up to the next 10 token codes from the last one that it successfully authenticated against. In the case of time-based tokens (TOTP), this value is split in half to look for tokens 5 steps back in time and 5 steps forward in time, where time is the current time. Note that the program is smart enough to know to not allow the use of time token code values older than the last one correctly authenticated.
- **Configuration Password** - Yubico specific token configuration password to protect token reconfiguration.
- **Generate Token** - Button to test token creation and login.
- **The seed for this token will not be sent to the client** - If the check box is cleared, Privileged Identity will display a QR code on the OATH login challenge page for the user. The user may scan code this with their smart phone to configure to configure their client automatically.

## Yubico OTP Properties

Yubico tokens support both the open OATH standard as well as its own proprietary standard called Yubico OTP. This section maintains the current status information if the user has selected and configured a Yubico OTP token. The Yubico native format passes a long stream of encoded characters (called modhex) that contain token identification information, number of times plugged in and the number of times the button was pressed in the current plugged in session.

When using Yubico OTP, all of the token configuration information must match what is stored within this program. Each time the token is used, the data stream is decoded and the token fully exposes its token identification, the number of times the unit has been plugged in, and the number of times that the button has been pressed. This program checks to make sure that the token information is correct for the current user and the user has not tried to replay a previous string of digits.

Yubico tokens can be delivered fully programmed from Yubico with a seed file ready for import into this program, or blank Yubico tokens can be programmed as OATH or Yubico OTP format.

- **Session Counter** - Last count received from token indicating how many times the token has been plugged into a USB port. The device accumulates an internal counter of how many times it has been plugged into a USB port. For a new token this value will be zero. When the Yubico OTP authenticates successfully against this system, this internal information is decoded, stored and displayed.
- **User for Session Counter** - This is the count of the number of times the button on the Yubico token was last pressed when plugged in and it authenticated with this application. Both of the two previous parameters are normally set to zero for a new token. Once a token has been decoded for the first time, the above two parameters will be set to the discovered values. Only tokens with values of the previous that are greater than the previous usage are accepted. Note that neither of these values can be determined if this program does not have the correct AES key for the token. The encryption key should be different between tokens.

- **Private Token ID** - The string of digits returned contains both clear text and encrypted data. When the token information is decrypted the token ID field identifies the token with an 8 digit number. This field may be pre-configured by the manufacturer, or may be set by the customer when programming blank tokens in the Yubico OTP standard format.

## Token Assignment Properties

One easy and quick way to deploy multi-factor authentication is to forgo the use of physical tokens and send users the current token value using an out-of-band communication channel such as email or SMS/text messaging. With this option it is possible to use either the Active Directory email address or pager information for a user to implement multi-factor authentication to send a time-based token value that has a limited lifetime of usage (defaulted to 15 minutes to use the token before it expires).

The prerequisites to using this feature:

1. Configure user tokens for TOTP 6 or 8 digits, generate a random seed, and properly associate the account and its type to the token.
2. Configure the option to send either email or SMS tokens
3. Email account must be defined in Active Directory or Active Directory has defined the Pager field for a user to contain the email alias for their SMS/text messaging device. Most pagers and cell phone providers provide an email alias for all phones on their network. The format is generally: `PhoneNumber@CarrierGateway.example.com`. See Wikipedia for SMS Gateways to get an updated list of email gateways for carriers worldwide. These user parameters can be set manually using Active Directory administration tools for Users and Computers.
4. If sending email to a domain outside of the corporate network, it may be required to configure the program's email configuration to use authenticated email access (to enable email relay). Alternatively, install our standalone email server: SMTP Express (available free from our web site for use with our products) and configure this program's email configuration to use SMTP Express as a local email gateway to relay email and SMS messages.
  - **Associated User** - The user name (login name) that will be used to enter the web application. If using an explicit account, enter the name as **UserName**. If entering a domain or directory user, enter the name as **DirectoryName\UserName**.
  - **User Login Format** - If the associated user is an explicit account (entered as **UserName**), the select **Username Only**. If the associated user is a user from a directory (entered as **DirectoryName\UserName**), select **Username and Authenticator**.
  - **Token Comment** - A comment for the user/token
  - **Email options** - HOTP tokens have nothing to email users as they rely on an external mechanism/device that is kept in sync with the Privileged Identity server. TOTP token users must be sent their token. This can happen via an email or SMS message using attributes as found in Active Directory.

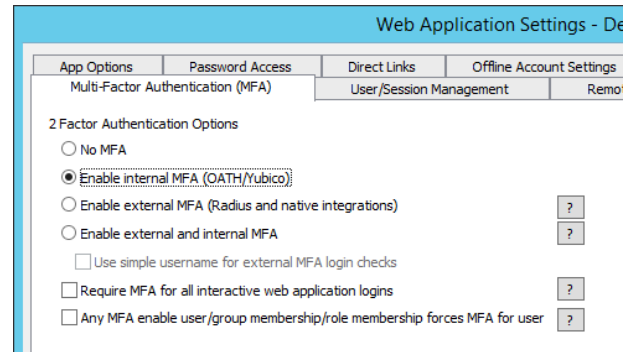


## Configure OATH for Web Client Access

For OATH MFA to work, a user must be enrolled for an OATH token (this can happen automatically), the web application must be configured to use Internal MFA and the identity must be configured with the proper delegation.

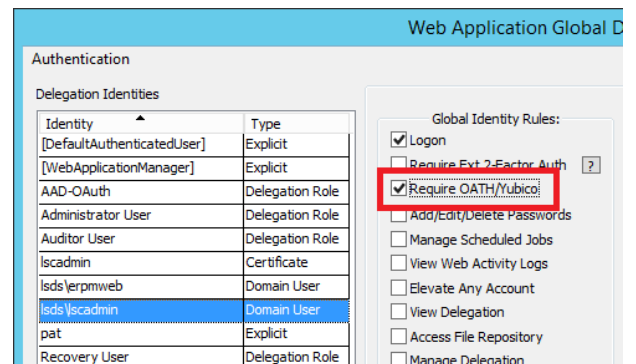
### Configure the Web Application Settings

For a user to leverage OATH MFA, the web application must be configured to support OATH-based authentication. To enable OATH checks within the web site, go to the **Security** tab within the **Web Application Options**. Select the option to **Enable Internal MFA (OATH/Yubico)**.



### Configure the Identity Delegations

1. To add OATH access checks to users of the web application and web service, go to **Delegations | Web Application Global Delegations** dialog.
2. Select the identity and enable **Require OATH/Yubico**.





## Additional OATH Resources

### Oath Specifications

#### Web site for OATH - Initiative for Open Authentication:

<https://openauthentication.org/>

#### Membership list for OATH:

<https://openauthentication.org/members/>

#### Official specs for OATH tokens:

HOTP: <http://www.ietf.org/rfc/rfc4226.txt>

TOTP: <http://tools.ietf.org/html/rfc6238>

### Overview - Yubikey

#### Yubico Web Site:

<http://www.yubico.com/>

#### Cross platform Software tools to program blank Yubico tokens (both OATH and Yubico OTP programming are supported in the same tools):

<https://www.yubico.com/products/services-software/download/yubikey-personalization-tools/>



**Note:** *There are tools to program the tokens one-by-one and also tools to do the programming of a lot of key at the same time (bulk programming).*

#### Yubico Store to Purchase Tokens:

<https://store.yubico.com/>

#### Specification for Yubikey OTP mode:

<https://docs.yubico.com/yesdk/users-manual/application-otp/yubico-otp.html>

Yubico's Yubikey also has its own native encoding and decoding standards. Rather than using a SHA1 and an attempt to figure out which token sequence is received, the Yubico token encrypts everything and provides a unique token id, session count plus the actual click count in the current session. The only thing to keep in mind is that the session/click count must always be greater than the previous one recorded, otherwise it is a replay and is ignored.

## DUO via RADIUS

Privileged Identity can leverage DUO multi-factor authentication. To make use of DUO for MFA, you will use the DUO RADIUS Proxy and it must be configured as a RADIUS Authentication Server entry and as a RADIUS MFA server.

### Configuring the DUO System

The DUO server has to be a proxy server and using **[duo\_only\_client]** settings as described in the DUO documentation here:

[https://duo.com/docs/authproxy\\_reference#ad\\_client](https://duo.com/docs/authproxy_reference#ad_client)

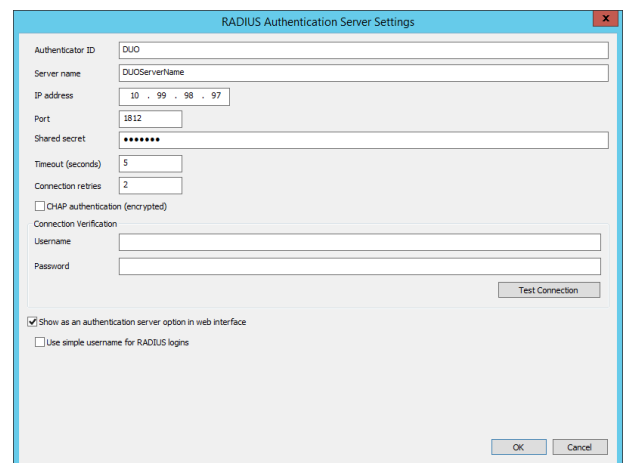
Set DUO to use RADIUS Duo Only using the Duo documentation here:

<https://duo.com/docs/radius>

### Configuring Privileged Identity to use DUO MFA

1. Open the management console and go to **Delegation | Authentication Servers**.
2. Click **Add RADIUS**.
3. Supply the following information:

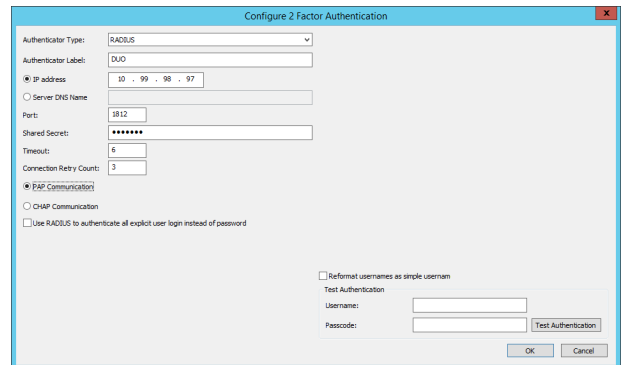
- **Authenticator ID** - This is the name that will be shown in the authenticators list during a web application logon. Valid characters are 0-9, a-z, and A-Z.
- **Server name** - The name of the RADIUS server as identified in DNS.
- **IP Address** - IP address of the RADIUS server. This will be the same IP entered below when configuring the MFA entry.
- **PORT** - The port of the RADIUS server.
- **Shared Secret** - The shared secret configured for your client in RADIUS. This will be the same password entered below when configuring the MFA entry. Be sure to make the web application server(s) are RADIUS clients on the RADIUS server!
- **Timeout (seconds)** - Default value is 5. This is the number of seconds to wait for a connection or reply to or from the RADIUS host.
- **Connection retries** - Default value is 2. The number of connection attempts to make to the RADIUS server before returning a connection error.
- **CHAP Authentication** - Leave CHAP authentication unselected.



4. Enable the option to **Show as an authentication server option in web interface**.
5. Click **OK** to add the RADIUS authentication server.
6. In the management console go to **Delegation | External 2 Factor Configuration**.
7. Set **Authenticator Type** to **RADIUS**.
8. Provide an **Authenticator Label** (friendly name).

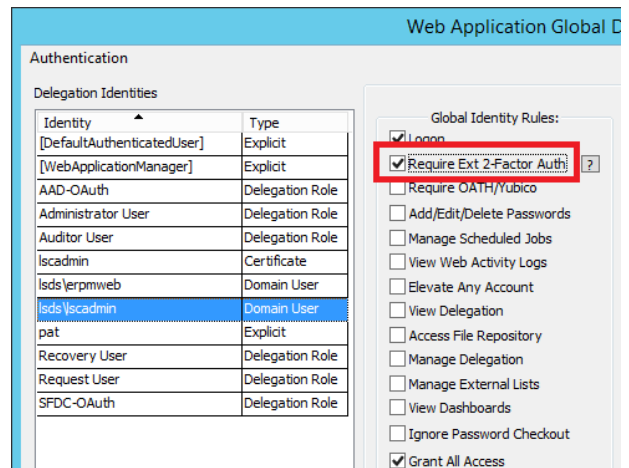
9. Supply the RADIUS server information:

- **IP Address or Server DNS Name** - Supply the IP address or DNS name of the RADIUS server.
- **Port** - The listener port of the RADIUS server. The default port is 1812.
- **Shared Secret** - The shared secret to communicate with the RADIUS server. This will be the same password entered above when configuring the authentication server address. Be sure to make the web application server(s) are RADIUS clients on the RADIUS server!
- **Timeout** - The preferred timeout value for the call to the RADIUS server.
- **Connection Retry Count** - The allowable amount of failures when trying to call the RADIUS server.
- **PAP Communication** - Configure the setting to use PAP communication.



10. Click **OK**.

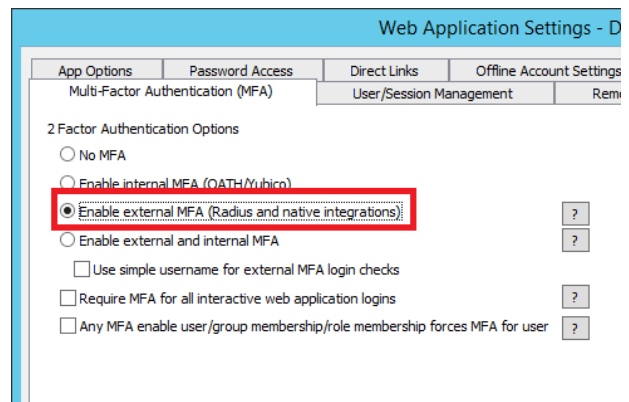
11. Go to **Delegation | Web Application Global Delegation Rules**, select the target identity and enable the option to **Require Ext 2-Factor Authentication**.



## Configure the Web Application Settings

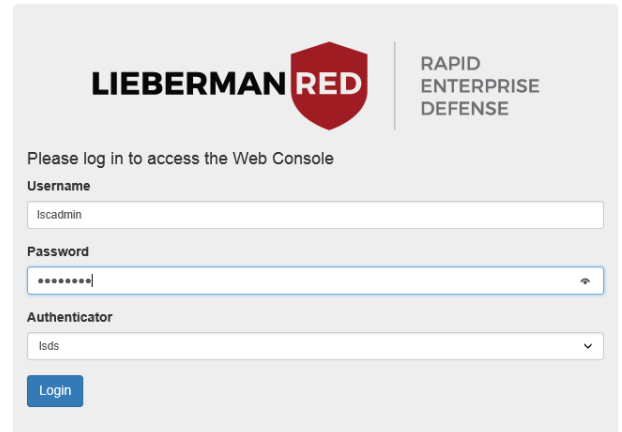
For a user to leverage OATH MFA, the web application must be configured to support OATH-based authentication. To enable OATH checks within the web site, go to the **Security** tab within the **Web Application Options**. Select the option to **Enable external MFA (OATH/Yubico)**.

See the installation guide for more information on Multi-Factor Authentication options.



## Logging in with DUO

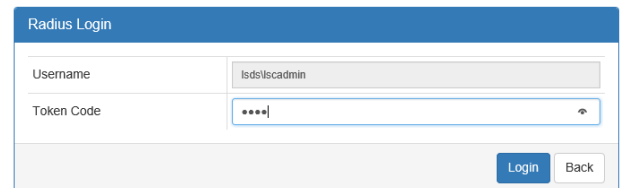
When a user logs into the web site, they will login with their normal username, password, and domain authenticator: do not select the DUO Authenticator entry.



The screenshot shows the Lieberman Red login interface. At the top left is the Lieberman Red logo, and at the top right is the text "RAPID ENTERPRISE DEFENSE". Below the logo, it says "Please log in to access the Web Console". The form contains three main sections: "Username" with a text input field containing "lscadmin"; "Password" with a masked text input field containing "\*\*\*\*\*"; and "Authenticator" with a dropdown menu showing "lscds". A blue "Login" button is located below the dropdown.

After clicking Login, the user will be presented with the RADIUS Login screen. The username field will be populated with the username based on the information presented on the login screen.

In the Token Code field, type the word push, in lower case, and click Login.



The screenshot shows the RADIUS Login screen. It has a blue header with the text "Radius Login". Below the header, there are two input fields: "Username" which is pre-filled with "lscds/lscadmin", and "Token Code" which is masked with "\*\*\*\*". At the bottom right, there are two buttons: a blue "Login" button and a white "Back" button.

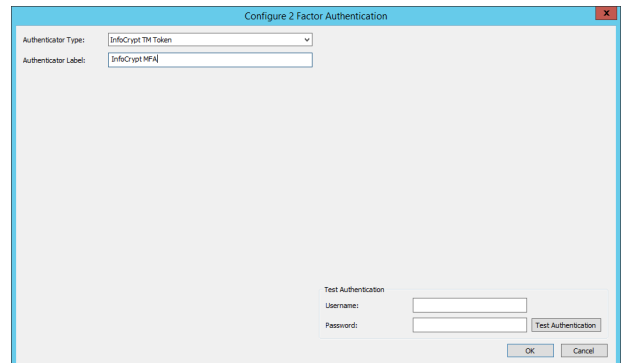
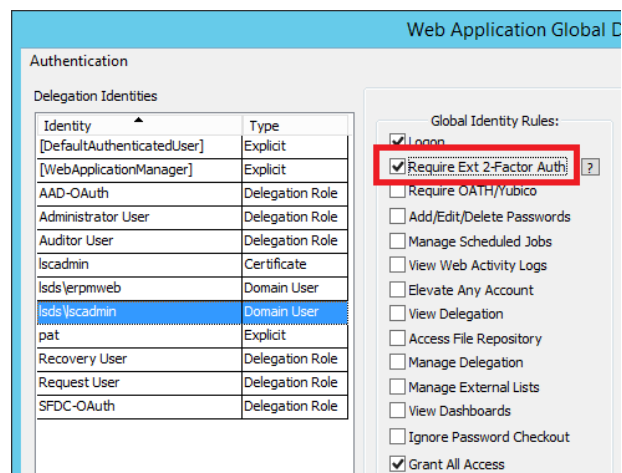
## InfoCrypt

To utilize InfoCrypt 2 factor authentication, the InfoCrypt agent must be installed on the Privileged Identity server which will be taking part in the authentication. If the web application is installed on a separate server, then that server will require the InfoCrypt agent be installed locally. If access to the management console will require InfoCrypt's 2 factor authentication, then the InfoCrypt agent must be installed on the machine hosting the management console. In order to install the agent, it is required to run InfoCrypt's installation package, get a copy of the shared secret from the target InfoCrypt authentication server, and point the agent to the authentication server during the installation. These steps are all covered by the InfoCrypt agent installation/usage guide.

## Configuring Privileged Identity for InfoCrypt

Install the InfoCrypt MFA Agent before proceeding.

1. In the management console go to **Delegation | External 2 Factor Configuration**.
2. Set **Authenticator Type** to **InfoCrypt TM Token**.
3. Provide an **Authenticator Label** (friendly name).
4. Use the **Test Authentication** option to test the integration.
5. Go to **Delegation | Web Application Global Delegation Rules**, select the target identity and enable the option to **Require Ext 2-Factor Authentication**.

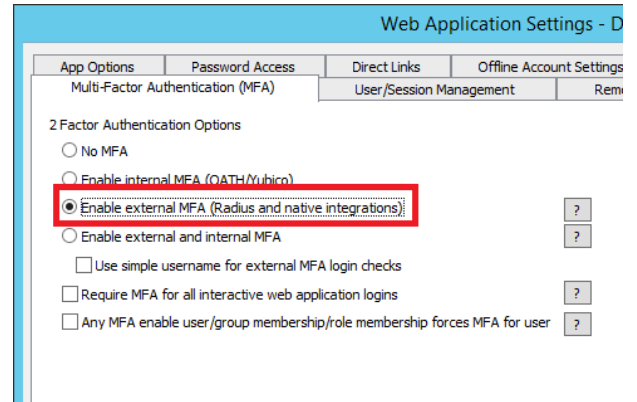



Identity	Type
[DefaultAuthenticatedUser]	Explicit
[WebApplicationManager]	Explicit
AAD-OAuth	Delegation Role
Administrator User	Delegation Role
Auditor User	Delegation Role
Isadmin	Certificate
Isds\erpmweb	Domain User
Isds\jscadmin	Domain User
pat	Explicit
Recovery User	Delegation Role
Request User	Delegation Role
SFDC-OAuth	Delegation Role

## Configure the Web Application Settings

For a user to leverage OATH MFA, the web application must be configured to support OATH-based authentication. To enable OATH checks within the web site, go to the **Security** tab within the **Web Application Options**. Select the option to **Enable external MFA (OATH/Yubico)**.

See the installation guide for more information on Multi-Factor Authentication options.



## RADIUS 2-Factor

For any other form of two factor authentication where it is preferred to go through a RADIUS server rather than install a local agent or service, use the RADIUS 2-Factor option. The identities may be standard identities (users, groups, etc.) or may be added as RADIUS users.

Vendors such as SafeNet, DUO, RSA, and others may supply this functionality. Configuration of their services for use with RADIUS is covered by the vendor's specific documentation.



**Note:** This RADIUS MFA implementation does not support any advanced features outside of the RADIUS specification that the vendor may otherwise provide through the use of normal MFA agents, for example next token or on demand tokens by RSA.

## Configuring Privileged Identity for RADIUS MFA

Configure the target RADIUS before proceeding and add all Privileged Identity component hosts that will participate in RADIUS MFA as RADIUS clients.

1. In the management console go to **Delegation | External 2 Factor Configuration**.

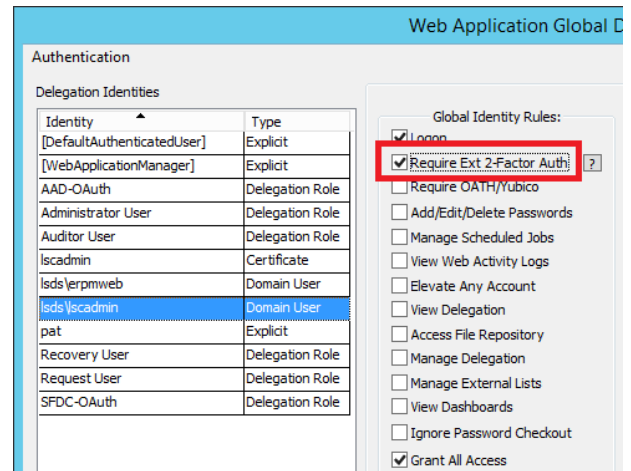
2. Set **Authenticator Type** to **RADIUS**.

3. Provide an **Authenticator Label** (friendly name).

4. Supply the RADIUS server information:

- **IP Address or Server DNS Name** - Supply the IP address or DNS name of the RADIUS server.
- **Port** - The listener port of the RADIUS server. The default port is 1812.
- **Shared Secret** - The shared secret to communicate with the RADIUS server. Be sure to make the web application server (s) are RADIUS clients on the RADIUS server!
- **Timeout** - The preferred timeout value for the call to the RADIUS server.
- **Connection Retry Count** - The allowable amount of failures when trying to call the RADIUS server.
- **PAP / CHAP** - The preferred encryption method for the RADIUS server.

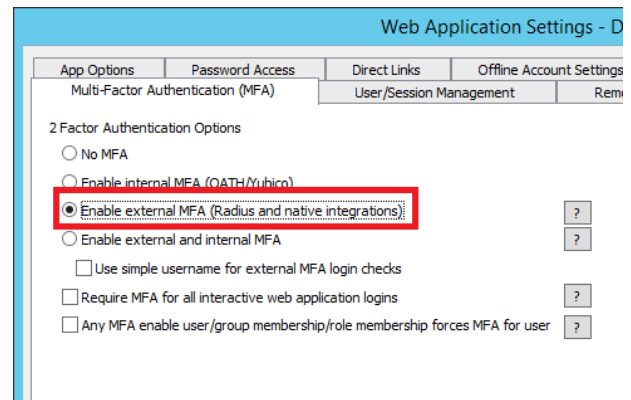
- Go to Delegation | Web Application Global Delegation Rules, select the target identity and enable the option to **Require Ext 2-Factor Authentication**.



## Configure the Web Application Settings

For a user to leverage OATH MFA, the web application must be configured to support OATH-based authentication. To enable OATH checks within the web site, go to the **Security** tab within the **Web Application Options**. Select the option to **Enable external MFA (OATH/Yubico)**.

See the installation guide for more information on Multi-Factor Authentication options.





## RADIUS 2-Factor for Explicit Accounts

Privileged Identity supports the use of explicit accounts. These are accounts that do not exist anywhere but in the context of Privileged Identity. They are not beholden to normal directory policies for password complexity, aging, or history. Accounts like this are used most often when there is no central directory to rely on. To help improve security for these accounts, it is possible to require two-factor authentication for these accounts.

For any other form of two factor authentication where it is preferred to go through a RADIUS server rather than install a local agent or service, use the RADIUS 2-Factor option. The identities may be standard identities (users, groups, etc.) or may be added as RADIUS users.

Vendors such as SafeNet, DUO, RSA, and others may supply this functionality. Configuration of their services for use with RADIUS is covered by the vendor's specific documentation.

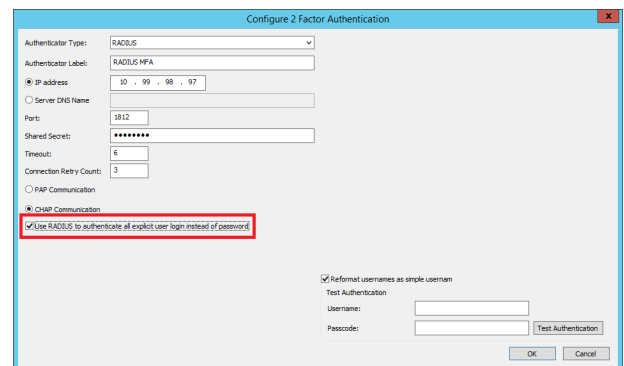


**Note:** This RADIUS MFA implementation does not support any advanced features outside of the RADIUS specification that the vendor may otherwise provide through the use of normal MFA agents, for example next token or on demand tokens by RSA.

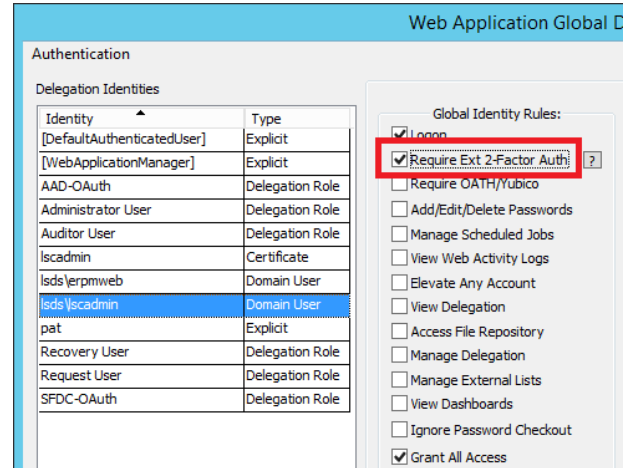
## Configuring Privileged Identity for RADIUS MFA

Configure the target RADIUS before proceeding and add all Privileged Identity component hosts that will participate in RADIUS MFA as RADIUS clients.

1. In the management console go to **Delegation | External 2 Factor Configuration**.
2. Set **Authenticator Type** to **RADIUS**.
3. Provide an **Authenticator Label** (friendly name).
4. Supply the RADIUS server information:
  - **IP Address or Server DNS Name** - Supply the IP address or DNS name of the RADIUS server.
  - **Port** - The listener port of the RADIUS server. The default port is 1812.
  - **Shared Secret** - The shared secret to communicate with the RADIUS server. Be sure to make the web application server (s) are RADIUS clients on the RADIUS server!
  - **Timeout** - The preferred timeout value for the call to the RADIUS server.
  - **Connection Retry Count** - The allowable amount of failures when trying to call the RADIUS server.
  - **PAP / CHAP** - The preferred encryption method for the RADIUS server.
5. Enable **Use RADIUS to authenticate all explicit user login instead of password**.



1. Go to **Delegation | Web Application Global Delegation Rules**, select the target identity and enable the option to **Require Ext 2-Factor Authentication**.

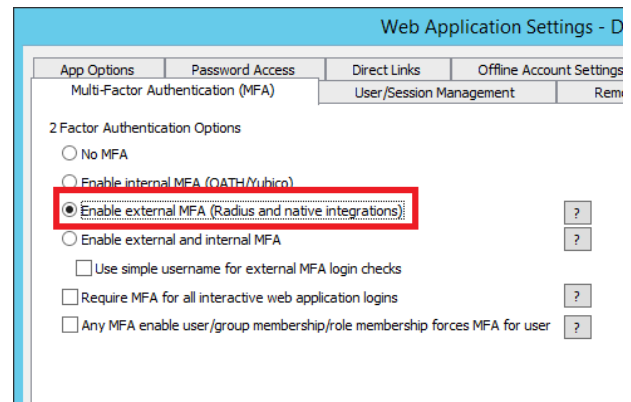


## Configure the Web Application Settings

For a user to leverage OATH MFA, the web application must be configured to support OATH-based authentication. To enable OATH checks within the web site, go to the **Security** tab within the **Web Application Options**. Select the option to **Enable external MFA (OATH/Yubico)**.

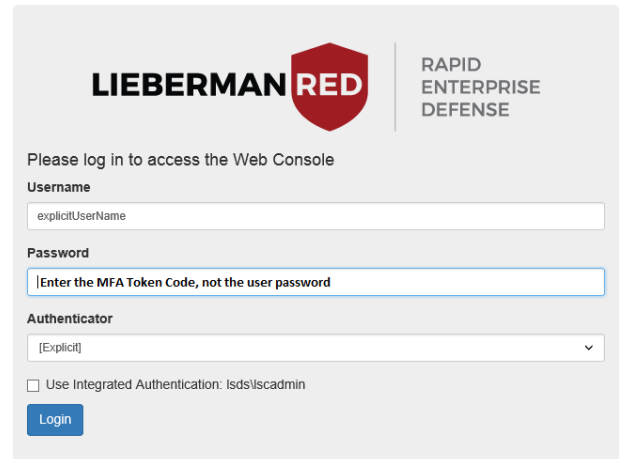
See the installation guide for more information on Multi-Factor Authentication options.

The web application security option to **Use simple user name for external MFA login checks** must be selected. To enable this check box, first select the **Enable external and internal MFA**, then enable **Use simple user name for external MFA login checks**.



## Explicit User MFA Login

When the explicit user logs in, supply the correct user name, set the Authenticator to [Explicit], then in lieu of the password supply the pin or token number for the account. If the two factor system successfully authenticates the user name, the user will be logged in directly.



**LIEBERMAN RED** | RAPID ENTERPRISE DEFENSE

Please log in to access the Web Console

**Username**  
explicitUserName

**Password**  
Enter the MFA Token Code, not the user password

**Authenticator**  
[Explicit]

Use Integrated Authentication: Isds\iscadmin

Login

## RSA SecurID

Support for RSA SecurID authentication is available for both the management console application and the web application. Privileged Identity supports RSA client agent versions 7 and version 8.

To use RSA via RADIUS, please see "[RADIUS 2-Factor](#)" on page 355 for more information.

Support for the console requires that the RSA SecurID client agent software be installed and configured to talk to a working authentication server prior to enabling the feature. Prior to enabling the support for SecurID in Privileged Identity, install and configure the RSA SecurID authentication server and correctly configure the client agent software on the Privileged Identity host system that users will be authenticating through. This is typically the web application server and/or management console.

The basic premise regarding authentication problems with RSA is this: If the RSA client software on the Privileged Identity host can successfully authenticate the user, the web application should be able to use RSA MFA as well.

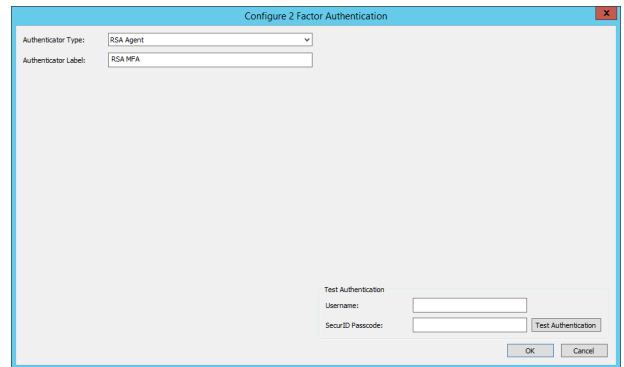


**Tip:** The authentication methodology for the console requires that the logon name for the user accessing the console match the default login name associated with the SecurID token in the SecurID authentication server database. This means that if the token is assigned to a user with a default login name of **DomainName\UserName**, then when logging on to access the console using that SecurID token, log into the machine using the account **DomainName\UserName**. The console and web application can authenticate using either the fully qualified user logon name or the simple logon name (without prefixing the **DomainName**). To change this setting, use the program options page for RSA SecurID settings in the console, and the web application options page for the web application.

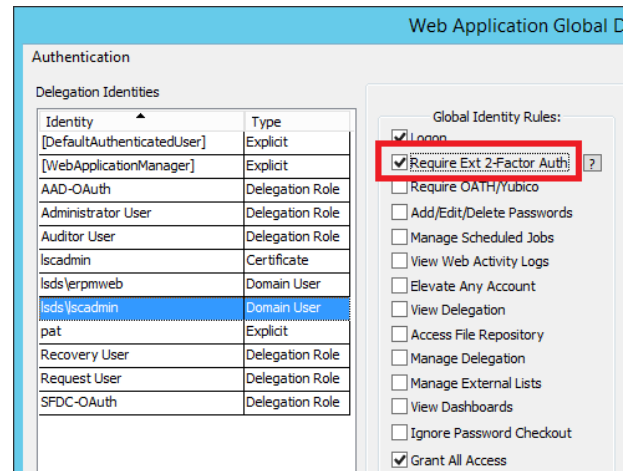
## Configuring Privileged Identity for RSA MFA

Install the RSA MFA Agent before proceeding.

1. In the management console go to **Delegation | External 2 Factor Configuration**.
2. Set **Authenticator Type** to **RSA Agent**.
3. Provide an **Authenticator Label** (friendly name).
4. Use the **Test Authentication** option to test the integration.



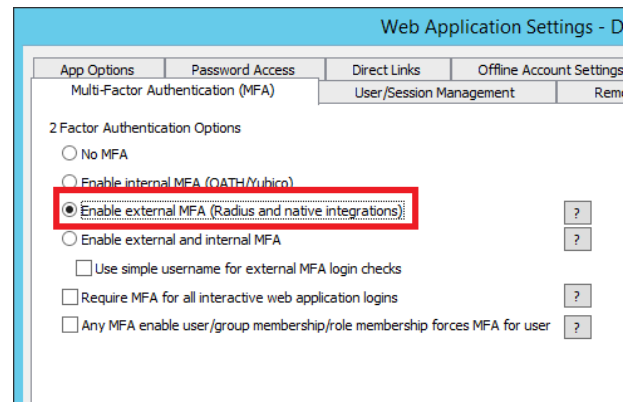
- Go to **Delegation | Web Application Global Delegation Rules**, select the target identity and enable the option to **Require Ext 2-Factor Authentication**.



## Configure the Web Application Settings

For a user to leverage OATH MFA, the web application must be configured to support OATH-based authentication. To enable OATH checks within the web site, go to the **Security** tab within the **Web Application Options**. Select the option to **Enable external MFA (OATH/Yubico)**.

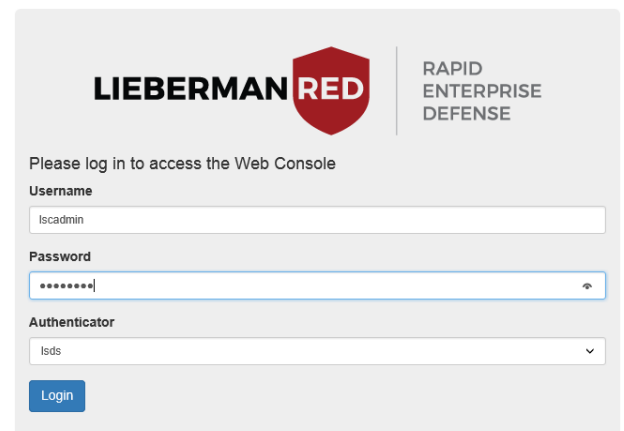
See the installation guide for more information on Multi-Factor Authentication options.



## Logging in with RSA MFA

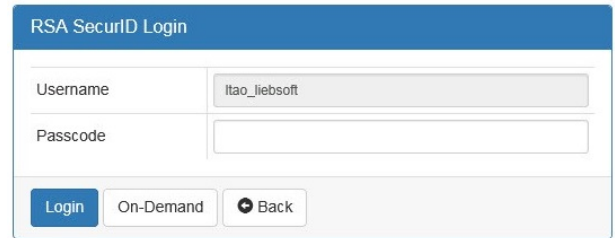
When a user logs into the web site, they will login with their normal username, password, and domain authenticator.

After clicking Login, the user will be presented with the RSA SecurID Login screen. The username field will be populated with the username based on the information presented on the login screen.



If the user is not required to use an on-demand token code, they should supply their current passcode and click Login. If they are required to use an on-demand token, they should click **On-Demand Token Code**. If the PIN was previously initialized they will be logged in. If they must use an on-demand token, one will be emailed to them. The specifics of the any additional steps are outlined below.

If the user is not required to use an on-demand token code, they should just click Login. If they are required to use an on-demand token, they should click **On-Demand Token Code**. If the PIN was previously initialized they will be logged in. If they must use an on-demand token, one will be emailed to them. The specifics of the any additional steps are outlined below.

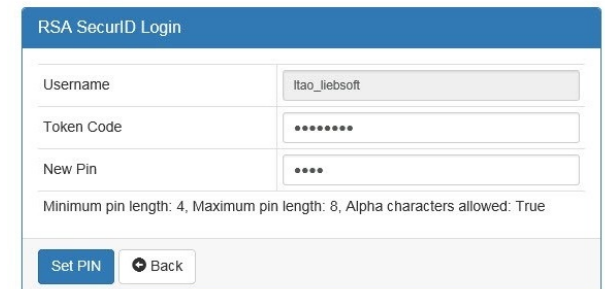


## Uninitialized PIN

Enter the passcode from the SecurID token and then click the **Login** button. This will complete the login process and redirect to the main management page. If the PIN code for the login has not been initialized for the SecurID token, the PIN will need to be initialized. The PIN number can be set up through the web application. The process looks like this:

The specifics of the any additional steps are outlined below.


New PIN required. Please wait until the token code changes, then enter your new PIN with the next tokencode



Enter the next tokencode on your device as well as input a new PIN of your choice. If the PIN is not accepted, there will be a notification and the login screen will reappear. If the PIN is accepted, then supply the next passcode (PIN + tokencode) to complete the login process.

Another possible scenario is that RSA SecurID will require next passcode when attempting to authenticate. In that case, supply the next two passcodes in order to login.

New PIN accepted - please enter the next passcode to login



## On-Demand Token

If the user is configured to use an on demand token, then on the initial RSA screen enter the current PIN into the passcode dialog box and select **On-Demand Token Code**. The user will receive their on-demand token code in their email.

The user will now enter the on-demand PIN into the **PIN** field and enter the token code that was emailed to them into the **On Demand Tokencode** field.

Please enter your PIN and the on-demand tokencode you received

RSA SecurID Login

Username	<input type="text" value="ltao_liebsoft"/>
PIN	<input type="password" value="...."/>
On Demand Tokencode	<input type="password" value="....."/>

Login

↩ Back

## On-Demand Token with Next PIN Set

If the user is configured to use an on demand token and the initial PIN has not yet been initialized, then when the user attempts to login with their RSA PIN, they will be prompted again for the current PIN and to establish a new PIN. Once both PINs are entered, click **Set PIN**.

The user must enter their new PIN that was just set, then click **Get Tokencode**.

The user will receive their on-demand token code in their email.

The user will now enter the on-demand PIN into the PIN field and enter the token code that was emailed to them into the Passcode field.

New PIN required. Please re-enter the current PIN and your new PIN

RSA SecurID Login

Username	<input type="text" value="ltao_liebsoft"/>
Current PIN	<input type="password" value="....."/>
New Pin	<input type="password" value="...."/>

Minimum pin length: 4, Maximum pin length: 8, Alpha characters allowed: True

Set PIN

↩ Back

New PIN accepted - please enter your new PIN to generate an on-demand tokencode

RSA SecurID Login

Username	<input type="text" value="ltao_liebsoft"/>
PIN	<input type="password" value="...."/>

Get Tokencode

↩ Back

Please enter your PIN and the on-demand tokencode you received

RSA SecurID Login

Username	<input type="text" value="ltao_liebsoft"/>
PIN	<input type="password" value="...."/>
On Demand Tokencode	<input type="password" value="....."/>

Login

↩ Back

## Verify RSA SecurID Communication

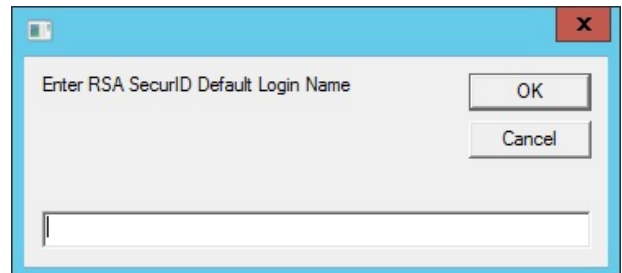
In order for Privileged Identity to be able to authenticate SecurID tokens, the RSA SecurID version 7 or version 8 client software must be installed and configured correctly on the system where the console is running. The client installation can be tested using the test scripts that are distributed with Privileged Identity. In order for the test scripts to run, the ActiveX control named **LiebSoftRSA SecurIDCOM.ocx** must be registered. The file can be found in the program installation directory. This file will be registered automatically during the initial installation if elected at the time of install. During the installation of the web site, the necessary COM application and files are distributed and configured to the target web server. A COM Application called Liebssoft SecureID will be created and configured to run as the same identity running the web application's COM application.

The first test script is named **Rsa.vbs**. It is located in the **\Supplemental2FA\RSA** directory under the program installation directory. This script verifies that the ActiveX object has been registered successfully. When running the script, a message box indicating the version information for the ActiveX control will appear.

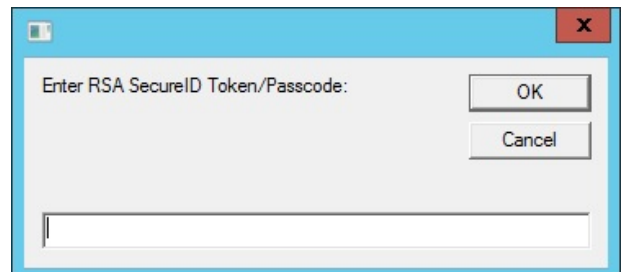
If the version information message box fails to appear, the ActiveX control was not registered correctly.



Once the control is verified as registered, the second test script to test with is **RsaAuth.vbs**. It is also located in the same directory. This script will prompt for a SecurID default login name and passcode and attempt to authenticate the user with the SecurID authentication server. This script will not handle more complex cases like next passcode required or PIN initialization requests, it will only attempt to authenticate a user token that is in the "enabled" state. Running the script should yield the following series of messages boxes. The first will prompt for a default login name.

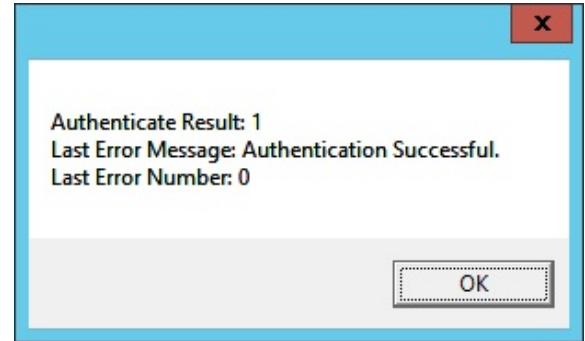


The next message box will ask for the passcode.



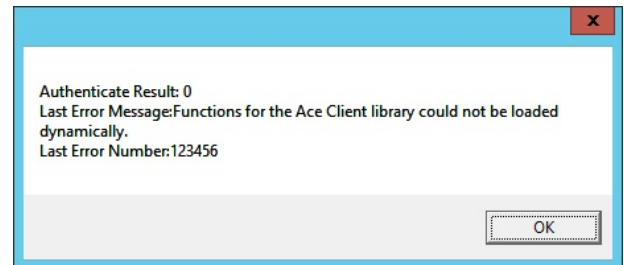
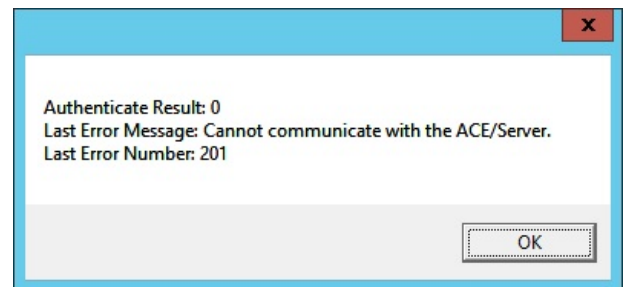


At this point the script will authenticate the token. If the authentication is successful, a message box indicating success will appear:



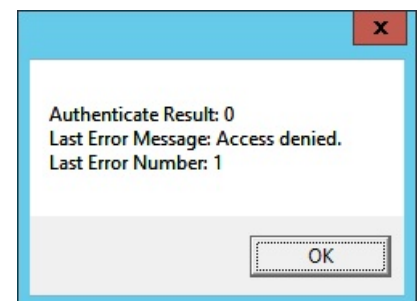
If the operation is not successful, one of three resulting message boxes will appear. The first two possibilities show error messages indicating that communication to the authentication server could not be established or the RSA client is not installed:

If either of these message boxes appear, it means that either the RSA Client software is not installed/configured correctly on the system, the authentication server is not running, the authentication service on the authentication server is down, or the authentication server could not be reached on the network.



The third possible failure message box indicates that there was a problem authenticated the user and passcode pair, but the communication with the authentication server was successful.

The message does not indicate what the specific problem was by design, in order to minimize attackers gaining any information as to what went wrong with the authentication attempt.



## SafeNet

To utilize SafeNet 2 factor authentication, the SafeNet agent must be installed on the ERPM server which will be taking part in the authentication. If the ERPM web site is installed on a separate server, then that server will required the SafeNet agent be installed locally. If access to the ERPM administrative console will require SafeNet's 2 factor authentication, then the SafeNet agent must be installed on the machine hosting the ERPM console. In order to install the agent, you need to run SafeNet's installation package, get a copy of the shared secret from the target Safenet authentication server, and point the agent to the authentication server during the installation. These steps are all covered by the Safenet agent installation/usage guide:

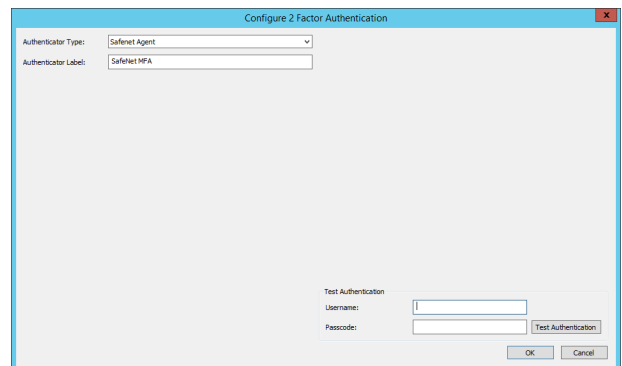
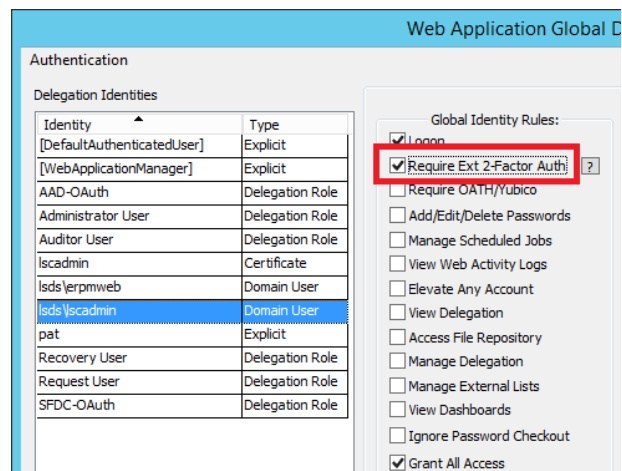
[http://www.safenet-inc.com/resources/integration-guide/data-protection/Safenet\\_Authentication\\_Service/SafeNet\\_Authentication\\_Service\\_Windows\\_Logon\\_Agent\\_Configuration\\_Guide/](http://www.safenet-inc.com/resources/integration-guide/data-protection/Safenet_Authentication_Service/SafeNet_Authentication_Service_Windows_Logon_Agent_Configuration_Guide/)

To use SafeNet via RADIUS, please see "[RADIUS 2-Factor](#)" on page 355 for more information.

## Configuring Privileged Identity for SafeNet Agent

Install the SafeNet Agent before proceeding.

1. In the management console go to **Delegation | External 2 Factor Configuration**.
2. Set **Authenticator Type** to **SafeNet Agent**.
3. Provide an **Authenticator Label** (friendly name).
  
4. Use the **Test Authentication** option to test the integration.
5. Go to **Delegation | Web Application Global Delegation Rules**, select the target identity and enable the option to **Require Ext 2-Factor Authentication**.

Identity	Type
[DefaultAuthenticatedUser]	Explicit
[WebApplicationManager]	Explicit
AAD-OAuth	Delegation Role
Administrator User	Delegation Role
Auditor User	Delegation Role
Isadmin	Certificate
Isds\erpmweb	Domain User
Isds\iscadmin	Domain User
pat	Explicit
Recovery User	Delegation Role
Request User	Delegation Role
SFDC-OAuth	Delegation Role

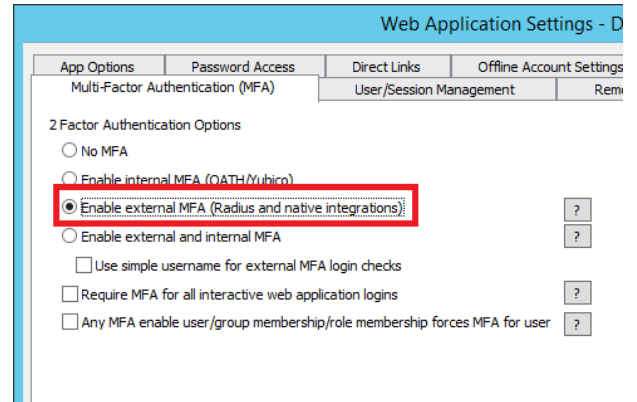
**Global Identity Rules:**

- Logon
- Require Ext 2-Factor Auth** ?
- Require OATH/Yubico
- Add/Edit/Delete Passwords
- Manage Scheduled Jobs
- View Web Activity Logs
- Elevate Any Account
- View Delegation
- Access File Repository
- Manage Delegation
- Manage External Lists
- View Dashboards
- Ignore Password Checkout
- Grant All Access

## Configure the Web Application Settings

For a user to leverage OATH MFA, the web application must be configured to support OATH-based authentication. To enable OATH checks within the web site, go to the **Security** tab within the **Web Application Options**. Select the option to **Enable external MFA (OATH/Yubico)**.

See the installation guide for more information on Multi-Factor Authentication options.



# YubiKey

YubiKey is based on the OATH standards and will use the OATH configuration dialogs in Privileged Identity to enroll its tokens. See "OATH 2-Factor" on page 336 for more information on enrolling the tokens. Keep reading to see the YubiKey specific configurations required in the YubiKey servers.

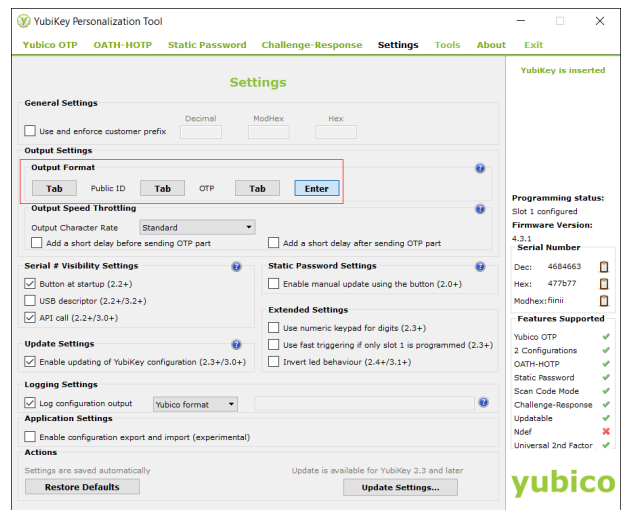
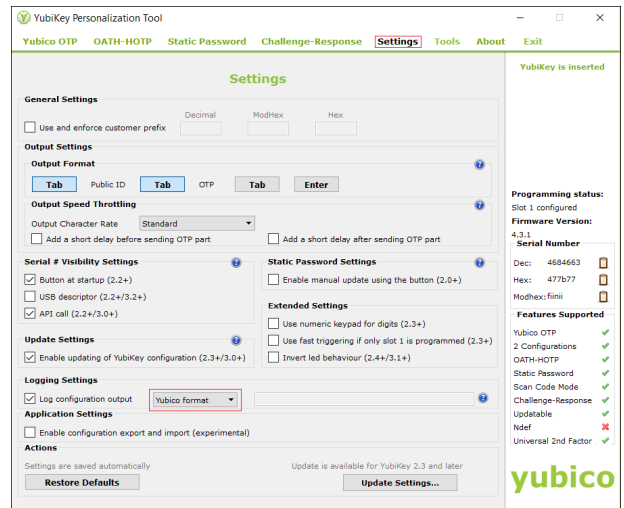
## Using YubiKey to Authenticate to Privileged Identity

To use YubiKey in either OATH-HOTP mode or Yubico OTP mode you will need to download the YubiKey Personalization Tool (preferred) from the Yubico web site here:

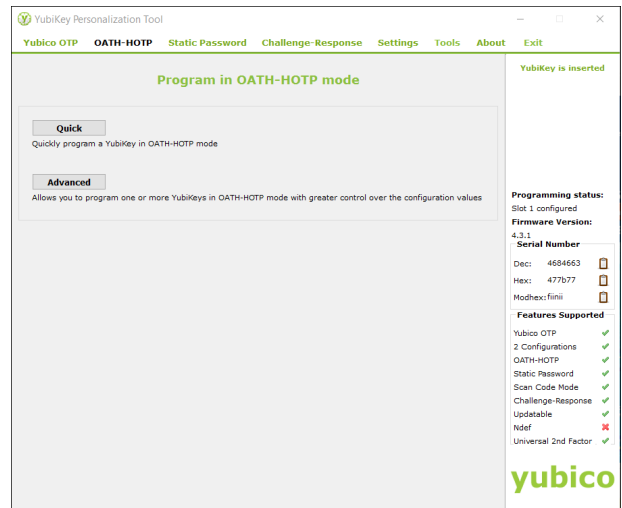
<https://www.yubico.com/support/downloads/>

### To Configure the YubiKey

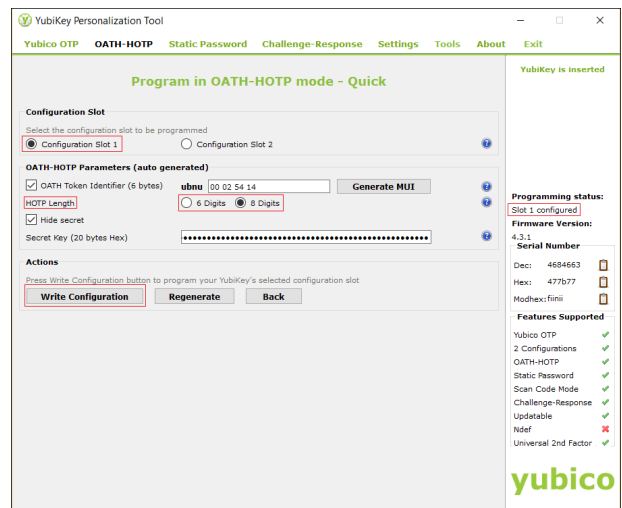
1. Install and open the YubiKey Personalization Tool software, insert the YubiKey into a USB slot, and click **Settings**. The "Settings" screen opens and the status panel (on the right side of the screen) indicates that the YubiKey is inserted.
2. Find the **Logging Settings** section and choose **Yubico format** from the **Log configuration output** dropdown menu.
3. Find the **Output Format** section and, if needed, update it so that only the **Enter** button is selected. (If this section is not configured correctly, the token will be inserted into the wrong field on the **LoginOATH.asp** page.)



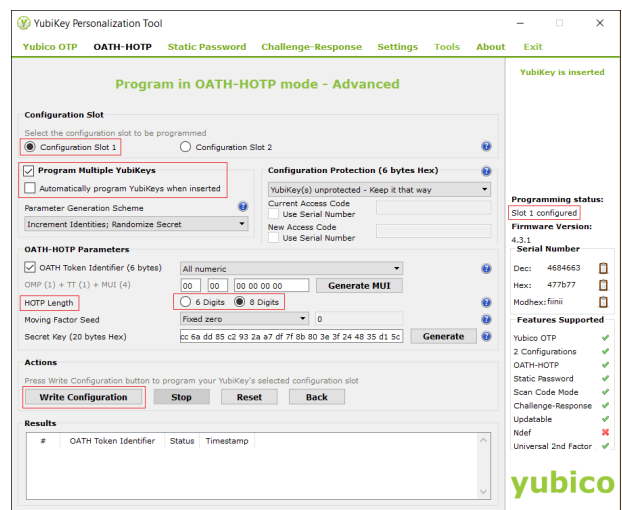
4. Click either **OATH-HOTP** or **Yubico OTP** at the top of the screen. Either the "Program in OATH-HOTP mode" screen or the "Program in Yubico OTP mode" screen opens.
5. If you are personalizing one YubiKey, click **Quick**; if you are personalizing multiple keys, click **Advanced**. One of the following screens opens: "Program in OATH-HOTP mode - Quick" or "Program in OATH-HOTP mode - Advanced."



6. Complete the following if you selected Quick; otherwise, go to the next step.
  - a. In the **Configuration Slot** section, choose the appropriate option. Refer to the status panel on the right side of the screen.
  - b. In the **OATH-HOTP Parameters** section, choose either **6 Digits** or **8 Digits** for the **HOTP length**.
  - c. Click **Write Configuration**. After clicking Write Configuration, the YubiKey Personalization Tool creates the .csv file that you will need to import into Privileged Identity in the next section. Choose the location to save the file.



7. Complete the following if you selected Advanced; if you selected Quick go to the next section.
  - a. In the **Configuration Slot** section, choose the appropriate option. Refer to the status panel on the right side of the screen.
  - b. In the **OATH-HOTP Parameters** section, choose either **6 Digits** or **8 Digits** for the **HOTP length**.
  - c. If you are personalizing multiple keys at one time, select **Program Multiple Yubikeys**. To make the process faster, choose **Automatically program Yubikeys when inserted**.
  - d. Click **Write Configuration**. After clicking Write Configuration, the YubiKey Personalization Tool creates the .csv file that you will need to import into Privileged Identity in the next section. Choose the location to save the file.



## To Import the YubiKey Key Tokens into Privileged Identity

See "OATH With Existing Tokens" on page 338 for steps to import the token import file.

## To Configure an Identity to Require the OATH/Yubico Token

See "Configure OATH for Web Client Access" on page 348 for steps to configure the web application and identity for MFA.

## Use YubiKey Configured as a Smart Card

To use YubiKey as a smart card, you will need to download the YubiKey PIV Manager (with graphic interface) from the Yubico web site here:

<https://www.yubico.com/support/downloads/>

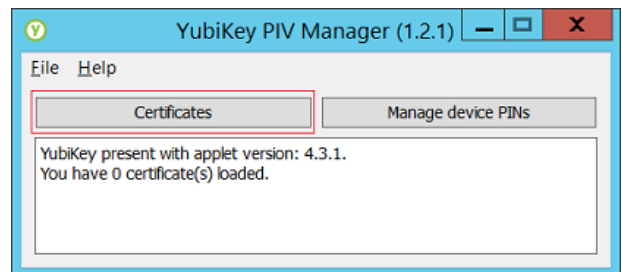
### To Configure the YubiKey as a Smart Card

The following example uses a Windows Enterprise CA to create a user certificate and publish the certificate in Active Directory. These steps also assume that a Smartcard User template is available to the user running this process from that CA. Your steps will vary if sending to an offline or commercial CA.

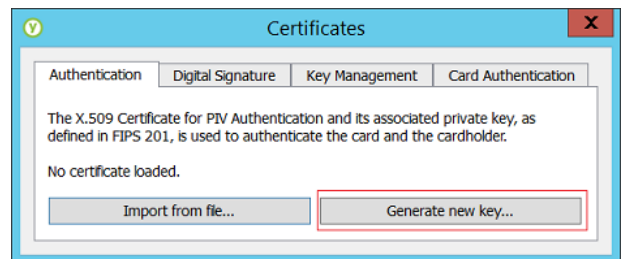
Please consult your Yubico specialists and Yubico documentation for in depth help and guidance.

1. Install the YubiKey PIV Manager software, insert the YubiKey into a USB slot, and click **Certificates**.

You are prompted to create a PIN for the YubiKey. You can change this PIN at any time by clicking "Manage device PINs."

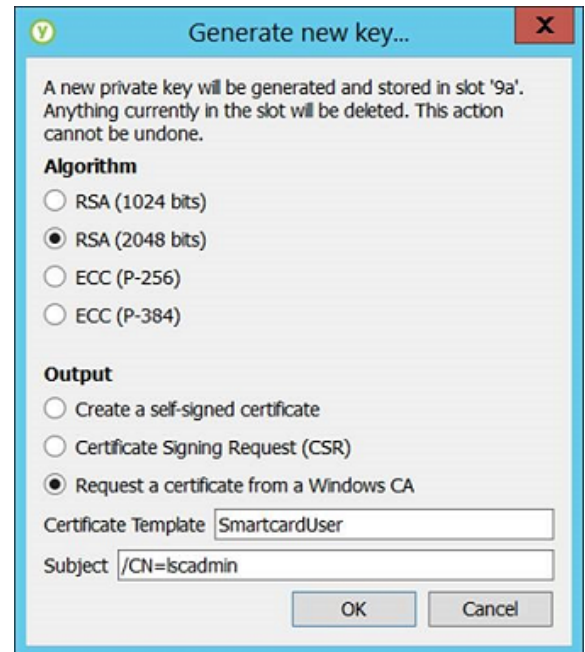


2. Click **Generate new key**. (Or, if the certificate has been exported previously, click **Import from file**.)



3. Complete the form by choosing the algorithm and the output. In the following example the selected output is **Request a certificate from a Windows CA**.

- Enter **SmartcardUser** for the **Certificate Template**.
- Enter **/CN=username** for the **Subject**.

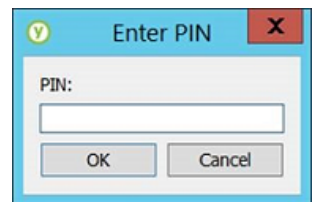


**Note:** The user's email address must be listed in Active Directory to use the SmartcardUser certificate template. If the email address is not listed, then you will get an error when trying to generate a new certificate.

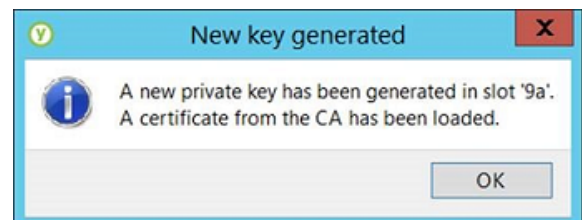
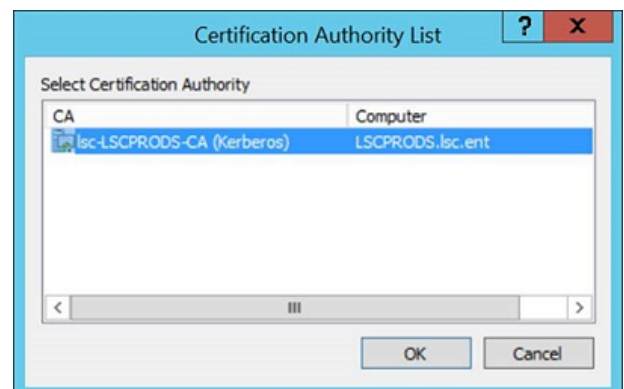
4. Click **OK**.

5. Enter the PIN.

6. Click **OK**.



7. Choose the CA that will generate the new certificate, then click **OK** to generate a new key. The new certificate is added to the published certificates for the user in Active Directory and in their personal certificate store.





## Configure Privileged Identity to use the Yubikey Smart Card

Also, certificate authentication must be allowed in both IIS and in the web application configuration. See the Installation Guide for more information.

# Manage Identities and Delegations for Password and System Access

This section describes creating identities and setting up delegations to provide access to the managed and stored passwords.

The basic process is:

1. Add an identity
2. Assign a permission

## Add Identities for Password and System Access

Identities, in the scope of Privileged Identity, refers to any of the following objects which are granted delegations (permissions) to perform some action within Privileged Identity:

- **Active Directory Users:** Users from any trusted domain may be granted access.
- **Active Directory Groups:** Groups from any trusted domain may be granted access.
- **Roles Containing LDAP/OAuth/SAML users:** Privileged Identity roles are collections of identities from LDAP directories as well as OAuth or SAML providers such as Okta or Azure and others.
- **LDAP Users:** Users from an LDAP compliant directory.
- **RADIUS users:** Users from anywhere that can use RADIUS for authentication.
- **Certificates (certificates, biometric, smart cards, etc.)** - Certificate enabled users cover all forms of access where a certificate is provided to the user.
- **Explicit accounts:** Explicit accounts exist only in the context of Privileged Identity. These accounts are often used for troubleshooting or when no other directory option exists. There is no password policy that can be applied to these accounts.

The following subsections provide information on adding each of these identity types.



*Using a non-trusted or non-Microsoft directory will require additional configuration. For more information, please see "Configure Authentication Servers" on page 310 for more information.*

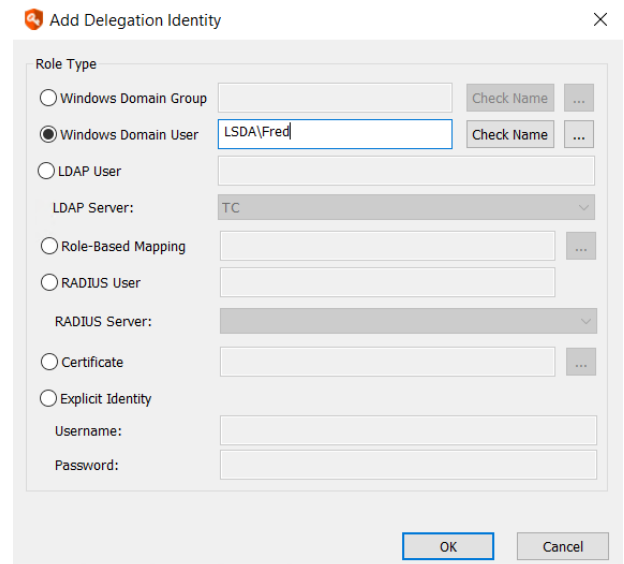
## Active Directory Users

Privileged Identity can add users from trusted Active Directory domains. Trusted domains are setup as authentication server entries automatically and require no further configuration. Non-trusted domains should be configured as "[LDAP Authentication Servers](#)" on page 311 or "[SAML Authentication Servers](#)" on page 320.

Users can be added from the management console, web application, or programmatically.

### Add a Domain User from the Management Console

1. From the management console, select **Delegation > Web Application Global Delegation Permissions**.
2. Click the **Add** button in the lower left corner.
3. Select **Windows Domain User**.
4. Add the username as **DomainName\UserName** where DomainName is the NetBIOS name of the domain and Username is the pre-windows 2000 account name for the user. You can either type in the name or click the ellipses (...) to browse for the user (note this uses legacy APIs to locate the users).
5. If you typed in the name, click **Check Name** to validate the name you just typed in.
6. Click **OK** to add the user.
7. The user will appear in the identity list with a type of Domain User and be granted the global right of Logon.

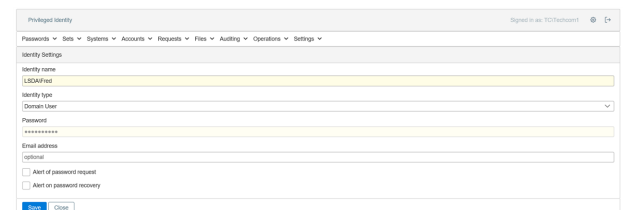


### Add a Domain User from the Web Application

Users may be added to the web application by any user with either of the following permissions:

- All Access
- Manage Delegations

1. In the web application select **Settings > Delegation** from the menu.
2. Click the **(+ New Identity)** button.
3. Type in the identity name as **DomainName\UserName** where DomainName is the NetBIOS name of the domain and Username is the pre-windows 2000 account name for the user. There is no browse feature. If the name cannot be properly verified, the name will not be added and there will be no further errors shown to the user.



4. Select the type as **Domain User**.
5. Click **Save** to add the user.
6. The user is added and granted the global right of **Logon**.

## Add a Domain User Programmatically

- From PowerShell, call `New-LSDelegationIdentity`.
- From SOAP, call `DelegationOps_CreateIdentity`.
- From REST, call `/REST/Delegation/Identity`.

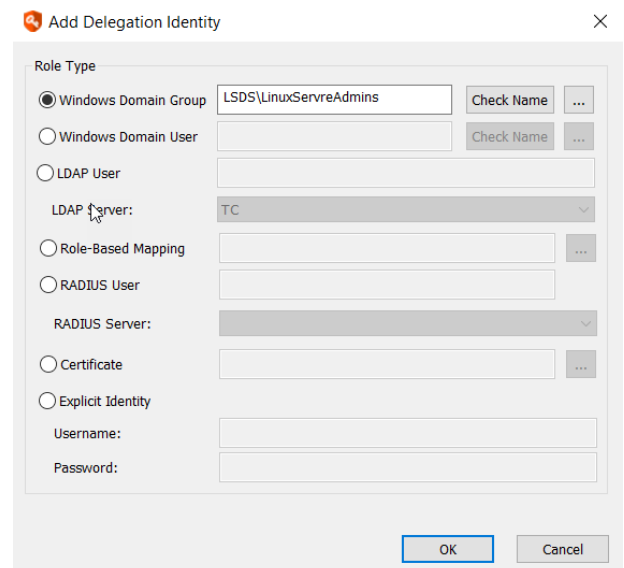
## Active Directory Groups

Privileged Identity can add users from trusted Active Directory domains. Trusted domains are setup as authentication server entries automatically and require no further configuration. Non-trusted domains should be configured as "[LDAP Authentication Servers](#)" on page 311 or "[SAML Authentication Servers](#)" on page 320.

Groups can be added from the management console, web application, or programmatically.

### Add a Domain Group from the Management Console

1. From the management console, go to **Delegation > Web Application Global Delegation Permissions**.
2. Click the **Add** button in the lower left corner.
3. Select **Windows Domain Group**.
4. Add the group name as **DomainName\GroupName** where DomainName is the NetBIOS name of the domain and GroupName is the pre-windows 2000 account name for the group. You can either type in the name or click the ellipses (...) to browse for the user (note this uses legacy APIs to locate the groups).
5. If you typed in the name, click **Check Name** to validate the name you just typed in.
6. Click **OK** to add the group.
7. The user will appear in the identity list with a type of Domain Group and be granted the global right of Logon.

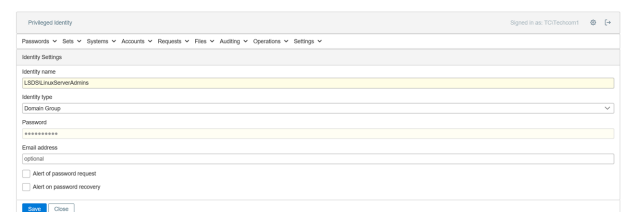


### Add a Domain Group from the Web Application

Users may be added to the web application by any user with either of the following permissions:

- All Access
- Manage Delegations

1. In the web application select **Settings > Delegation** from the menu.
2. Click the **New Identity (+)** button.
3. Type in the identity name as **DomainName\GroupName** where DomainName is the NetBIOS name of the domain and GroupName is the pre-windows 2000 account name for the group.
4. Select the type as **Domain Group**.



5. Click **Save** to add the group.
6. The group will be added and granted the global right of **Logon**.

## Add a Domain Group Programmatically

- From PowerShell, call `New-LSDelegationIdentity`.
- From SOAP, call `DelegationOps_CreateIdentity`.
- From REST, call `/REST/Delegation/Identity`.

## Roles for LDAP, OAuth, and SAML Users

Privileged Identity can create roles to group multiple user objects into a single identity. Roles can contain LDAP users, OAuth users, or SAML users. New Roles can only be created from the management console or programmatically.

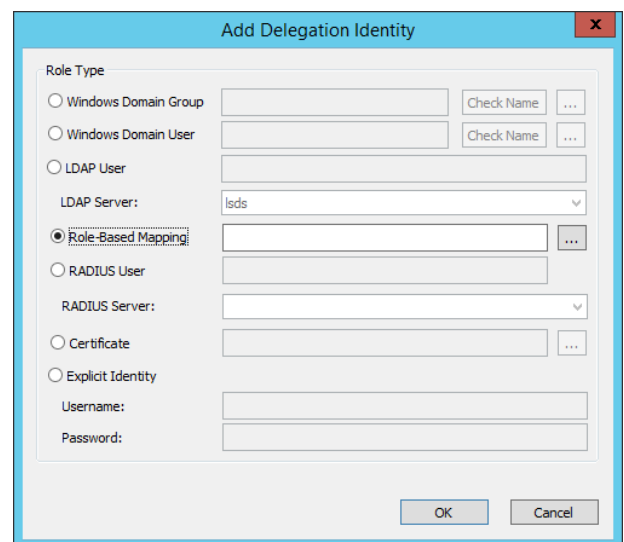
To add identities to roles, a specific authentication server entry will need to be created first:

- For LDAP Users, see "[LDAP Authentication Servers](#)" on page 311.
- For OAuth Users, see "[OAuth Authentication Servers](#)" on page 313.
- For SAML Users, see "[SAML Authentication Servers](#)" on page 320.

This section describes how to create a role and how to add a user to the role.

### Adding a Role from the Management Console

1. From the management console, go to **Delegation | Web Application Global Delegation Permissions**.
2. Click the **Add** button in the lower left corner.
3. Select **Role-Based Mapping**.
4. Type in a name for the role, then click **OK** to add the role.
5. The role will appear in the identity list with a Type set to Delegation Role.

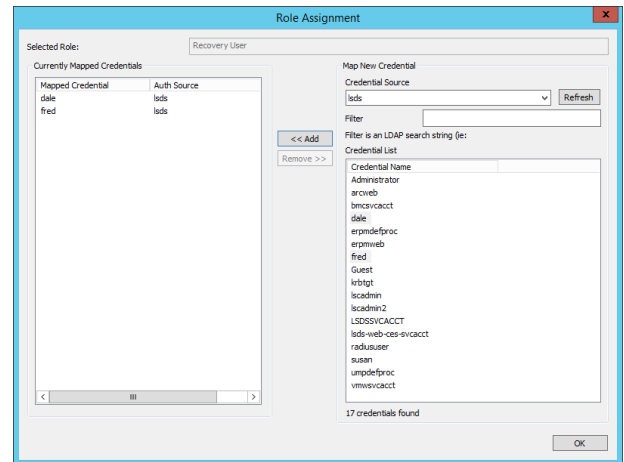


### Adding a User to a Role from the Management Console

Once a role has been added, users may be added to the role.

1. From the management console, go to **Delegation | Web Application Global Delegation Permissions**.
2. Select the target role, then click the **Assign** button in the lower left corner.
3. There is a slightly different process when adding a directory or OAuth user versus a SAML user.
  - For Directory and OAuth users, choose the target credential source (Authentication Server) then click **Refresh** to find the users from the directory. The list of users will be drawn based on the credential search filters for the authentication server entry. Use the filter to limit the search results.
  - For SAML users, choose the target credential source (Authentication Server), then supply the login name of the SAML user. Typically this is an email address. There is no search function for SAML users.

4. Select the user(s) from the Credential List field, then click the **Add** button. Users will be moved to the Currently Mapped Credentials field.
5. Repeat the steps as necessary to add users from any required authentication servers.
6. Click **OK** when done.



## Add a Role and Assigning Users Programmatically

See the Programmers Guide for more information.

### Adding a Role

- From PowerShell, call `New-LSDelegationIdentity`.
- From SOAP, call `DelegationOps_CreateIdentity`.
- From REST, call `/REST/Delegation/Identity`.

### Adding Users to a Role

- From SOAP, call `DelegationOps_RoleMappingPermission_Add`.
- From REST, call `/REST/Delegation/Identity/Role`.



## LDAP Users

Privileged Identity grants delegation permissions directly to LDAP users. LDAP User identities can only be created from the management console or programmatically.

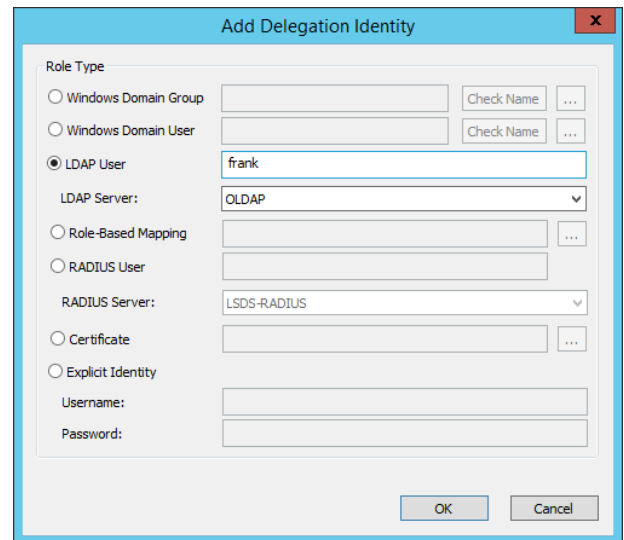
To add LDAP users as identities a specific authentication server entry will need to be created first:

- For LDAP Users, see "[LDAP Authentication Servers](#)" on page 311.

This section describes how to create an LDAP user identity.

### Adding an LDAP User from the Management Console

1. From the management console, go to **Delegation | Web Application Global Delegation Permissions**.
2. Click the **Add** button in the lower left corner.
3. Select **LDAP User**.
4. Select the appropriate LDAP Server
5. Type in a name for the LDAP, then click **OK** to add the role.
6. The role will appear in the identity list with a Type set to LDAP User.



The screenshot shows a dialog box titled "Add Delegation Identity". It has several radio buttons for "Role Type": "Windows Domain Group", "Windows Domain User", "LDAP User" (which is selected), "Role-Based Mapping", "RADIUS User", "Certificate", and "Explicit Identity". The "LDAP User" section has a text field containing "frank" and a dropdown menu for "LDAP Server" set to "LDAP". Other sections like "RADIUS User" and "Certificate" have their respective "Server" dropdowns and "Check Name" buttons. At the bottom, there are "OK" and "Cancel" buttons.

### Add an LDAP User Programmatically

See the Programmers Guide for more information.

- From PowerShell, call `New-LSDelegationIdentity`.
- From SOAP, call `DelegationOps_CreateIdentity`.
- From REST, call `/REST/Delegation/Identity`.

## RADIUS Users

Privileged Identity grants delegation permissions to RADIUS users. RADIUS User identities can only be created from the management console or programmatically.

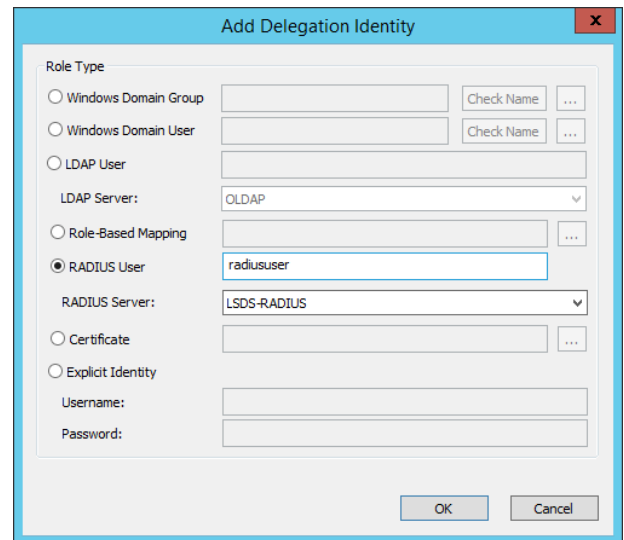
To add RADIUS users as identities a specific authentication server entry will need to be created first:

- For RADIUS Users, see "[RADIUS Authentication Servers](#)" on page 334.

This section describes how to create a RADIUS user identity.

### Adding a RADIUS User from the Management Console

1. From the management console, go to **Delegation | Web Application Global Delegation Permissions**.
2. Click the **Add** button in the lower left corner.
3. Select **RADIUS User**.
4. Select the appropriate RADIUS Server
5. Type in a name for the RADIUS user, then click **OK** to add the role.
6. The role will appear in the identity list with a Type set to RADIUS User.



The screenshot shows a dialog box titled "Add Delegation Identity". Under the "Role Type" section, the "RADIUS User" radio button is selected. The "RADIUS Server" dropdown menu is set to "LSDS-RADIUS". The "Username" field contains the text "radiususer". There are "OK" and "Cancel" buttons at the bottom right of the dialog.

### Add an LDAP User Programmatically

See the Programmers Guide for more information.

- From PowerShell, call `New-LSDelegationIdentity`.
- From SOAP, call `DelegationOps_CreateIdentity`.
- From REST, call `/REST/Delegation/Identity`.

## Certificates

Certificates can be used in place of typical user-based names and passwords. Privileged Identity can generate certificates or you can map the user's public key from any certificate authority.

When using this type of authentication the user's public key is added as identity into the delegation system. The user of course, has the corresponding private key. IIS must be configured for certificate authentication and to at least request user certificates. Privileged Identity must be configured to allow client certificates for user authentication and authorization. See the installation guide for more information on the configuration of the solution.

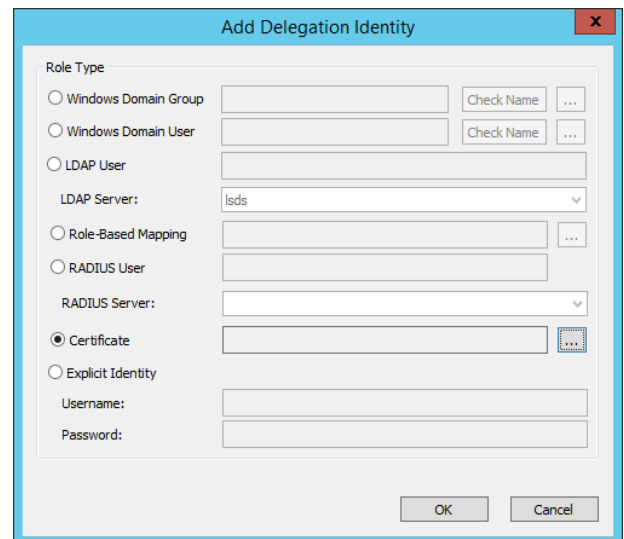
When the user attempts connection to the web application, if the user has only one certificate in their personal key store (on the Windows host or in the browser being used), the browser will pass that one key when IIS requests it. If the private key matches the stored public key, the user may be granted access to the web application if no other checks are being performed such as multi-factor authentication or the requirement of an additional login.

If the user has multiple certificates in their personal key store (on the Windows host or in the browser being used), the browser will prompt the user to select the private key to use for authentication. If the private key matches the stored public key, the user may be granted access to the web application if no other checks are being performed such as multi-factor authentication or the requirement of an additional login.

Certificates can be imported and mapped using the management console only.

### Add a Certificate-Based Identity from the Management Console

1. From the management console, go to **Delegation | Web Application Global Delegation Permissions**.
2. Click the **Add** button in the lower left corner.
3. Select **Certificate**.
4. Click the ellipses (...) to open the Enrolled Certificates dialog.
5. Select to enroll a certificate from one of the following locations:
  - **Enroll Cert(s) from File** - Select the user's public key file (.cer or .pfx) and click **OK**.
  - **Enroll Cert(s) from System** - Certificates added to any certificate store on the local host can be added for authentication.
  - **Create New Client Cert** - This will launch the makecert.exe program to create a root certificate (if not already found) for signing user certificates, created by makecert.exe which can then be distributed to users and enrolled. Certificates created through this process will be located in the CertificateUtils folder in the installation directory.



The screenshot shows a dialog box titled "Add Delegation Identity". It contains several options for "Role Type":

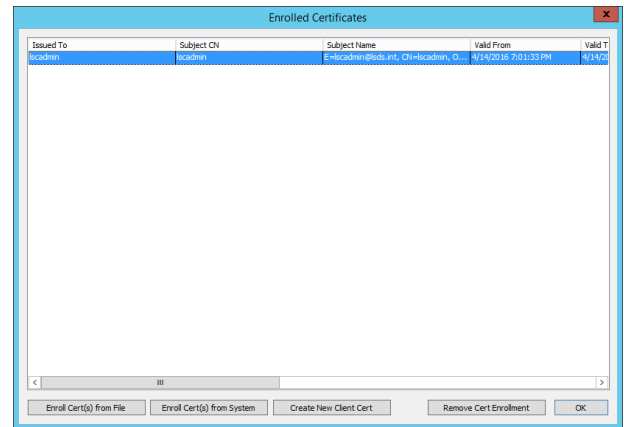
- Windows Domain Group
- Windows Domain User
- LDAP User
- Role-Based Mapping
- RADIUS User
- Certificate
- Explicit Identity

Additional fields include:

- LDAP Server: lsds
- RADIUS Server: (empty)
- Username: (empty)
- Password: (empty)

Buttons for "Check Name" and "OK" are visible. The "Certificate" option has a button with an ellipsis (...).

6. Highlight one certificate.
7. Click **OK** to add the role.
8. The certificate will appear in the identity list with a Type set to Certificate and the name will be set to the value in the Subject CN column of the certificate.



## Explicit Accounts

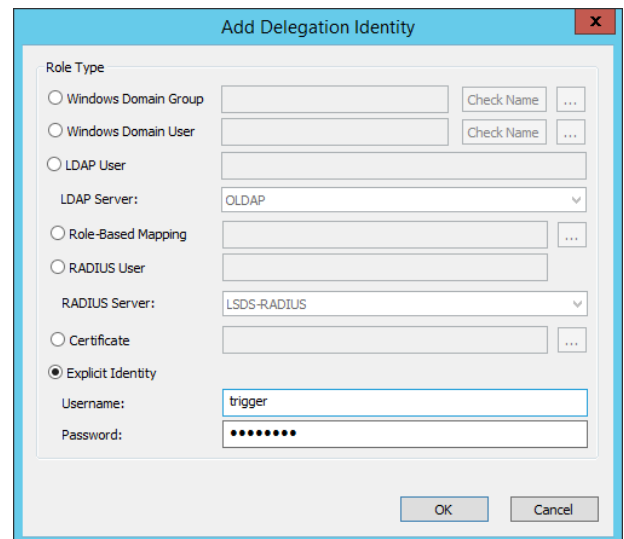
Privileged Identity can create and leverage its own identities for access to stored credentials. These explicit identities do not exist anywhere but in the scope of this product. Explicit users have no definable password policy which means the password could be as simple as one single character.

Typically, they are used in troubleshooting scenarios or where there is no central directory available.

Explicit users can be added from the management console, web application, or programmatically.

### Add an Explicit User Account from the Management Console

1. From the management console, go to **Delegation | Web Application Global Delegation Permissions**.
2. Click the **Add** button in the lower left corner.
3. Select **Explicit Identity**.
4. Supply a username and password.
5. Click **OK** to add the user.
6. The user will appear in the identity list with a type of Domain User and be granted the global right of Logon.



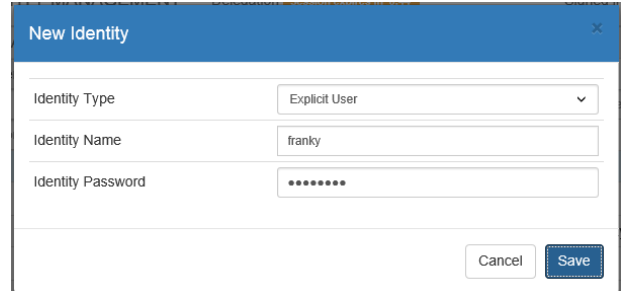
The screenshot shows a dialog box titled "Add Delegation Identity" with a close button (X) in the top right corner. Under the "Role Type" section, several radio buttons are listed: "Windows Domain Group", "Windows Domain User", "LDAP User", "Role-Based Mapping", "RADIUS User", "Certificate", and "Explicit Identity". The "Explicit Identity" radio button is selected. To the right of these options are input fields and "Check Name" buttons. For the "Explicit Identity" option, the "Username" field contains the text "trigger" and the "Password" field contains a series of dots representing a masked password. At the bottom of the dialog, there are "OK" and "Cancel" buttons.

### Add an Explicit User Account from the Web Application

Users may be added to the web application by any user with either of the following permissions:

- All Access
- Manage Delegations

1. In the web application go to **Settings | Delegation**.
2. Click the **Add New Identity** button (+).
3. Select the type as **Explicit User**.
4. Supply an **Identity Name** and **Identity Password**.
5. Click **Save** to add the user.
6. The user will be added and granted the global right of Logon.



## Add a Domain User Programmatically

- From PowerShell, call `New-LSDelegationIdentity`.
- From SOAP, call `DelegationOps_CreateIdentity`.
- From REST, call `/REST/Delegation/Identity`.

## Administrate Managed Password Permissions

Managed passwords are those passwords associated with systems placed in management sets, as opposed to Shared Credential Lists (SCLs) or personal password stores. This nomenclature is used for passwords actively managed by the solution as well as imported/static passwords associated with account on these systems.

As there are many different levels at which permissions may be assigned, this topic is divided into multiple sections which outline how to assign permissions at those different levels.

Identities are initially created at the global level and will be assigned the logon permission upon creation.

The different delegation levels are:

- **"Global Delegations" on page 393** - Assign one or more identities access to portions of the web application and/or one or more management sets. There is no granular access at this level; a user with one or more permissions has all those permissions to all management sets listed for them in the global delegations dialog.
- **"Per-Management Set Delegations" on page 405** - Assign an identity specific permissions to a specific management set. A management set contains systems. Accounts are associated with systems. Therefore a user with access to a management set has that same level of access to all systems and thus all accounts in that management set. Use this level of permissions when it is desired to grant one set of permissions to a specific list of system, but the same identity should have different access to a different set of systems.
- **"Per-System Delegations" on page 408** - Assign an identity specific permissions to a specific system. Accounts are associated with systems. Therefore a user with access to a system has that same level of access to all accounts on that system. Use this level of permissions when it is desired to grant access only to accounts on a system owned by the target identity and nothing else.
- **"Per-Account Delegations" on page 411** - Assign an identity specific permissions to a specific account on a specific system. Use this level of permissions when it is desired to grant an identity a specific set of permissions to only one specific account on one specific system.
- **"Per-Job Delegations" on page 414** - Assign an identity the ability to view and run a particular job from the web application.

When assigning permissions at multiple level, keep in mind that permissions are cumulative and there is no DENY permissions in the delegation system. Unless explicitly granted, permissions are implicitly denied. This means the best practice is to be less permissive at higher levels (Global) and grant more permissions as you get closer to the target account (Per-Account).

For example an identity needs to be able to recover passwords for the built-in administrator accounts on all the workstations s/he manages. However, the identity occasionally needs to perform work on related servers which are dispersed across multiple management sets, but this is done with the approval of the server admins. The management sets and permissions would be configured as follows:

- Multiple management sets for the servers.
- One management set for the workstations.
- Global delegations grant this identity:
  - Logon
  - View Accounts
  - Request Password Access
- Global delegations list the following management sets for this identity:
  - Multiple management sets for the servers.

- Per-Management set delegations for the identity are configured as follows:
  - Workstations management set which grants the identity:
    - View accounts
    - Recover Passwords



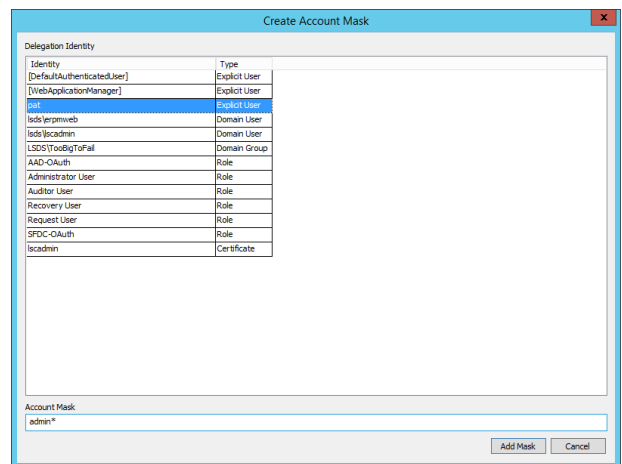
## Account Masks

Account Masks are designed to minimize the list of user accounts that might be exposed to a user who has access to an entire list of systems. For example, an account mask of admin\* will ensure the target identity only sees accounts beginning with "admin".

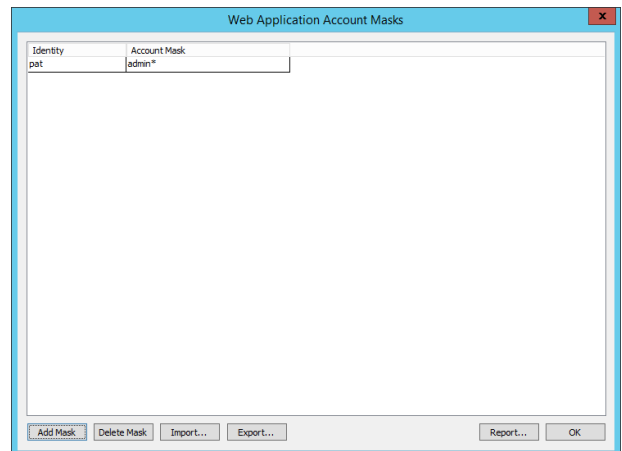
Account masks can be configured from the management console or programmatically.

### Configuring an Account Mask in the Management Console

1. In the management console go to **Delegation | Web Application Account Masks**.
2. Click **Add Mask**.
3. Select the target identity from the list.
4. Specify an account mask. The format uses a dos style wild card format. Use \* or ? for wildcards.
5. Click **Add Mask**.



6. Use the **Import** or **Export** buttons to import or export account masks (described below). Click **Report** to generate a report of the account masks.



### Importing Account Masks

To import permissions use the account masks dialog or go to **Delegation | Import/Export Delegation Rules | Import Account Masks**.

The required import format is:

```
IdentityName,AccountMask
```

The identity name does not need to pre-exist. The account name does not need to pre-exist.

For example, an identity named **demo\bob** that would have password recovery/request restricted to accounts that end with the characters **admin** the import entry for them would be:

```
demo\bob,*admin
```

## Configuring an Account Mask Programmatically

See the Programmers Guide for more information.

- From PowerShell, call `Set-LSDelegationPermissionAccountMask`.
- From SOAP, call `DelegationOps_AccountMaskPermission_Add`.
- From REST, call `/REST/Delegation/AccountMask`.

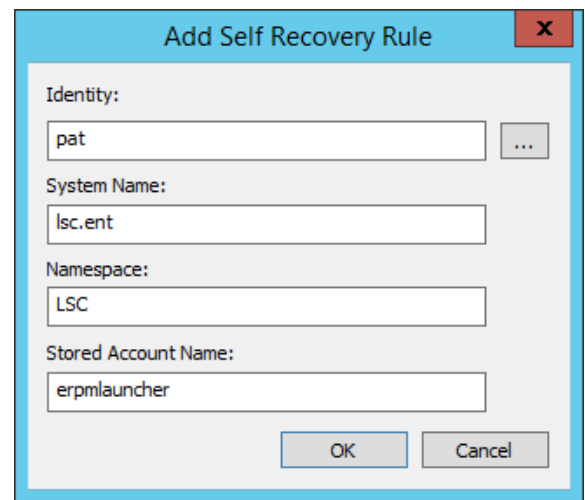
## Self-Recovery Permissions

*Self-recovery* rules are designed to grant full access to a single account. This is the simplest way to grant all access to a single account. To provide granular permissions for accessing an account, use per-account permissions.

Self-recovery rules can be configured from the management console or programmatically.

### Configuring an Account Mask in the Management Console

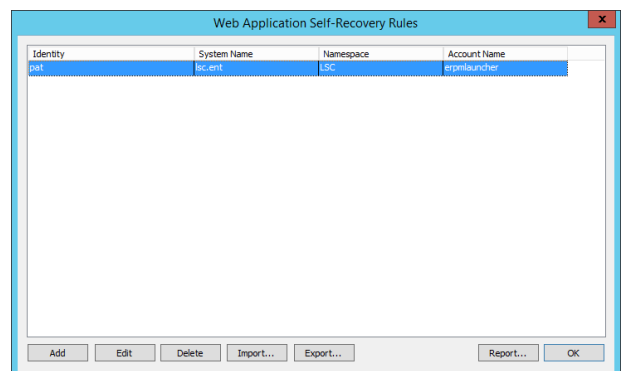
1. In the management console go to **Delegation | Web Application Self-Recovery Permissions**.
2. Click **Add**.
3. Supply the following information:
  - **Identity** - Select the target identity which will be granted the self recovery rule.
  - **System Name** - The target system name.
  - **Namespace** - The target system's namespace. See "[Namespace Values](#)" on page 589 for more information on available namespaces.
  - **Stored Account Name** - The account on the target system that the identity will have full access to.
4. Click **OK**.



Use the **Import** or **Export** buttons to import or export account masks (described below). Click **Report** to generate a report of the account masks.

### Importing Self-Recovery Rules

To import permissions use the self-recovery rules dialog or go to **Delegation | Import/Export Delegation Rules | Import Self-Recovery Permissions**.



Identity	System Name	Namespace	Account Name
pat	lsc.ent	LSC	erpmlauncher

The required import format is:

```
IdentityName, SystemName, NameSpace, AccountName
```

The identity name does not need to pre-exist. The account name does not need to pre-exist.

For example, an identity named **demo\bob** that would have access to an account named rimlauncher on a domain named lsc.ent with a namespace of LSC would be:

```
demo\bob,lsc.ent,LSC,rmlauncher
```

## Configuring Self-Recovery Permissions Programmatically

See the Programmers Guide for more information.

- From PowerShell, call `New-LSDelegationPermissionForSelfRecovery`.
- From SOAP, call `DelegationOps_SelfRecoveryPermission_Add`.
- From REST, call `/REST/Delegation/SelfRecoveryPermission`.

## Global Delegations

Global delegations assign one or more identities access to portions of the web application and one or more management sets. There is no granular access at this level; a user with one or more permissions has all those permissions to all management sets listed for them in the global delegations dialog.

About Global permissions:

- Global permissions can be defined via the management console, web application or programmatically.
- Global permissions will override permissions at a lower level. For example, if you have access to a management set that contains a system where you have been granted recover passwords, but a lower level, such as per-system permissions, grants you only request passwords, your effective rights will be the more permissive (cumulative) permissions, which means you will effectively have recover passwords for that system.
- Global permissions are the only permissions that can be time restricted.
- Global permission time restrictions can only be set in the management console.

For additional information, see the following sections:

- See "[Web Application Global Permissions in the Management Console](#)" on page 394 for directions on assigning global permissions via the management console.
- See "[Web Application Global Permissions in the Web Application](#)" on page 401 for directions on assigning global permissions via the web application.
- See "[Web Application Global Permissions Time Restrictions](#)" on page 403 for directions on configuring time-based restrictions for global permissions.

## Add Global Delegations Programmatically

See the Programmers Guide for more information.

- From PowerShell, call `New-LSDelegationIdentity`.
- From SOAP, call `DelegationOps_CreateIdentity`.
- From REST, call `/REST/Delegation/Identity`.

## Web Application Global Permissions in the Management Console

The global permissions dialog has five components:

- **Delegation Identities** - This component lists all identities added into Privileged Identity. When a new identity is added it is automatically granted the global Logon permission. Identities can be added, deleted edited (Password and assign role) or the entire list can be exported or a new list imported using the specific buttons at the bottom of the delegation identities field.
- **Permissions** - The permissions component lists all global permissions assigned to a selected identity. Permissions are divided into two sub-sections: Global Rules which affect access to the entire web application and Rules for management sets which are applied to all assigned management sets defined in the Management Sets field on the right side of this dialog.
- **Time restrictions** - Time restrictions define which rights are always available or restricted based on date and/or time of day.
- **Management Sets** - Lists the management sets assigned to the selected identity. At this level there is no ability to grant permissions to a specific management set. All management sets are treated equally for the purposes of granting permissions to a specific identity.
- **Alert Settings** - Specify one or more email addresses separated by semi-colon (;) to send alerts to. This email address field is used to send alerts when:
  - The identity is configured to grant requests for access and someone else triggers a request on a system the identity can grant access to.
  - A managed account password is retrieved from the web interface for an account that belongs to a management set assigned to the identity.
  - The identity requests a password, this email address will appear in the reply to email field. Upon the requests grant or denial, this email address will receive the email reply indicating the requests status.

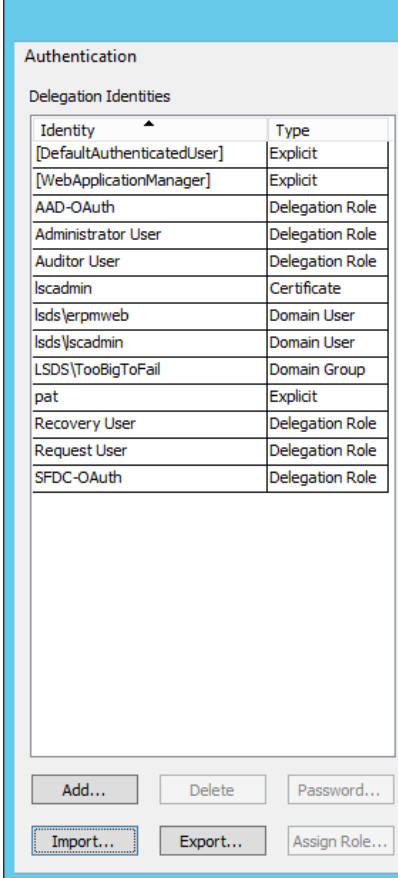
## Delegation Identities in the Management Console

The Delegation Identities section lists all identities added to the solution. The identity column lists the exact name of the identity and the type column lists the type of identity it is.

For long lists, there is no search function.

Use the following buttons to:

- **Add** - Add a new identity. See "[Add Identities for Password and System Access](#)" on [page 374](#) for more information.
- **Delete** - Remove an identity from the solution.
- **Password** - Allows resetting the password of an explicit identity.
- **Import** - Import a list of delegations and their permissions from a CSV file. See "[Import and Export Global Permissions](#)" on [page 398](#) for more information.
- **Export** - Export the current list of delegations and their permissions. See "[Import and Export Global Permissions](#)" on [page 398](#) for more information.
- **Assign Role** - Select a role and assign LDAP, OAuth, or SAML users to the role.



Authentication	
Delegation Identities	
Identity	Type
[DefaultAuthenticatedUser]	Explicit
[WebApplicationManager]	Explicit
AAD-OAuth	Delegation Role
Administrator User	Delegation Role
Auditor User	Delegation Role
Iscaadmin	Certificate
Isds\erpmweb	Domain User
Isds\iscaadmin	Domain User
LSDS\TooBigToFail	Domain Group
pat	Explicit
Recovery User	Delegation Role
Request User	Delegation Role
SFDC-OAuth	Delegation Role

## Global Identity Rules

Global identity rules grant permissions to a selected identity with a scope of the entire web application; these permissions are not applicable to a specific management set. These permissions apply to both the web application and web service. These permissions do not apply to management console access.

Following is a description of the global identity rules.

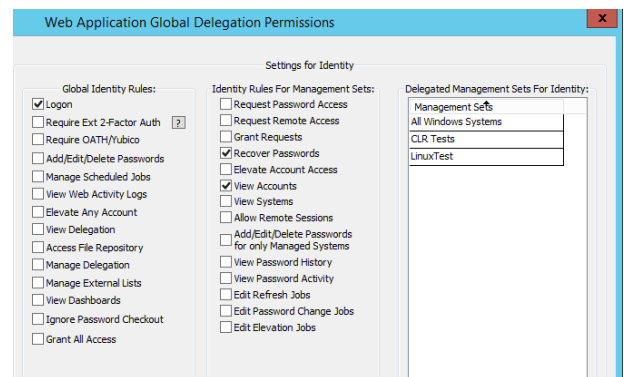
- **Logon** - Allows the identity to logon to the web application or web service.
- **Require Ext 2-Factor Auth** - The identity will be required to use an external multi-factor authentication solution to logon. External MFA includes, RSA, Safenet, RADIUS, and Infocrypt. External MFA does not include the OATH/Yubico forms of MFA.
- **Require OATH/Yubico** - The identity will be required to use the internal OATH MFA system to logon.
- **Add/Edit/Delete Passwords** - The identity will be able to add, edit, or delete passwords via the web application. This permissions can potentially allow a user to add a system they should not have access to into a management set they do have access to, thus granting them some kind of access to existing managed passwords on that system.
- **Manage Scheduled Jobs** - The identity will be allowed to view and run any and all jobs.
- **View Web Activity Logs** - The identity will be allows to view and export all audit logs.

- **Elevate Any Account** - Allows access to the global arbitrary account elevation feature.
- **View Delegation** - Allows the identity to view but not edit all delegations.
- **Access File Repository** - Allows use of the file repository. The identity will be allowed to add new content but will still require permissions to access content placed by other identities.
- **Manage Delegation** - Allows editing of delegations. This should be combined with View Delegation to allow the identity to properly edit permissions in the web application.
- **Manage External Lists** - Grants full control to all shared credential lists.
- **View Dashboards** - Grants access to add and view dashboards to their user profile.
- **Ignore Password Checkout** - Allows the identity to check out a password programmatically even if the password is already checked out to another identity at that time.
- **Grant All Access** - Full control of the web application.

## Identity Rules for Management Sets

Identity Rules for Management Sets are the permissions the selected identity will have over the management sets listed in the Management Sets field.

- **Request Password Access** - Allows the identity to request a password.
- **Request Remote Access** - Allows the identity to request remote access (SSH or RDP) as well as request access via the application launcher.
- **Grant Requests** - The identity will be able to grant or deny requests for accounts on systems in the management sets listed in the Management Sets field. The user may also receive password/remote access requests if the email address field is configured and the option Send Email Alerts for All Password Requests on Managed Systems is also enabled.
- **Recover Passwords** - The identity will be able to retrieve passwords without have to first request access. If recover and request passwords are both granted, the user will effectively be granted recover. The identity must also be grated View Accounts to recover passwords from the web application.
- **Elevate Account Access** - The identity will be allowed to use self-service account elevation to add themselves to a pre-configured group for a pre-configured period of time on a target Windows or Linux host from one of the configured management sets. The identity must also be granted View Systems, to leverage this feature from the web application.
- **View Accounts** - Allow the user to see all discovered accounts on the target systems.
- **View Systems** - Allow the user to see all systems in target management sets.
- **Allow Remote Sessions** - The identity will be allowed to initiate RDP and SSH access (if enabled) as well as to use the application launcher (if enabled) without having to first request access.
- **Add/Edit/Delete Passwords for only Managed Systems** - The identity will be able to add, edit or delete passwords for systems that already exist in target management sets.
- **View Password History** - The identity will be allowed to view historical passwords for the target account. The identity must also be granted View Systems, to leverage this feature from the web application.
- **View Password Activity** - The identity will be allowed to view the activity logs surrounding the particular target account. The identity must also be granted View Systems, to leverage this feature from the web application.
- **Edit Refresh Jobs** - This option is for programmatic access only via the web service. Allows a user to edit any refresh jobs associated with the systems and management sets they are delegated access to.





- **Edit Password Change Jobs** - This option is for programmatic access only via the web service. Allows a user to edit any password change jobs associated with the systems and management sets they are delegated access to.
- **Edit Elevation Jobs** - This option is for programmatic access only via the web service. Allows a user to edit any account elevation jobs associated with the systems and management sets they are delegated access to.

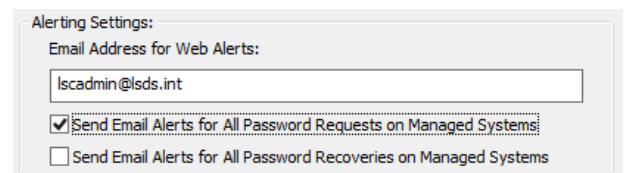
In the screen shot below, the identity has been granted the global logon permission and will be allowed to recover passwords and view accounts on all system in the All Windows Systems, CLR Tests and LinuxTest management sets.

## Alert Settings

The alert settings define one or more email addresses for the selected identity. Multiple email addresses should be separated by a semi-colon (;).

The identity will only receive alerts for systems/accounts located in management sets listed in the Delegated Management Sets for Identity field. If no management sets are listed, the identity will not receive any alerts for any operations regarding requests or recoveries. However, the identity may still grant requests (when Grant Requests is assigned), but will not receive alerts from Privileged Identity for those requests.

- **Email address** - The semi-colon delimited list of email addresses that may be used for alerts. This email address is also automatically used to fill out the web form request for password or remote access. When the request is granted or denied, the identity will receive the appropriate alert at that email address. The user may update the address in their own web settings or supply a different email address at request time.
- **Send Email Alerts for All Password Requests on Managed Systems** - If a password or access request is made for an account/system in a listed management set and the user is configured to grant those requests, they will receive an email alert for that request.
- **Send Email Alerts for All Password Recoveries on Managed Systems** - If a password is retrieved for an account/system in a listed management set, they will receive an email alert for that request.



## Restrict Permissions by Schedule

Global permissions can be restricted based on a schedule. Time restrictions can only be manipulated via the management console. See "[Web Application Global Permissions Time Restrictions](#)" on page 403 for more information.

## Import and Export Global Permissions

Global permissions can be imported or exported from the Global Permissions dialog or from **Delegation | Import/Export Delegation Rules | {Import | Export } Global Delegation {Option}**.

The import/export rules perform these functions:

- **Export from the Global Delegations Dialog** - Exports all identities, permissions and management sets.
- **Global Delegation Identities** - Import/Export only the listed identities. The expected import format is:

```
IdentityName, IdentityType, Password
```

- **Global Delegation Identity Permissions** - Import/Export permissions assigned to identities. Importing does not add identities if missing. Permissions will only be assigned if the identities currently exist. The expected import format is:

```
IdentityName, PermissionBits
```

- **Global Delegation Identity Management Sets** - Import/Export the listed identities and their assigned management sets. Importing does not add identities if missing. Importing only assigns management sets if the identities exist. The expected import format is:

```
IdentityName, ManagementSetName
```

- **Global Delegation Identities and Permissions** - Import/Export the identities and all assigned permissions. Importing does add the identities if missing and assigns permissions. The expected import format is:

```
IdentityName, IdentityType, Password, PermissionBits
```

- **Global Delegation Identities and Management Sets** - Import/Export the identities and assigned management sets, but not permissions. Importing does add the identities if missing and assigns management sets. The expected import format is:

```
IdentityName, IdentityType, Password, ManagementSetName
```

- **Global Delegation Identities, Permissions, and Management Sets** - Import/Export the complete global delegation structure. On import, any missing delegation elements will be added. The expected import format is:

```
IdentityName, IdentityType, Password, PermissionBits, ManagementSetName
```

When performing an import, you will be prompted if you would like to update existing permission entries that are duplicated. Choosing yes means to overwrite the existing permission structure for an existing identity. Choosing no means that entry will be skipped if there is a conflict.

### Identity Types:

For `identity_type`, specify one of the following integers:

- **Explicit identity** - 0
- **Active Directory domain user** - 1
- **Active Directory global security group** - 2
- **Self-Recovery identity** - 3
- **Role** - 4
- **RADIUS user** - 5
- **Certificate** - 6

For **password**, this parameter is valid only for explicit identities and should contain the clear text password. If this value is present for other account types, the value will be ignored.

For **identity\_name** use the fully-qualified name of the identity to be added. For example, format a domain user as follows:

```
DomainName\UserName
```

RADIUS users should be input as follows:

```
Authenticator\[Domain\]UserName
```

where

- **Authenticator** is the name of the authentication server entry for the RADIUS server
- **Domain\UserName** is the fully-qualified name of the user account.

With RADIUS, the domain name may be optional, depending on where the user IDs are coming from.

## Global Permissions (shown in order presented in dialog):

- **Logon** = 1
- **Require Ext 2-Factor Auth** = 65536
- **Require OATH/Yubico** = 2097152
- **Add/Edit/Delete Passwords** = 1024
- **Manage Scheduled Jobs** = 8
- **View Web Activity Logs** = 16
- **Elevate Any Account** = 1048576
- **View Delegation** = 64
- **Access File Repository** = 262144
- **Manage Delegation** = 128
- **Manage External Lists** = 8388608
- **View Dashboards** = 16777216
- **Ignore Password Checkout** = 134217728
- **Grant All Access** = 2048

## Global Permissions for Management Sets (shown in order presented in dialog):

- **Request Password Access** = 4096
- **Request Remote Access** = 2147483648
- **Grant Password Requests** = 32768
- **Recover Passwords** = 256
- **Elevate Account** = 524288
- **View Accounts** = 4
- **View Systems** = 2
- **Allow Remote Sessions** = 131072
- **Add/Edit/Delete Passwords for only Managed Systems** = 4194304
- **View Password History** = 33554432
- **View Password Activity** = 67108864
- **Edit Refresh Jobs** = 268435456
- **Edit Password Change Jobs** = 1073741824
- **Edit Elevation Jobs** = 536870912

Permissions are cumulative. To calculate the permissions a given identity should have, add the bit values together to come up with a final number. For example, an identity named **demobob** that can **login**, **view accounts**, and **recover passwords** will have a bit mask of  $1 + 4 + 256$  for a combined value of 261. Thus the import entry for them would be:

```
demo\bob, 261
```

## Web Application Global Permissions in the Web Application

Users with any of the following permissions can edit delegations in the web application:

- All Access
- Manage Delegations <> Change Delegation

### Set Delegations in the Web Application

1. In the web application go to **Settings | Delegation**.
2. An identity must have been previously added. Click the **Edit Permissions** button (pencil icon) next to the identity. Identity attributes such as email and alert settings can be set here.
3. Expand Global Permissions.
4. Enable any required permissions.

While the management console and web application set the same permissions, the names are not presented in the same format for all permissions. Following is a description of the permissions in the format listed. If the names in the management console differ, they are shown in parenthesis.

- **All Access (Grant All Access)** - Full control of the web application.
- **Logon** - Allows the identity to logon to the web application or web service.
- **Require External 2 Factor (Require Ext 2-Factor Auth)** - The identity will be required to use an external multi-factor authentication solution to logon. External MFA includes, RSA, Safenet, RADIUS, and Infocrypt. External MFA does not include the OATH/Yubico forms of MFA.
- **Require OATH Authentication (Require OATH/Yubico)** - The identity will be required to use the internal OATH MFA system to logon.
- **View Systems** - Allow the user to see all systems in target management sets.
- **Elevate Account Access** - The identity will be allowed to use self-service account elevation to add themselves to a pre-configured group for a pre-configured period of time on a target Windows or Linux host from one of the configured management sets. The identity must also be granted View Systems, to leverage this feature from the web application.
- **Elevate Any Account** - Allows access to the global arbitrary account elevation feature.
- **View Accounts** - Allow the user to see all discovered accounts on the target systems.
- **View Password History** - The identity will be allowed to view historical passwords for the target account. The identity must also be granted View Systems, to leverage this feature from the web application.
- **View Password Activity** - The identity will be allowed to view the activity logs surrounding the particular target account. The identity must also be granted View Systems, to leverage this feature from the web application.
- **View Passwords (Recover Passwords)** - The identity will be able to retrieve passwords without have to first request access. If recover and request passwords are both granted, the user will effectively be granted recover. The identity must also be granted View Accounts to recover passwords from the web application.
- **Ignore Password Checkout** - Allows the identity to check out a password programmatically even if the password is already checked out to another identity at that time.
- **Request Remote Access** - Allows the identity to request remote access (SSH or RDP) as well as request access via the application launcher.

Edit Identity	
Identity Name	pat
Identity Type	Explicit User
Email Address	None
Alert On Password Request	<input type="checkbox"/>
Alert On Password Recovery	<input type="checkbox"/>
Global Permissions	
All Access	<input type="checkbox"/>
Logon	<input checked="" type="checkbox"/>
Require External 2 Factor	<input type="checkbox"/>
Require OATH Authentication	<input type="checkbox"/>

- **Request Passwords (Request Password Access)** - Allows the identity to request a password.
- **Grant Password Requests (Grant Requests)** - The identity will be able to grant or deny requests for accounts on systems in the management sets listed in the Management Sets field. The user may also receive password/remote access requests if the email address field is configured and the option Send Email Alerts for All Password Requests on Managed Systems is also enabled.
- **Allow Remote Sessions** - The identity will be allowed to initiate RDP and SSH access (if enabled) as well as to use the application launcher (if enabled) without having to first request access.
- **Add/Edit/Delete Passwords** - The identity will be able to add, edit, or delete passwords via the web application. This permissions can potentially allow a user to add a system they should not have access to into a management set they do have access to, thus granting them some kind of access to existing managed passwords on that system.
- **Add/Edit/Delete Passwords on Managed Systems (Add/Edit/Delete Passwords for only Managed Systems)** - The identity will be able to add, edit or delete passwords for systems that already exist in target management sets.
- **View Web Activity Logs** - The identity will be allows to view and export all audit logs.
- **View Delegation** - Allows the identity to view, but not edit, all delegations.
- **Change Delegation (Manage Delegation)** - Allows editing of delegations. This should be combined with View Delegation to allow the identity to properly edit permissions in the web application.
- **Manage Password Lists (Manage External Lists)** - Grants full control to all shared credential lists.
- **Access File Store (Access File Repository)** - Allows use of the file repository. The identity will be allowed to add new content but will still require permissions to access content placed by other identities.
- **Edit Refresh Jobs** - This option is for programmatic access only via the web service. Allows a user to edit any refresh jobs associated with the systems and management sets they are delegated access to.
- **Edit Web Panels (View Dashboards)** - Grants access to add and view dashboards to their user profile.

Use the management console to set the two following delegations: Edit Password Change Jobs and Edit Elevation Jobs.

## Web Application Global Permissions Time Restrictions

Time restrictions on global permissions allow certain privileges to be available at only certain times.

To configure time restrictions, a user must first be configured with all potential permissions. Then those permissions can then be configured for availability at certain times.

Consider the following scenario:

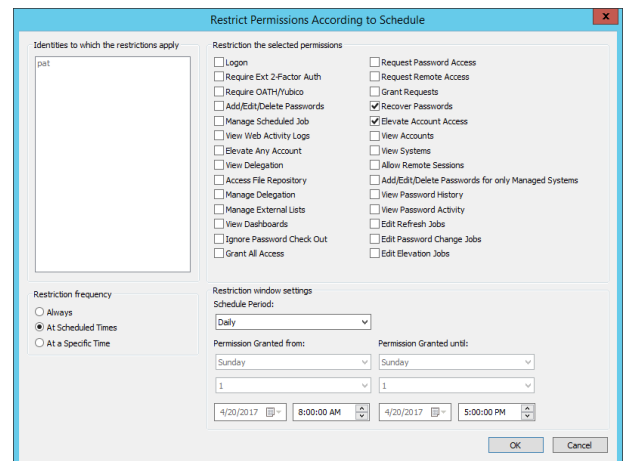
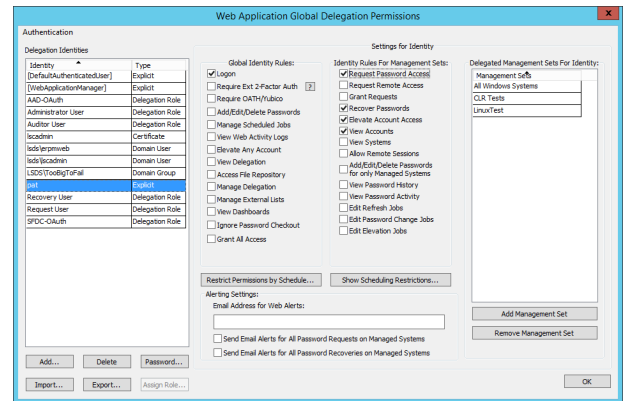
- User Pat has been granted Logon and Request Password Access, Recover Passwords, Elevate Account Access, and View Accounts for systems in three specific management sets.
- As Pat has both request password access and recover password access, Pat will be allowed to recover the passwords at any point in time.
- The desired behavior is for Pat to be able to retrieve passwords and escalate his account during normal business hours, but after hours, Pat must request password access and should not be allowed to escalate his account to a higher status.

Time restrictions can help.

### Adding Time Restrictions

1. Select **Pat** then click **Restrict Permissions by Schedule**.
2. Set the **Restriction Frequency** as appropriate. **At Scheduled Times** should be used for recurring events while **At a Specific Time** is useful for one time restrictions, such as when a contractor may come to work on your environment for only a defined period of time.
3. Remove the permissions that require no restrictions. Specifically, the requirement in this example is to ensure that Pat may only recover passwords and elevate account access during Pat's normal work schedule. Thus, those two permissions are the only permissions that will be left selected on this dialog.
4. Set the **Restriction Frequency** to **At Scheduled Times** as this restriction will occur every day.
5. Set the **Schedule Period** to **Daily**.
6. Set the start and stop times to encompass Pat's work schedule of 8AM to 5PM.
7. Click **OK**.

This will have the net effect of allowing Pat to recover passwords and elevate account access only from 8AM to 5PM every day. Outside of those hours, Pat will need to request password access and will not be permitted to elevate account access.

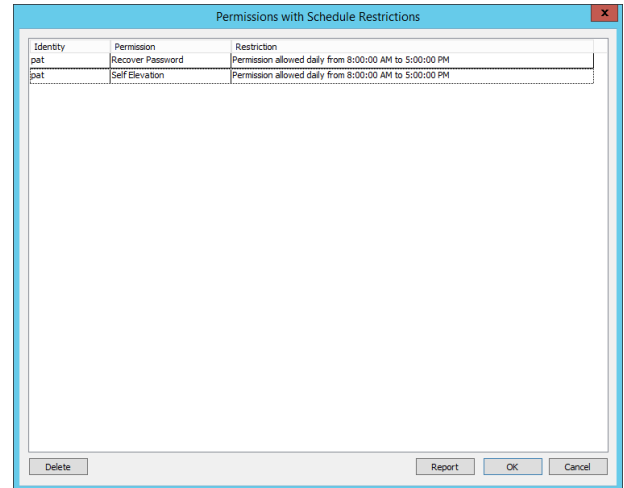


## Viewing Time Restrictions

Once a time restriction has been added, click **Show Scheduling Restrictions** to show them.

To delete a specific restriction, select the restriction then click **Delete**.

To report on the restrictions in the dialog, click the **Report** button. Report on specific time restrictions by selecting them first, then click **Report**.





## Per-Management Set Delegations

*Per-management set* permissions are used to assign an identity specific permissions to a specific management set.

About per-management set permissions:

- Per-management set permissions can be defined via the management console, web application or programmatically.
- Per-management set permissions override permissions at a lower level. For example, if you have access to a management set that contains a system where you have been granted recover passwords, but a lower level, such as per-system permissions, grants you only request passwords, your effective rights will be the more permissive (cumulative) permissions, which means you will effectively have recover passwords for that system.
- Per-management set permissions cannot be time restricted.

### Add Per-Management Set Delegations Programmatically

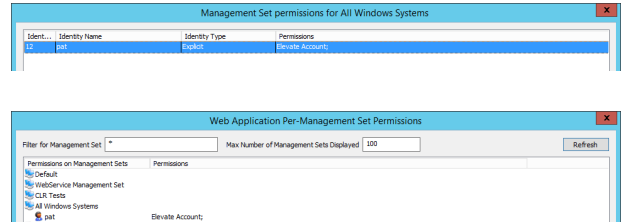
- From PowerShell, call **Set-LSDelegationPermissionOnManagementSet**.
- From SOAP, call **DelegationOps\_SetPermissionOnManagementSet**.
- From REST, call **/REST/Delegation/ManagementSet**.

## Web Application Per-Management Set Permissions in the Management Console

### Create Per-Management Set Permissions

1. In the management console, go to **Delegation > Web Application Per-Management Set Permissions**.
2. Right-click on a management set and select **Add Identities for Management Set**.
3. Click **Add Identity to List**.
4. Select an identity and click **OK**.
5. Select the identity and click **Edit Permissions on Identity**.
  - **Change Included Systems/Devices Management Set:** Add or remove systems from the management set.
  - **Request Password Access:** Request access to a password.
  - **Request Remote Access:** Request remote access (including app launcher) using an account to the target system.
  - **Grant Password Requests:** Grant or deny a password request or remote access request.
  - **Recover Passwords:** Retrieve a stored or managed password.
  - **Elevate Account:** Elevate your account status to a predefined group for a predefined period of time.
  - **View Accounts:** View the list of accounts on the systems in the management set.
  - **View Systems:** View the list of systems in the management set.
  - **Allow Remote Session:** Initiate an RDP or SSH or app launcher session without requiring a request process.
  - **Notify on Change:** When a password request is made and the type of request is listed as a **Change**, the identity is notified.
  - **Notify on Incident:** When a password request is made and the type of request is listed as an **Incident**, the identity is notified.

6. Add the desired permissions and click **OK**.
7. Click **OK**. The management set lists the added identities and their permissions.



## Import Permissions

Permissions can be imported in bulk. The expected format is:

```
IdentityName,ManagementSetName,PermissionBits
```

When importing these delegations, the logon user must preexist.

Following is the list of permission bits:

- **Change Included Systems/Devices Management Set** = 8388608
- **Request Password Access** = 4096
- **Request Remote Access** = 2147483648
- **Grant Password Requests** = 32768
- **Recover Passwords** = 256
- **Elevate Account** = 524288
- **View Accounts** = 4
- **View Systems** = 2
- **Allow Remote Session** = 131072
- **Notify on Change** = Value cannot be set
- **Notify on Incident** = Value cannot be set

Permissions are cumulative. To calculate the permissions a given identity should have, add the bit values together to come up with a final number.



**Example:** An identity named **demo\bob** that can **view accounts**, and **recover passwords** has a bit mask of 4 + 256 for a combined value of 260. Thus the import entry for them is:

```
demo\bob,All Windows Systems,260
```

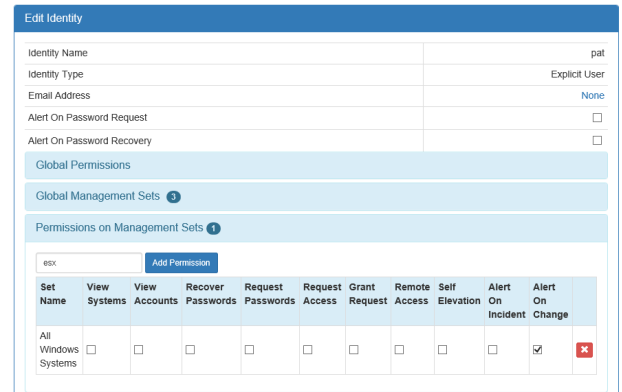
## Web Application Per-Management Set Permissions in the Web Application

Users with any of the following permissions can edit delegations in the web application:

- All Access
- Manage Delegations <> Change Delegation

## Set Delegations in the Web Application

1. In the web application, go to **Settings > Delegation**.
2. An identity must have been previously added. Click the **Edit Permissions** button (pencil icon) next to the identity. Identity attributes such as email and alert settings can be set here.
3. Expand **Permissions on Management Sets**.
4. Enable any required permissions.



Set Name	View Systems	View Accounts	Recover Passwords	Request Passwords	Request Access	Grant Request	Remote Access	Self Elevation	Alert On Incident	Alert On Change
All Windows Systems	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>

While the management console and web application set the same permissions, the names are not presented in the same format for all permissions. Following is a description of the permissions in the format listed. If the names in the management console differ, they are shown in parenthesis.

- **View Systems:** View the list of systems in the management set.
- **View Accounts:** View the list of accounts on the systems in the management set.
- **Recover Passwords:** Retrieve a stored or managed password.
- **Request Passwords (Request Password Access):** Request access to a password.
- **Request Access (Request Remote Access):** Request remote access (including app launcher) using an account to the target system.
- **Grant Request (Grant Password Requests):** Grant or deny a password request or remote access request.
- **Remote Access (Allow Remote Session):** Initiate an RDP or SSH or app launcher session without requiring a request process.
- **Self Elevation (Elevate Account):** Elevate your account status to a predefined group for a predefined period of time.
- **Alert on Incident (Notify on Incident):** When a password request is made and the type of request is listed as an **Incident**, the identity is notified.
- **Alert on Change (Notify on Change):** When a password request is made and the type of request is listed as a **Change**, the identity is notified.

Use the management console to set the following delegation: **Change Included Systems/Devices Management Set**.

## Per-System Delegations

*Per-system* permissions are used to assign an identity specific permissions to a specific system or device.

About per-system permissions:

- Per-system permissions can be defined via the management console, web application or programmatically.
- Per-system permissions override permissions at a lower level. For example, if you have access to a system where you have been granted recover passwords, but a lower level, such as account permissions, grants you only request passwords, your effective rights will be the more permissive (cumulative) permissions, which means you will effective have recover passwords for that system and all accounts on it.
- Per-system permissions cannot be time restricted.

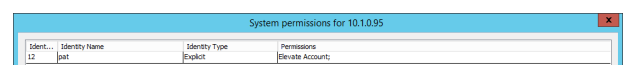
### Add Per System Delegations Programmatically

- From PowerShell, call **Set-LSDelegationPermissionOnSystem**.
- From SOAP, call **DelegationOps\_SetPermissionOnManagementSystem**.
- From REST, call **/REST/Delegation/System**.

## Web Application Per System Permissions in the Management Console

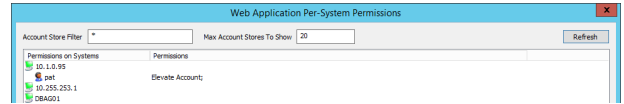
### Create Per System Permissions

1. In the management console, go to **Delegation > Web Application Per-System Permissions**.
2. Right-click on a system and select **Edit Managers for System**.
3. Click **Add Identity to List**.
4. Select an identity and click **OK**.
5. Select the identity and click **Edit Permissions on Identity**.
  - **View Accounts:** View the list of accounts on the systems in the management set.
  - **View Systems:** View the list of systems in the management set.
  - **Request Password Access:** Request access to a password.
  - **Request Remote Access:** Request remote access (including app launcher) using an account to the target system.
  - **Grant Password Requests:** Grant or deny a password request or remote access request.
  - **Recover Passwords:** Retrieve a stored or managed password.
  - **Elevate Account:** Elevate your account status to a predefined group for a predefined period of time.
  - **Allow Remote Session:** Initiate an RDP or SSH or app launcher session without requiring a request process.
  - **Notify on Change:** When a password request is made and the type of request is listed as a **Change**, the identity is notified.
  - **Notify on Incident:** When a password request is made and the type of request is listed as an **Incident**, the identity is notified.
6. Add the desired permissions and click **OK**.



Ident...	Identity Name	Identity Type	Permissions
12	pat	exploit	Elevate Account:

- Click **OK**. The management set lists the added identities and their permissions.



## Import Permissions

Permissions can be imported in bulk. The expected format is:

```
IdentityName, SystemIdentifier, PermissionBits, SystemType
```

When importing these delegations, the logon user needs to preexist.


Following are a list of system type bits:

- **Windows** = 1
- **Linux/Unix** = 2
- **Cisco** = 3
- **SQL Server** = 4
- **Oracle** = 5
- **MySQL** = 6
- **Custom** = 7
- **Sybase** = 8
- **DRAC** = 9
- **IPMI** = 10
- **LDAP** = 11

Following is the list of permission bits:

- **View Accounts** = 4
- **View Systems** = 2
- **Request Password Access** = 4096
- **Request Remote Access** = 2147483648
- **Grant Password Requests** = 32768
- **Recover Passwords** = 256
- **Elevate Account** = 524288
- **Allow Remote Session** = 131072
- **Notify on Change** = Value cannot be set
- **Notify on Incident** = Value cannot be set

Permissions are cumulative. To calculate the permissions a given identity should have, add the bit values together to come up with a final number.

 **Example:** For example, an identity named **demo\bob** that can **view accounts**, and **recover passwords** has a bit mask of 4 + 256 for a combined value of 260. Demo bob is granted permissions on an oracle database host. Thus the import entry for them is:



```
demo\bob, oradbhost\orcl.lsc.ent,260,5
```

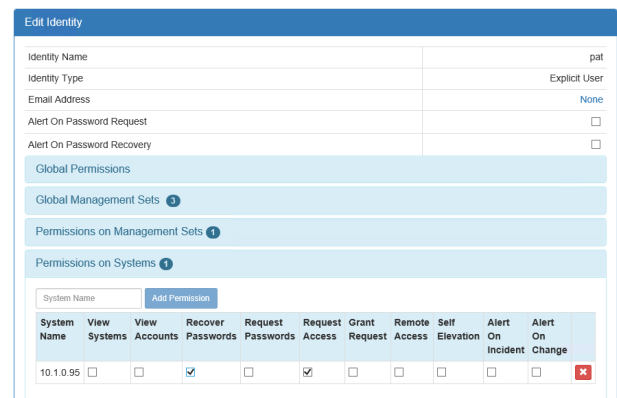
## Web Application Per-System Permissions in the Web Application

Users with any of the following permissions can edit delegations in the web application:

- All Access
- Manage Delegations <> Change Delegation

### Set Delegations in the Web Application

1. In the web application, go to **Settings > Delegation**.
2. An identity must have been previously added. Click the **Edit Permissions** button (pencil icon) next to the identity. Identity attributes such as email and alert settings can be set here.
3. Expand **Permissions on Systems**.
4. Enable any required permissions.



The screenshot shows the 'Edit Identity' interface. It includes fields for Identity Name, Identity Type, Email Address, and alert settings. The 'Permissions on Systems' section is expanded, showing a table with columns for System Name, View Systems, View Accounts, Recover Passwords, Request Passwords, Request Access, Grant Request, Remote Access, Self Elevation, Alert On Incident, and Alert On Change. The '10.1.0.95' system is selected, and the 'Recover Passwords' and 'Request Access' checkboxes are checked.

System Name	View Systems	View Accounts	Recover Passwords	Request Passwords	Request Access	Grant Request	Remote Access	Self Elevation	Alert On Incident	Alert On Change
10.1.0.95	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

While the management console and web application set the same permissions, the names are not presented in the same format for all permissions. Following is a description of the permissions in the format listed. If the names in the management console differ, they are shown in parenthesis.

- **View Systems:** View the list of systems in the management set.
- **View Accounts:** View the list of accounts on the systems in the management set.
- **Recover Passwords:** Retrieve a stored or managed password.
- **Request Passwords (Request Password Access):** Request access to a password.
- **Request Access (Request Remote Access):** Request remote access (including app launcher) using an account to the target system.
- **Grant Request (Grant Password Requests):** Grant or deny a password request or remote access request.
- **Remote Access (Allow Remote Session):** Initiate an RDP or SSH or app launcher session without requiring a request process.
- **Self Elevation (Elevate Account):** Elevate your account status to a predefined group for a predefined period of time.
- **Alert on Incident (Notify on Incident):** When a password request is made and the type of request is listed as an **Incident**, the identity is notified.
- **Alert on Change (Notify on Change):** When a password request is made and the type of request is listed as a **Change**, the identity is notified.

## Per-Account Delegations

*Per-account* permissions are used to assign an identity specific permissions to a specific account on a specific system.

About per-account permissions:

- Per-account permissions can be defined via the management console, web application or programmatically.
- Per-account permissions can be overridden by more permissive permissions at a higher level.
- Per-account permissions cannot be time restricted.

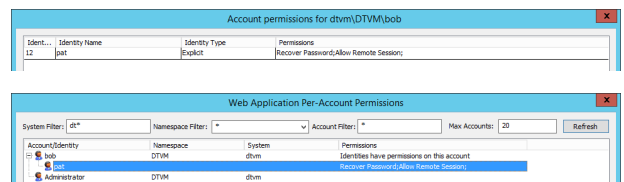
### Add Per Account Delegations Programmatically

- From PowerShell, call **Set-LSDelegationPermissionOnAccount**.
- From SOAP, call **DelegationOps\_SetPermissionOnAccount**.
- From REST, call **/REST/Delegation/StoredCredential**.

## Web Application Per Account Permissions in the Management Console

### Create Per Account Permissions

1. In the management console, go to **Delegation > Web Application Per-Account Permissions**.
2. Right click on a system and select **Edit Managers for Account**.
3. Click **Add Identity to List**.
4. Select an identity and click **OK**.
5. Select the identity and click **Edit Permissions on Identity**.
  - **View Accounts:** View the list of accounts on the systems in the management set.
  - **Request Password Access:** Request access to a password.
  - **Request Remote Access:** Request remote access (including app launcher) using an account to the target system.
  - **Grant Password Requests:** Grant or deny a password request or remote access request.
  - **Recover Passwords:** Retrieve a stored or managed password.
  - **Allow Remote Session:** Initiate an RDP or SSH or app launcher session without requiring a request process.
  - **Notify on Change:** When a password request is made and the type of request is listed as a **Change**, the identity is notified.
  - **Notify on Incident:** When a password request is made and the type of request is listed as an **Incident**, the identity is notified.
6. Add the desired permissions and click **OK**.
7. Click **OK**. The management set lists the added identities and their permissions.



## Import Permissions

Permissions can be imported in bulk. The expected format is:

```
SystemName,NameSpace,AccountName,IdentityName,PermissionBits
```

When importing these delegations, the logon user must preexist.

Following is the list of permission bits:

- **View Account** = 4
- **Request Password Access** = 4096
- **Request Remote Access** = 2147483648
- **Grant Password Requests** = 32768
- **Recover Passwords** = 256
- **Allow Remote Session** = 131072
- **Notify on Change** = Value cannot be set
- **Notify on Incident** = Value cannot be set

Permissions are cumulative. To calculate the permissions a given identity must have, add the bit values together to come up with a final number.



**Example:** An identity named **demo\bob** that can **view accounts**, and **recover passwords** has a bit mask of 4 + 256 for a combined value of 260. Demo bob will be granted permissions on a Windows member server named DTVM for an account named firecall. Thus the import entry for them is:

```
demo\bob, dtvm, DTVM, firecall, 260
```



For more information on available namespaces, please see "[Namespace Values](#)" on page 589.

## Web Application Per-Account Permissions in the Web Application

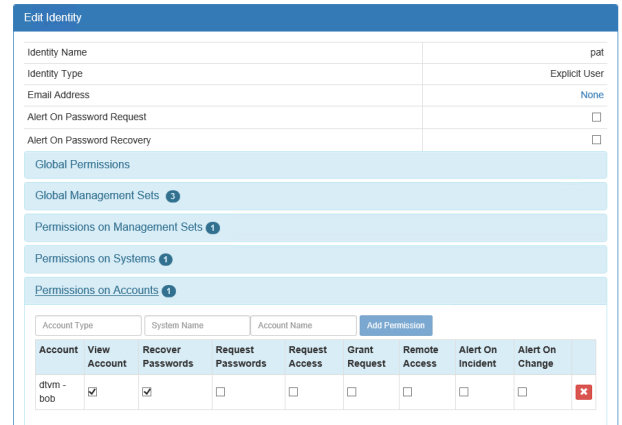
Users with any of the following permissions can edit delegations in the web application:

- All Access
- Manage Delegations <> Change Delegation



## Set Delegations in the Web Application

1. In the web application, go to **Settings > Delegation**.
2. An identity must have been previously added. Click the **Edit Permissions** button (pencil icon) next to the identity. Identity attributes such as email and alert settings can be set here.
3. Expand **Permissions on Accounts**.
4. Enable any required permissions.



The screenshot shows the 'Edit Identity' interface with the following sections:

- Global Permissions**
- Global Management Sets** (1)
- Permissions on Management Sets** (1)
- Permissions on Systems** (1)
- Permissions on Accounts** (1)

The 'Permissions on Accounts' section includes a table with the following columns:

Account	View Account	Recover Passwords	Request Passwords	Request Access	Grant Request	Remote Access	Alert On Incident	Alert On Change
dtvm - bob	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

While the management console and web application set the same permissions, the names are not presented in the same format for all permissions. Following is a description of the permissions in the format listed. If the names in the management console differ, they are shown in parenthesis.

- **View Account:** View the list of accounts on the systems in the management set.
- **Recover Passwords:** Retrieve a stored or managed password.
- **Request Passwords (Request Password Access):** Request access to a password.
- **Request Access (Request Remote Access):** Request remote access (including app launcher) using an account to the target system.
- **Grant Request (Grant Password Requests):** Grant or deny a password request or remote access request.
- **Remote Access (Allow Remote Session):** Initiate an RDP or SSH or app launcher session without requiring a request process.
- **Alert on Incident (Notify on Incident):** When a password request is made and the type of request is listed as an **Incident**, the identity is notified.
- **Alert on Change (Notify on Change):** When a password request is made and the type of request is listed as a **Change**, the identity is notified.

## Per-Job Delegations

Per-job permissions are used to assign an identity control of a specific job. This enables the identity to view job properties and run the job at will via the web site or web application.

About Per-Job permissions:

- Per Job permissions can be defined via the management console, web application or programmatically.
- Per Job permissions cannot be time restricted.

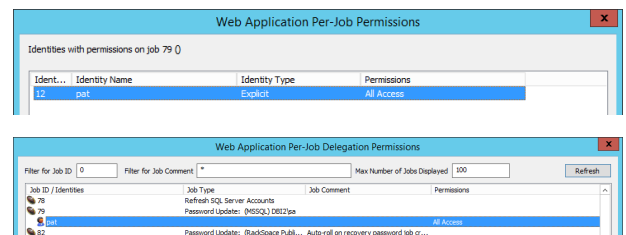
### Add Per Job Delegations Programmatically

- From PowerShell, call **Set-LSDelegationPermissionOnJob**.
- From SOAP, call **DelegationOps\_SetPermissionJob**.
- From REST, call **/REST/Delegation/Job**.

## Web Application Per-Job Permissions in the Management Console

### Create Per Job Permissions

1. In the management console, go to **Delegation > Web Application Per-Job Permissions**.
2. Select a job and click **Edit**.
3. Click **Add** to add an identity to list.
4. Select an identity and click **OK**.
5. Check the appropriate specific permissions to assign. By default, the identity can **View** and **Edit** the job, but you can uncheck **Edit** so that the user can only **View** the job.
6. Click **OK**.
7. Click **OK**. The management set lists the added identities and their permissions.
8. Click **OK**.



## Web Application Per-Job Permissions in the Web Application

Users with any of the following permissions can edit delegations in the web application:

- All Access
- Manage Delegations <> Change Delegation

## Set Delegations in the Web Application

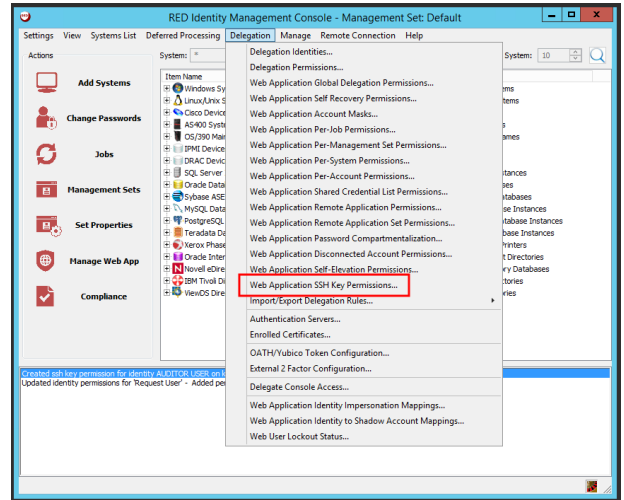
1. In the web application, go to **Settings > Delegation**.
2. An identity must have been previously added. Click the **Edit Permissions** button (pencil icon) next to the identity. Identity attributes such as email and alert settings can be set here.
3. Expand **Permissions on Jobs**.
4. Supply the target **Job ID** and click **Add Permission**.
5. Check the appropriate permissions to assign.

# Assign SSH Key Permissions in the Management Console

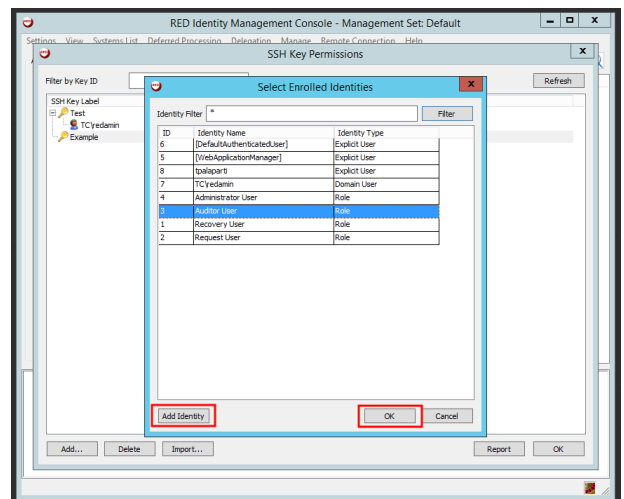
SSH Key permissions are used to assign a specific identity access to an SSH key and manage the rights for that identity. SSH Key permissions can be defined in the management console, as described here, via an API, or in the Web Application.

## Assign Identities to SSH Keys

1. In the management console, go to **Delegation > Web Application SSH Key Permissions....**

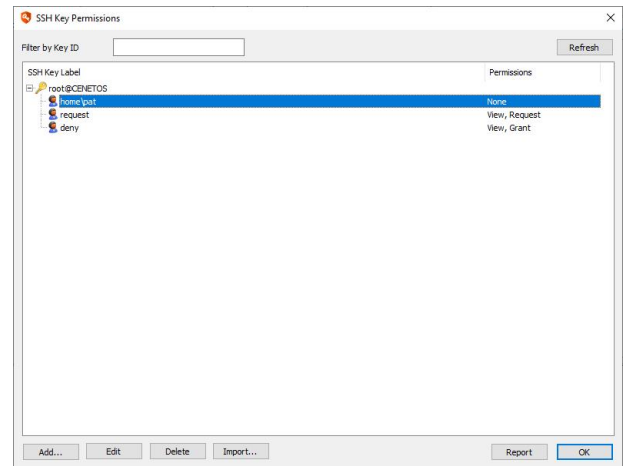


2. From the **SSH Key Label** list, click on the key you wish to assign an identity.
3. Click **Add**.
4. From the **Select Enrolled Identities** dialog, choose the identity you wish to have permission to use the SSH key.



**Note:** To add a new identity, click the **Add Identity** button, and complete the **Add Delegation Identity** information.

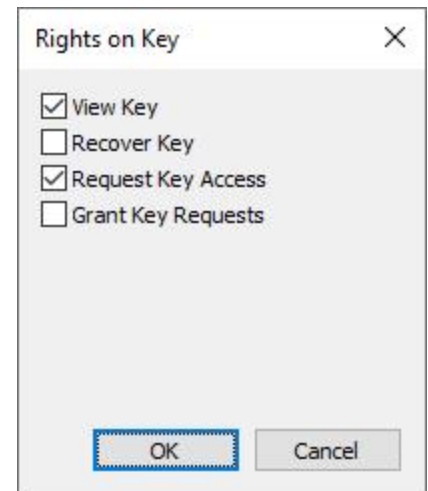
5. Select a key and identity, and click **OK**. In the **SSH Key Label** list, the identity name appears under the key.



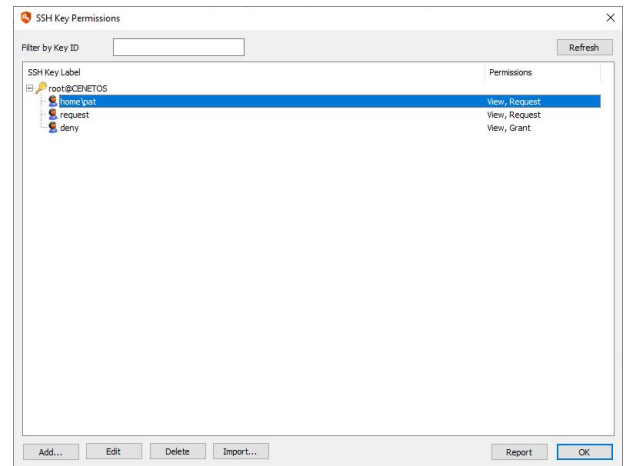
## Manage SSH Key Permissions

The default SSH key permissions for an identity are **View Key** and **Recover Key**. Other possible permissions are **Request Key Access** and **Grant Key Requests**. An identity may also have no permissions.

1. To change the permissions, select the identity and click **Edit**. A window showing permissions appears.



2. Check or uncheck the desired permissions and click **OK**. The window closes and the updated key permissions display.



3. To remove an identity's permission to use the SSH key, uncheck all permissions or select the identity and click **Delete** to remove it from the key.

To generate a report about SSH Key permission data, click **Report**, and select the report parameters from the **Report Generator** dialog.

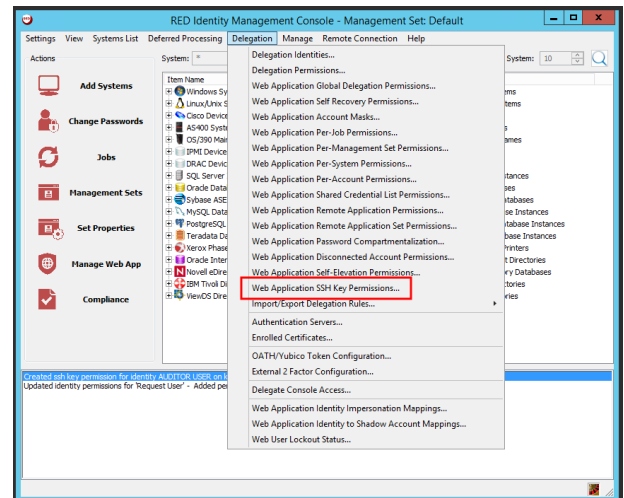
## Import Identity and SSH Key Permission Information

To assign multiple identities to multiple keys, you can upload a file listing the permissions. The file must contain one SSH Key permission entry per line, and it must be formatted as follows:

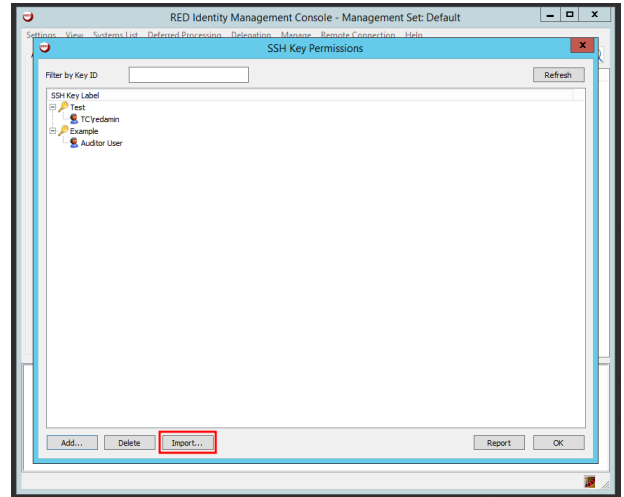
```
Identity Name, SSH Key Label
```

Follow the steps below to import SSH Key permissions into the system.

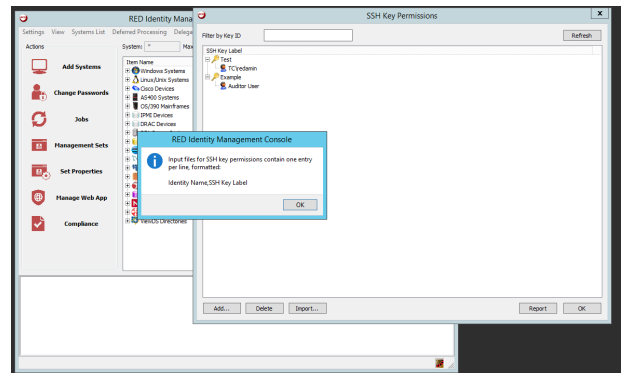
1. In the management console, go to **Delegation > Web Application SSH Key Permissions....**



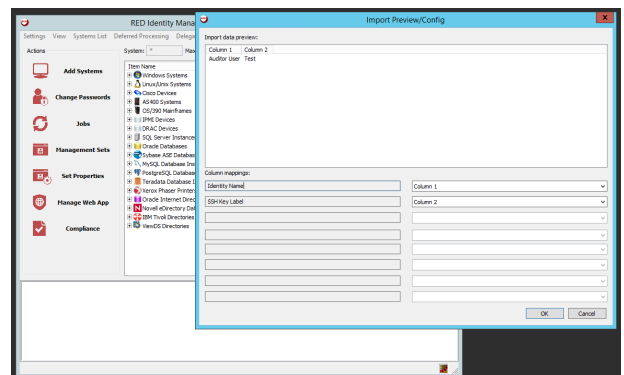
2. Click **Import...** A message appears, indicating how to format the input file.



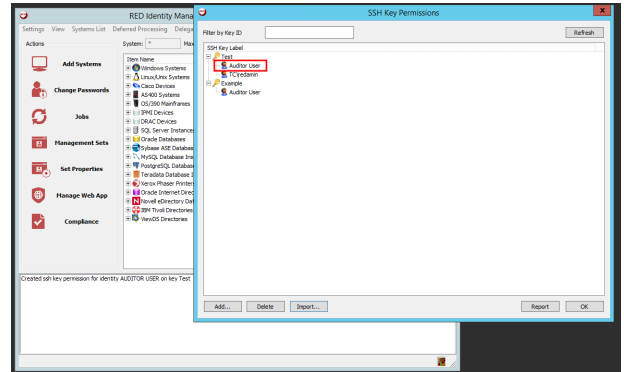
3. Click **OK**.



4. Locate and select the input file from your system. Click **Open**.
5. From the **Import Preview/Config**, verify the column mappings are correct. If needed, under **Column Mappings**, you can change the column associations by changing the selections in the dropdowns. Before confirming the import, make sure your identity and SSH key labels are correctly paired.



- When finished, click **OK**, and your permissions appear in the **SSH Key Label** list. The default permission is **View Access**, which can be changed for individual identities as required.

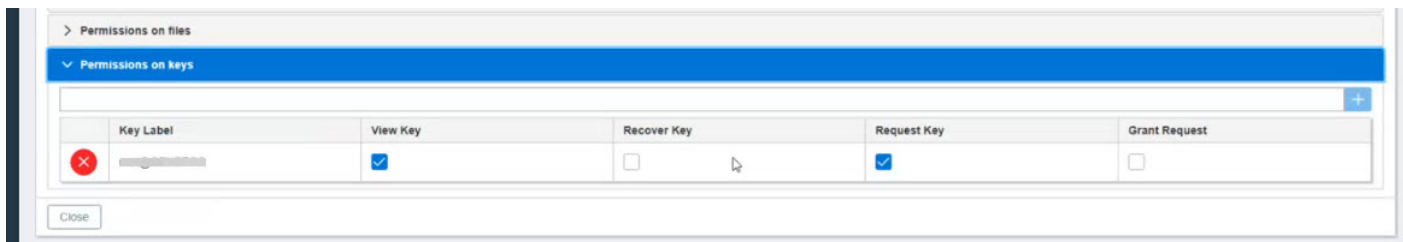




## Assign SSH Key Permissions in the Web Application

SSH Key permissions are used to assign a specific identity access to an SSH key and manage the rights for that identity. SSH Key permissions can be defined in the Web Application as described here, via an API, or in the Management Console.

1. In the web application menu, go to **Settings > Delegation**.
2. Click the ellipsis at the left end of the row for the **Identity Name** to which to grant key permissions.
3. Click **Edit Identity** to change the key permissions.
4. Click **Permissions on keys** to expand the section. It might be necessary to scroll down to see the section.
5. The default SSH key permissions for an identity are **View Key** and **Recover Key**. Other possible permissions are **Request Key Access** and **Grant Key Requests**. Check or uncheck the desired permissions.



## Request and Authorize SSH Keys from the Web Application

Users with access can request an SSH key.

### Request an SSH Key

1. Log into the web application.
2. Click **Passwords > Keys**. Use the filters at the top of the page to help locate keys.
3. Click the menu button for the desired key, and select **Request Key**.
4. Complete the Request Notification:
  - **Request Comment**: The reason for the request.
  - **Requestor Email Address**: The email address for notification that a response to the request has been issued.
  - **Requestor Display Name**: Your name.
  - Select the **Request Type**.
  - Set the time for the request.
  - Click **Request**.
5. The **Request Notification** closes and the **Request Status** displays.
6. You can return to the **Request Status** by clicking **My Key Requests** at the top of the page.
7. Another user, with appropriate permissions to grant or deny key requests, sees the request under **Passwords > All Key Requests**. After clicking **Accept** or **Deny**, this user can enter a comment.
8. The original user can view the response under **My Key Requests**.
9. Click the key icon to view the key details for approved requests. The key details can be downloaded.
10. The key is only released until the key details display is closed.



**Note:** Unlike passwords, keys are not checked out. Once you close the key details display, you must request the key again if it is required again.



**Note:** Users must have appropriate access to request keys. Please see "[Assign SSH Key Permissions in the Web Application](#)" on page 421 or "[Assign SSH Key Permissions in the Management Console](#)" on page 416.

# Manage Shared Credential List Permissions

Shared Credential Lists (SCLs) are lists of non-managed passwords meant to be shared with other users. Once SCLs are created and passwords imported, permissions must be granted to identities to permit various types of access to the SCL. Permissions can be granted via the management console, web application, or programmatically.

For more information on performing these actions in the management console or web application, please see...

- Shared Credential List Permissions in the Management Console
- Shared Credential List Permissions in the Web Application

## Adding SCL Permissions Programmatically

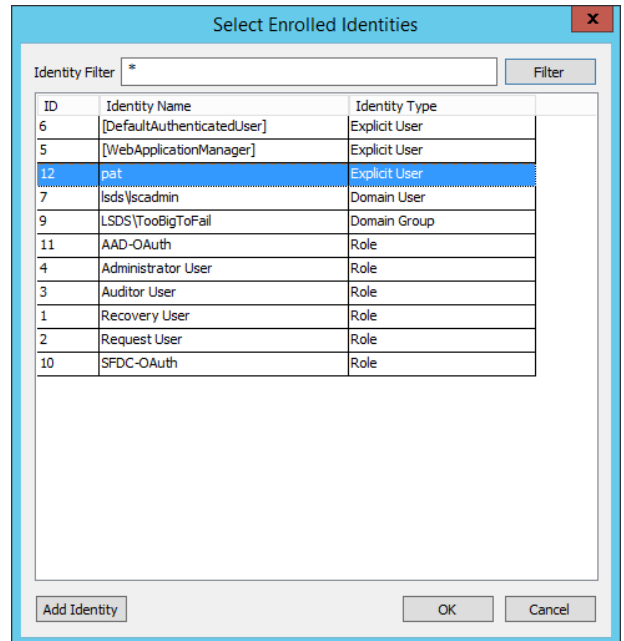
See the Programmers Guide for more information.

- From PowerShell, call Set-LSDelegationOnSharedCredentialList.
- From SOAP, call DelegationOps\_SetPermissionForSharedCredentialList.
- From REST, call /REST/Delegation/SharedCredentialList.

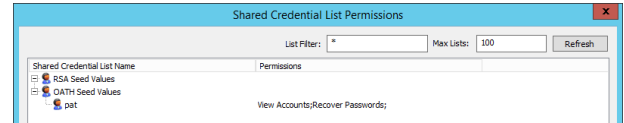
# Shared Credential List Permissions in the Management Console

## Adding SCL Permissions via the Management Console

1. In the management console, go to **Manage | Web Application Shared Credential List Permissions**.
2. Right-click on the target SCL and select **Add Identity to Password List**.



3. Select one or more target identities and click **OK**. The identity will now be visible under the SCL with no permissions.

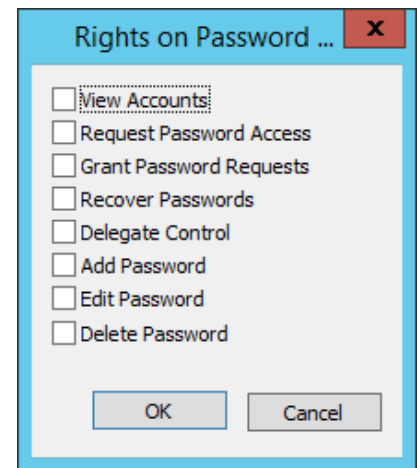


4. Right-click on the identity and select **Edit Permissions of Identity**.
5. Select the desired permissions:

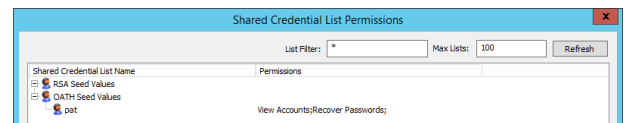


**Note:** Permissions are cumulative. If a given users is associated with one or more identities, all sets of permissions will be evaluated for the user.

- **View Accounts** - Allows the user to see the list of credentials within the list, but provides no further access.
- **Request Password Access** - Allows the user to request the access to the password. Access must be granted by a user who has Grant Password Requests.
- **Grant Password Requests** - Allows the user to grant or deny requests made for a password in the SCL.
- **Recover Passwords** - Allows the user to retrieve the password without have to follow a request workflow.
- **Delegate Control** - Allows the user to manage the delegations of the SCL. Be careful when granting this permission as this user may also grant themselves any other permission on the SCL.
- **Add Password** - Allows adding a credential to the SCL.
- **Edit Password** - Allows modifying the attributes and password for a credential in the SCL.
- **Delete Password** - Allows deleting any credential from the SCL.



6. Click **OK**.
7. Click **OK** to finish adding the entry.



## Import Permissions

Permissions can be imported in bulk. The expected format is:

```
SharedCredentialList, IdentityName, PermissionBits
```

When importing these delegations, the logon user needs to pre-exist.

Following is the list of permission bits:

- **View Accounts** = 1
- **Request Password Access** = 64
- **Grant Password Requests** = 128
- **Recover Passwords** = 256
- **Delegate Control** = 2

- **Add Password** = 4
- **Edit Password** = 8
- **Delete Password** = 16

Permissions are cumulative. To calculate the permissions a given identity should have, add the bit values together to come up with a final number. For example, an identity named **demo\bob** that can **view the list accounts, and recover passwords** will have a bit mask of 1 + 256 for a combined value of 257. The target list name is External Vendor Accounts. Thus the import entry would be:

```
External Vendor Accounts,demo\bob,257
```


## Shared Credential List Permissions in the Web Application

### Adding SCL Permissions via the Web Site

Permissions can be added to an SCL via the web site by any user that has any of the following permissions:

- **All Access** - Can do anything in the web site.
- **Manage External Lists** - Can do anything to any SCL.
- **Change Delegation (Delegate Control)** - Ability to delegate controls on a specific SCL.

1. In the web application, go to **Passwords | Edit Shared Password Lists**.
2. Click the **Edit permissions** button (4 horizontal lines) next to the target SCL.
3. Select the target identity from the Delegation Identity dropdown list, then click the **Add Permissions for Identity** button (+).
4. Enable the check box for the desired permissions:

Delegation Identity		[DefaultAuthenticatedUser] +									
Identity Name	Identity Type	View List	Change Delegation	Add Password	Edit Password	Delete Password	Recover Password	Request Password	Grant Password Request		
pat	Explicit User	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	



**Note:** Permissions are cumulative. If a given users is associated with one or more identities, all sets of permissions will be evaluated for the user.

- **View List** - Allows the user to see the list of credentials within the list, but provides no further access.
- **Change Delegation** - Allows the user to manage the delegations of the SCL. Be careful when granting this permission as this user may also grant themselves any other permission on the SCL.
- **Add Password** - Allows adding a credential to the SCL.
- **Edit Password** - Allows modifying the attributes and password for a credential in the SCL.
- **Delete Password** - Allows deleting any credential from the SCL.
- **Recover Passwords** - Allows the user to retrieve the password without have to follow a request workflow.
- **Request Password** - Allows the user to request the access to the password. Access must be granted by a user who has Grant Password Requests.
- **Grant Password Request** - Allows the user to grant or deny requests made for a password in the SCL.

To remove a permission, deselect the appropriate check box. To remove an identity, click the red X at the end of the permission row.

## Retrieve Passwords

This section describes accessing managed and stored passwords.



*For information on sharing personal passwords, please see "Share Personal Passwords" on page 449.*

Using the Application Launcher to establish a privileged session is described in detail in the Application Launcher & Session Recording Guide.

Passwords can be retrieved from the web application, management console, or programmatically.

All managed passwords and shared credential list passwords are encrypted and the encrypted value is written to the primary data store. These passwords can be viewed from the management console at any time by a console administrator. Moreover, a history of every password ever added to the password store or attempted are also stored. By default, this information is kept indefinitely.

## Retrieve Managed Passwords

There are five ways to retrieve a managed password:

- Retrieve the password from the web application
- Request access to the password from the web application
- Retrieve the password programmatically
- Request access to the password programmatically
- View the passwords from the management console

The delegations required to perform a password retrieval or request (along with granting or denying that request) through the web application or web service are identical.

Viewing passwords from the management console requires the user has access to the management console.



*For more information on performing these actions in the management console or web application, please see:*

- *"Retrieve Managed Passwords from the Web Application" on page 429*
- *"Request and Grant Access to Managed Passwords from the Web Application" on page 431*
- *"View Managed Passwords in the Management Console" on page 433*

## Programmatic Access to Managed Passwords

See the programmers reference for more information.

### Password Retrieval

- From Powershell, call...
  - Get-LSPasswordWithReason
  - Get-LSPasswordWithoutReason
  - Get-LSPasswordIgnoreCheckout
- From SOAP, call...
  - AccountStoreOps\_StoredCredential\_CheckOut
  - AccountStoreOps\_StoredCredential\_GetIgnoreCheckout
- From REST, call...
  - /REST/StoredCredential
  - /REST/StoredCredential/IgnoreCheckout

## Password Request

- From Powershell, call...
  - **Request** - Set-LSPasswordRequest
  - **Get list of requests** - Get-LSListofPasswordRequests
  - **Grant** - Grant-LSPasswordRequest
  - **Deny** - Deny-LSPasswordRequest
- From SOAP, call...
  - **Request** - AccountStoreOps\_StoredCredential\_Request
  - **Get List of Requests** - AccountStoreOps\_StoredCredential\_ListPasswordRequests
  - **Grant** - AccountStoreOps\_StoredCredential\_GrantRequest
  - **Deny** - AccountStoreOps\_StoredCredential\_DenyRequest
- From REST, call...
  - **Request** - /REST/StoredCredential/Request
  - **Get list of requests** - /REST/StoredCredentials/Request
  - **Grant** - /REST/StoredCredential/Request/Grant
  - **Deny** - /REST/StoredCredential/Request/Deny



## Retrieve Managed Passwords from the Web Application

To retrieve a password, the identity logged into the web application must have appropriate permissions to retrieve the password.

**i** For more information on creating delegations, please see "[Manage Identities and Delegations for Password and System Access](#)" on page 374.

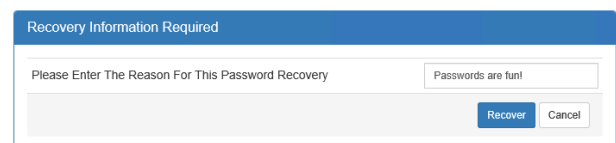
You may add frequently accessed passwords to your favorites dashboard panel by clicking the **Add to favorites** button for the account or on the recovered password page.

After selecting the **Recover Password** option, you may be prompted for a recovery comment and incident, if these options are configured and enabled. Once the recovery criteria is met, you can view the password.

You can view, copy to clipboard, show phonetic pronunciation of the characters in the password, extend the checkout, or check the password back in.

### Retrieve a Password

1. In the web application, go to **Passwords > Managed**. Use the filters at the top of the page to help locate credentials. Alternatively...
  - Go to **Systems** and **Stored Passwords** for the target system. This will switch the user to the **Managed Passwords** page, filtered for just the one system.
  - Go to **Accounts**. Any account with a stored password will have a **Stored Password** button (blue user icon). Click this button to switch the user to the **Managed Passwords** page, filtered for just the one system and that one account.
2. Click the **ellipsis** button for the account, and then select **Recover Password** to retrieve the password.
3. Other options available for an account based on system configuration and user permissions include:
  - **SSH (>\_)**: Use the mind-term Java-based control to establish a session to the target system.
  - **RDP**: The Microsoft ActiveX control or optionally application launcher to launch an RDP session to the target system.
  - **Account History (list)**: Shows all previously managed passwords (limited by password history settings) for this account.
  - **Account Activity (clock)**: Shows all previous accesses to this account.
  - **Edit Password (pencil)**: Change the stored password for this account. This does not force a password change of the account on the system.
  - **Edit Details (pencil on paper)**: Edit information such as comment, description, and web site link.
  - **Delete Password (X)**: Delete the stored/managed password for this account. This does not delete password history for the account.
4. You may be prompted for a recovery comment or incident number. Supply this information to continue, and then click **Recover**.



- The password is displayed. The website link and description is populated on this dialog only if the account's metadata was previously updated prior to this recovery.

**Password Recovery**

This checkout will expire in 0 days, 02 hours and 00 minutes. ✕

Namespace	[Linux]	📄
System Name	UBUNTU14	📄
Account Name	root	📄
Comment		📄
Website Link	<a href="#">🔗</a>	📄
Description		📄
Password	👁️ ••••••••••	📄

A
↻
♥

Check In

## Other Actions on this Page

- Each field provides a copy button. Clicking the **copy** button copies the field information to the user's clipboard.
- Click the **Phonetics** button (A) to display the phonetic pronunciation of the password characters in order.
- Click the **extend checkout** button (circular arrow) to extend the checkout. The checkout extension interval is defined in the web application settings.
- Add this password to the favorites dashboard by clicking the **Add to favorites** button (heart).

When done with the password, you can click the **Check In** button. This will check the password in ahead of schedule. If no action is taken, the password will be checked in automatically when the check out expires. The check out expiration is listed at the top of the page.

## Request and Grant Access to Managed Passwords from the Web Application

To request a password, the identity must have been granted appropriate permissions to request the password. In turn, another identity must have been granted the ability to grant or deny that request. See "[Manage Identities and Delegations for Password and System Access](#)" on page 374 for more information on creating delegations.

Once the user has been granted the appropriate access, the user will log into the web application, and request the password. A grantor will/may receive a notification and subsequently grant or deny the request. In either a grant or deny scenario, the requester will receive a notification. If the request is granted, the original requester may then retrieve the password.

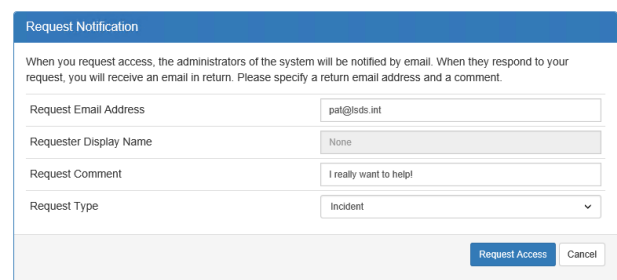
A user may add a frequently accessed password to their favorites dashboard panel by clicking the **Add to favorites** button (heart icon) next to the account or on the recovered password page.

After the user clicks the recover password button (blue credit card icon), they may be prompted for a recovery comment and/or incident. These are web application configurations that must be enabled and configured. Once the recovery criteria is met, the user will be presented with the password.

The user can view, copy to clipboard, show phonetic pronunciation of the characters in the password, extend the checkout, or check the password back in.

### Performing a Password Request

1. Log into the web application.
2. Click **Passwords > Managed**. Alternatively...
  - The user can go to **Systems** and **Stored Passwords** button for the target system. This will switch the user to the Managed Passwords page, filtered for just the one system.
  - The user can go to **Accounts**. Any account with a stored password will have a **Stored Password** button (blue credit card icon). Clicking this button will switch the user to the Managed Passwords page, filtered for just the one system and that one account.
3. Click the **Request Password** button (circular arrow).
4. On the request notification page, supply the following information:
  - **Request Email Address** - The user's email address will be filled in if the user has previously supplied an email address for a previous request, the email address was filled in on the identity's properties in the global delegations, or it could be read from Active Directory (if appropriate). The user may change this email address by supplying a different email address.
  - **Requester Display Name** - The user's display name as derived from Active Directory, if appropriate. This value cannot be changed by the user.
  - **Request Comment** - The user will supply a comment for the request. This will be added to the notification emails and program logs.
  - **Request Type** - Incident or Change. This affects who will be notified of the request if per management set, per system, or per account delegations are configured to "notify on incident" or "notify on change".



**Request Notification**

When you request access, the administrators of the system will be notified by email. When they respond to your request, you will receive an email in return. Please specify a return email address and a comment.

Request Email Address	<input type="text" value="pat@ltds.int"/>
Requester Display Name	<input type="text" value="None"/>
Request Comment	<input type="text" value="I really want to help!"/>
Request Type	<input type="text" value="Incident"/>

- The outstanding request will be added to the identity's Current Activity panel if the panel is enabled (**Settings | Session Settings > Main Panel Configuration > Current Activity**). The outstanding request may also be viewed by the requester by going to **Requests > My Managed Password Requests**. The request status will be **Waiting for Validation** until the request is granted or denied (or expires).

Request ID	System Name	Account Name	Request Status
1	DBSMASH2008.Isds.int	Administrator	Waiting for Validation



**Note:** There is no way to cancel an outstanding password request. It must expire or be denied. Also, an identity cannot renew or modify their request once made. It must expire or be denied before the user can issue a new request for the same credential.

## Grant or Deny a Password Request

Once a request is made, the request has a default lifetime of 60 minutes (per web application password configuration: Password request timeout Window). If the request is not granted within that window, it is automatically denied. Once the request has granted, the requester has a default window of 60 minutes (per web application password configuration: Password request grant timeout Window) to retrieve the password. If the password is not checked out before the window expires, the user must re-request the password. If the user checks out the password before the window expires, normal checkout timeout policies will apply regarding duration of checkout.

- Log into the web site as a user who can grant the request.
- Go to **Requests > All Managed Password Requests**. All outstanding requests will have a **Request Result** of Waiting for Validation.
- Choose to grant the request or deny the request using the grant (green thumb up) or deny (red thumb down) buttons.
- Upon granting or denying the request, the granter must supply a grant or deny comment. This comment will be sent to the requester in the grant or deny notification email. It will also be added to the program audit logs.
- Click **Grant** or **Deny Request**. You will then be returned to your default page.
- To view the status of the grant, go to **Requests > All Managed Password Requests**. The status of the request will have changed to reflect the grant or deny action.

**Grant Notification**

When you grant an access request, the user who requested the access will be notified by email. The email alert that will be sent notifying the user that their request has been processed will include the comment you specify here.

Grant Comment

Request ID	Requested By	System Name	Account Name	Request Result
1	pat	DBSMASH2008.Isds.int	Administrator	Password Request Granted

## Password Recovery following a Successful Password Request

Regardless of the password access being granted or denied, the user will receive an email notification at the Request Email address filled in during the initial request. If the password request was granted, the user will simply check out the password.

If the identity's Current Activity panel if the panel is enabled (**Settings > Session Settings > Main Panel Configuration > Current Activity**), the password retrieval may be initiated by clicking the recover password (down arrow) from here. The outstanding request may also be viewed by the requester by going to **Requests > My Managed Password Requests**. The request status will display a **Recover Password** button. The user may also navigate to the managed passwords page where the password request icon has changed to a recover password icon.

See "[Retrieve Managed Passwords from the Web Application](#)" on page 429 for more information on the process to follow for recovering a password from the web application.

## View Managed Passwords in the Management Console


All managed passwords and passwords from shared credential lists (but not the personal vault) can be retrieved from the management console.

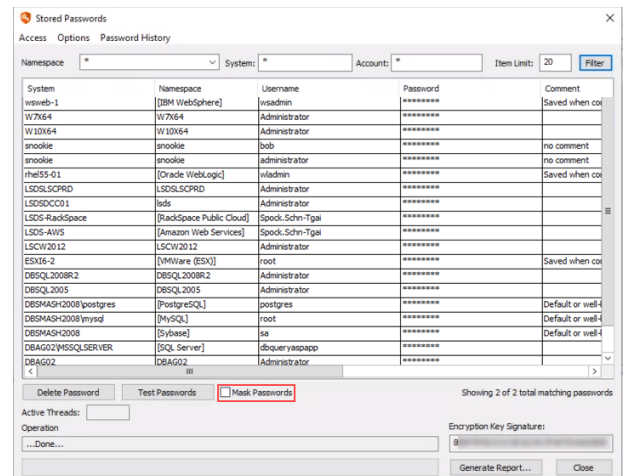
 For information on sharing personal passwords, please see "[Share Personal Passwords](#)" on page 449.

### View Passwords


The passwords shown in the stored passwords dialog are the current (last successfully set) passwords of the accounts in the list.

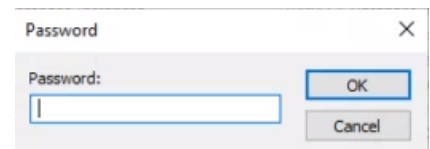
1. Go to **Manage > View Stored Managed Passwords**.
2. Double-click a password to view that stored password or deselect the **Mask Passwords** option to view all of the stored passwords.

 **Note:** Use the filters at the top of the dialog to help locate specific credentials.



3. Enter the recovery access password.

 **Note:** The first time an administrator attempts to access this function, they are prompted to create a recovery access password.



4. Use the **Generate Report** button to generate a report on the contents of the dialog.

### Validate Stored Passwords

Stored passwords can be tested from this dialog. Select a password (Windows or MS SQL or Linux/Unix), and then click **Test Passwords**. This will initiate a connection from this management console which will attempt a login with the selected account.

### Change the Default Recovery Access Password

1. Go to **Manage > View Stored Managed Passwords**.
2. Enter the recovery access password.



**Note:** *The first time an administrator attempts to access this function, they are prompted to create a recovery access password.*

3. Go to **Access > Change Recovery Access Password**.
4. Set the new password.

## Retrieve Shared Managed and Stored Passwords

There are four ways to retrieve a password that has been shared from managed and stored passwords:

- Retrieve the password from the web application.
- Request access to the password from the web application.
- Retrieve the password programmatically.
- Request access to the password programmatically.

The delegations required to perform a password retrieval or request (along with granting or denying that request) through the web application or web service are identical.

Viewing passwords from the management console requires the user have access to the management console.



*For more information on performing these actions in the management console or web application, please see...*

- *"Retrieve Shared Passwords from the Web Application" on page 437*
- *"Request and Grant Access to Shared Passwords from the Web Application" on page 439*
- *"View Shared Passwords in the Management Console" on page 441*

## Programmatic Access to Managed Passwords

See the programmers reference for more information.

### Shared Password Retrieval

- From PowerShell, call...
  - Get-LSSharedCredential
  - Get-LSSharedCredentialIgnoreCheckout
- From SOAP, call...
  - AccountStoreOps\_StoredCredential\_CheckOut
  - AccountStoreOps\_StoredCredential\_GetIgnoreCheckout
- From REST, call...
  - /REST/StoredCredential
  - /REST/StoredCredential/IgnoreCheckout

### Shared Password Request

- From Powershell, call...
  - **Request** - Set-LSSharedCredentialRequest
  - **Get list of requests** - Get-LSListOfSharedCredentialRequests
  - **Grant** - Grant-LSSharedCredentialRequest

- **Deny** - Deny-LSSharedCredentialRequest
- From SOAP, call...
  - **Request** - AccountStoreOps\_SharedCredential\_Request
  - **Get List of Requests** - AccountStoreOps\_SharedCredential\_ListPasswordRequests
  - **Grant** - AccountStoreOps\_SharedCredential\_GrantRequest
  - **Deny** - AccountStoreOps\_SharedCredential\_DenyRequest
- From REST, call...
  - **Request** - /REST/SharedCredential/Request
  - **Get list of requests** - /REST/SharedCredentials/Request
  - **Grant** - /REST/SharedCredential/Request/Grant
  - **Deny** - /REST/SharedCredential/Request/Deny



*For information on sharing personal passwords, please see "Share Personal Passwords" on page 449.*



## Retrieve Shared Passwords from the Web Application

To retrieve a shared credential password that has been shared from managed and stored passwords, the identity must have been granted appropriate permissions to retrieve the password. See ["Manage Identities and Delegations for Password and System Access"](#) on page 374 for more information on creating delegations.

**i** For information on sharing personal passwords, please see ["Share Personal Passwords"](#) on page 449.

Once the user has been granted the appropriate access, the user will log into the web application, and retrieve the password.

A user may add a frequently accessed password to their favorites dashboard panel by clicking the **Add to favorites** button (heart icon) next to the account or on the recovered password page.

After the user clicks the recover password button (down arrow), they may be prompted for a recovery comment and/or incident. These are web application configurations that must be enabled and configured. Once the recovery criteria is met, the user will be presented with the password.

The user can view, copy to clipboard, show phonetic pronunciation of the characters in the password, extend the checkout, or check the password back in.

### Performing a Password Retrieval

1. Login to the web application.
2. Click **Passwords > Shared Passwords List**.  
The user's filter will be set to All Lists which will show all passwords from all lists the identity has access to. The user can retrieve a password from the All Lists list or filter for a specific list.
3. Click the **Recover Password** button (blue credit card icon).
4. The user may be prompted for a recovery comment and/or incident number. Supply this information to continue, then click **Recover**.
5. The password will be displayed to the user.

System Name	Account Name	List Name	Last Change Time
Linux-Jonzz	rhenshaw	MidvaleHR	9/18/2018, 1:43:21 PM
Unix-Winn	wschott	MidvaleHR	9/18/2018, 1:42:47 PM
Win10-SuperGirl	kdanvers	MidvaleHR	9/18/2018, 1:40:27 PM
Mac107-Guardian	jolsen	MidvaleHR	9/18/2018, 1:39:12 PM

Recovery Information Required

Please Enter The Reason For This Password Recovery

Password Recovery

This checkout will expire in 0 days, 02 hours and 00 minutes.

Password List:   
 System Name:   
 Account Name:   
 Website Link:   
 Description:   
 Password:

## Other Actions on this Page

- Each field provides a copy button. Clicking the **copy** button will copy the field information to the user's clipboard.
- Click the **Phonetics** button (A) to display the phonetic pronunciation of the password characters in order.
- Click the **extend checkout** button (circular arrow) to extend the checkout. The checkout extension interval is defined in the web application settings.
- Add this password to the favorites dashboard by clicking the **Add to favorites** button (heart).

When done with the password, you can click the **Check In** button. This will check the password in ahead of schedule. If no action is taken, the password will be checked in automatically when the check out expires. The check out expiration is listed at the top of the page.

# Request and Grant Access to Shared Passwords from the Web Application

To request a shared credential password that has been shared from managed and stored passwords, the identity must have been granted appropriate permissions to request the password. In turn, another identity must have been granted the ability to grant or deny that request. See "[Manage Identities and Delegations for Password and System Access](#)" on page 374 for more information on creating delegations.

Once the user has been granted the appropriate access, the user will log into the web application, and request the password. A grantor may receive a notification and subsequently grant or deny the request. In either a grant or deny scenario, the requester will receive a notification. If the request is granted, the original requester may then retrieve the password.

A user may add a frequently accessed password to their favorites dashboard panel by clicking the **Add to favorites** button (heart icon) next to the account or on the recovered password page.

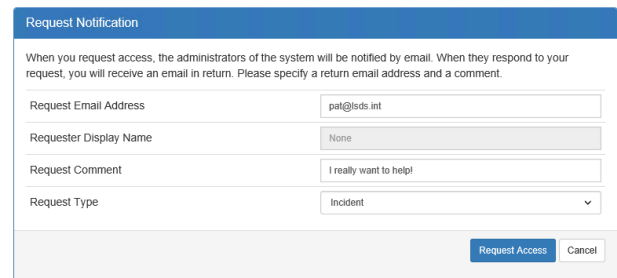
After the user clicks the recover password button (blue credit card icon), they may be prompted for a recovery comment and/or incident. These are web application configurations that must be enabled and configured. Once the recovery criteria is met, the user will be presented with a the password.

The user can view, copy to clipboard, show phonetic pronunciation of the characters in the password, extend the checkout, or check the password back in.

## Performing a Password Request

1. Login to the web application.
2. Click **Passwords > Shared Password Lists**.
3. Click the **Request Password** button (circular arrow).
4. On the request notification page, supply the following information:

- **Request Email Address** - The user's email address will be filled in if the user has previously supplied an email address for a previous request, the email address was filled in on the identity's properties in the global delegations, or it could be read from Active Directory (if appropriate). The user may change this email address by supplying a different email address.
- **Requester Display Name** - The user's display name as derived from Active Directory, if appropriate. This value cannot be changed by the user.
- **Request Comment** - The user will supply a comment for the request. This will be added to the notification emails and program logs.
- **Request Type** - Incident or Change. This affects who will be notified of the request if per management set, per system, or per account delegations are configured to "notify on incident" or "notify on change".



**Request Notification**

When you request access, the administrators of the system will be notified by email. When they respond to your request, you will receive an email in return. Please specify a return email address and a comment.

Request Email Address:

Requester Display Name:

Request Comment:

Request Type:

5. The outstanding request will be added to the identity's Current Activity panel if the panel is enabled (**Settings > Session Settings > Main Panel Configuration > Current Activity**). The outstanding request may also be viewed by the requester by going to **Requests > My Shared Password Requests**. The request status will be **Waiting for Validation** until the request is granted or denied (or expires).

Request ID	Password List	System Name	Account Name	Request Status	Request Time	Requested Time
2	OATH Seed Values	OATHSeedValue	robert	Waiting for Validation	4/24/2017 2:02:58 PM	Immediately

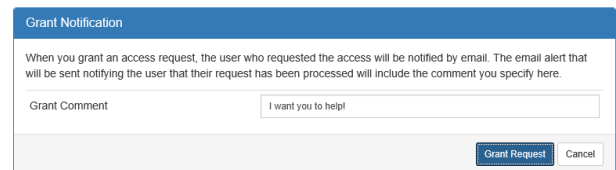


**Note:** There is no way to cancel an outstanding password request. It must expire or be denied. Also, an identity cannot renew or modify their request once made. It must expire or be denied before the user can issue a new request for the same credential.

## Grant or Deny a Password Request

Once a request is made, the request has a default lifetime of 60 minutes (per web application password configuration: Password request timeout Window). If the request is not granted within that window, it is automatically denied. Once the request has granted, the requester has a default window of 60 minutes (per web application password configuration: Password request grant timeout Window) to retrieve the password. If the password is not checked out before the window expires, the user must re-request the password. If the user checks out the password before the window expires, normal checkout timeout policies will apply regarding duration of checkout.

1. Login to the web site as a user who can grant the request.
2. Go to **Requests > All Managed Password Requests**. All outstanding requests will have a **Request Result** of Waiting for Validation.
3. Choose to grant the request or deny the request using the grant (green thumb up) or deny (red thumb down) buttons.
4. Upon granting or denying the request, the granter must supply a grant or deny comment. This comment will be sent to the requester in the grant or deny notification email. It will also be added to the program audit logs.
5. Click **Grant** or **Deny Request**. You will then be returned to your default page.
6. To view the status of the grant, go to **Requests > All Shared Password Requests**. The status of the request will have changed to reflect the grant or deny action.



Request ID	Requested By	Password List	System Name	Account Name	Request Result
2	pat	OATH Seed Values	OATHSeedValue	robert	Password Request Granted

## Password Recovery following a Successful Password Request

Regardless of the password access being granted or denied, the user will receive an email notification at the Request Email address filled in during the initial request. If the password request was granted, the user will simply check out the password.

If the identity's Current Activity panel if the panel is enabled (**Settings > Session Settings > Main Panel Configuration > Current Activity**), the password retrieval may be initiated by clicking the recover password (down arrow) from here. The outstanding request may also be viewed by the requester by going to **Requests > My Shared Password Requests**. The request status will display a **Recover Password** button. The user may also navigate to the managed passwords page where the password request icon has changed to a recover password icon.

See "[Retrieve Shared Passwords from the Web Application](#)" on page 437 for more information on the process to follow for recovering a password from the web application.

## View Shared Passwords in the Management Console

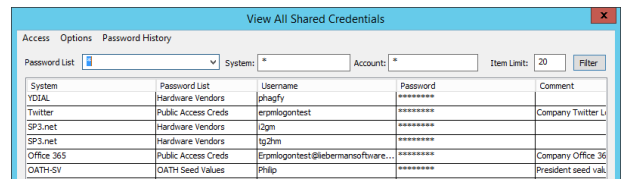
All managed passwords and passwords from shared credential lists (but not the personal vault) can be retrieved from the management console.

To launch the management console, the launching user must be a local host admin, have proper access to the database (when using integrated authentication to the database), pass any other console delegation MFA checks, and also have been granted the ability to do so (default) by the console delegations.


### Viewing Shared Passwords

The passwords shown in the shared passwords dialog are the current (last successfully set) passwords of the accounts in the list.

1. Go to **Manage > View All Shared Credentials**.
2. Enter the master password.



System	Password List	Username	Password	Comment
System	Hardware Vendors	phagfy	*****	
YDIAL	Public Access Creds	erpmlgontest	*****	Company Twitter L
Twitter	Hardware Vendors	l2gm	*****	
SP3.net	Hardware Vendors	tg2hm	*****	
Office 365	Public Access Creds	Erpmlgontest@liebermansoftware...	*****	Company Office 36
DATH-SV	DATH Seed Values	Philip	*****	President seed val

 **Note:** The first time an administrator attempts to access this function, they are prompted to create a recovery access password.


3. Double click a password to open and view the stored password.

Use the filters at the top of the dialog to help locate the credentials.

Use the **Generate Report** button to generate a report on the contents of the dialog.

### Changing the Default Master Password

1. Go to **Manage > View Stored Managed Passwords**.
2. Enter the master password.

 **Note:** The first time an administrator attempts to access this function, they are prompted to create a recovery access password.

3. Go to **Access > Change Recovery Access Password**.
4. Set the new password.

# Password History

Password history is automatically enabled for all managed accounts, shared credential list credentials, and personal password store credentials. Password history can be viewed from the management console or web application. Password history is currently not available through programmatic methods.

Historical passwords are useful when restoring backups or during troubleshooting. By default, history is enabled and no storage limitations are imposed.

## Password History Settings

Password history settings are controlled from the management console. To view or set password history settings:

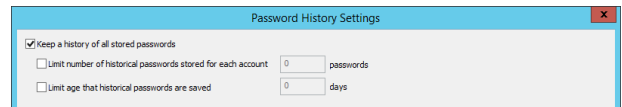
1. Go to **Manage > View Stored Managed Passwords**.
2. Enter the recovery access password.



**Note:** The first time an administrator attempts to access this function, they are prompted to create a recovery access password.

3. On the **Stored Passwords** dialog, go to **Password History > Change Password History Settings**.
4. Note the following settings:

- **Keep a history of all stored passwords:** When enabled (default), all passwords ever stored in the secure password store or attempted during a password change job will be kept in password history.
- **Limit number of historical passwords for each account:** When enabled (default is disabled), this will limit each managed account to only a specified number of historical passwords. Older passwords will be removed automatically.
- **Limit age that historical passwords are saved (days):** When enabled (default is disabled), this ensures that no historical passwords over a certain age range will be retained.



## View Historical Passwords In the Management Console

Password history can be viewed from the management console.

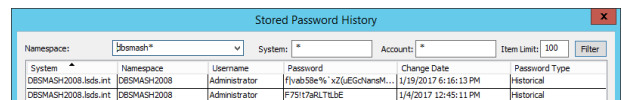
1. Go to **Manage > View Stored Managed Passwords**.
2. Enter the master password.



**Note:** The first time an administrator attempts to access this function, they are prompted to create a recovery access password.

3. On the **Stored Passwords** dialog, go to **Password History > View Managed Password History**.

Passwords that were successfully set previously will be listed in this dialog with **Password Type** set to **Historical**. Passwords that were not successfully set previously but were attempted will be listed in this dialog with **Password Type** set to **Attempted**.



System	Namespace	Username	Password	Change Date	Password Type
DBSMASH2008.lcbs.int	DBSMASH2008	Administrator	Fivab5be%*2(EGNameM...	1/19/2017 6:16:13 PM	Historical
DBSMASH2008.lcbs.int	DBSMASH2008	Administrator	F7517aRLTtDE	1/4/2017 12:45:11 PM	Historical

## Viewing Historical Passwords In the Web Application

Password history can be viewed from the web application by users with either of the following delegations:

- All Access
- View Password History

1. Log into the web application.
2. Go to the target account in either the Managed Passwords area or shared credential list.
3. Click the **Account History** button.

(DBSMASH2008.lsd.int) DBSMASH2008\Administrator	
Last Change Time	Password
1/19/2017 6:16:13 PM	!vab58e% xZ(uEGcNansM-HBP%ymo?+(IS3.)2fr@60Y9Z0[v8yP*x-\$BPHJA
1/4/2017 12:45:11 PM	F75H7aRLTLbE

## Retrieve Personal Passwords

A personal password can be retrieved using the web application or it can be retrieved programmatically.

The delegations required to perform a password retrieval or request, along with granting or denying that request through the web application or web service are identical.

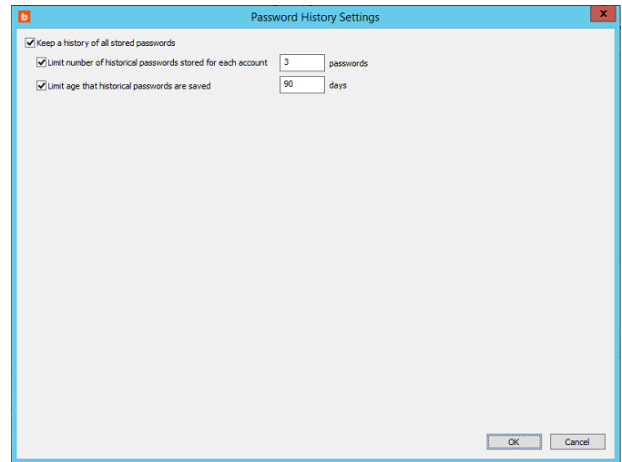
Password history is an option that must be enabled in the application settings:

1. In the management console go to **Manage > View Stored Managed Passwords**.
2. Enter the recovery access password.



**Note:** *The first time an administrator attempts to access this function, they are prompted to create a recovery access password.*

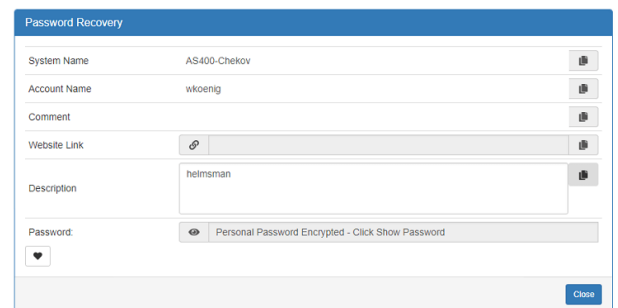
3. Make sure the three options are enabled and have non-zero values.



4. Go to **Password History > Change Password History Settings**.

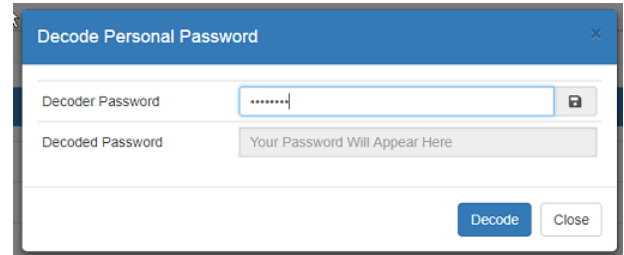
## Retrieve Personal Passwords Using the Web Application

1. Go to **Passwords > Personal**.
2. Click the **ellipsis** button for the credential, and then select **Recover Password**.
3. Click the eye icon to display the icon. If the credentials require a decoder password, enter the decoder password and click the **Decode** button. The password will be displayed to the user.





4. To the right of each field is a copy button. Click that button to copy the data in the corresponding field.



## Programmatic Access to Personal Passwords

- From SOAP, call AccountStoreOps\_PersonalCredential\_CheckOut.
- From REST, call /REST/PersonalCredential.

## Work with Compartmentalized Passwords (Four Eyes)

The following information applies to any and all platforms or stored credentials.

Privileged Identity can segregate a password into multiple pieces such that multiple people will need to check out their piece of a password in order to use the password. This is commonly called the two-person password rule or the four eyes principle (as in it takes two people, or 4 eyes) to recover a password. In Privileged Identity this feature is called password compartmentalization. In fact, the password may be compartmentalized into as many pieces as their characters in the password, which is why the name of the feature is not limited to the name 4 Eyes. For example, a 15 character password can be divided into 15 pieces requiring 15 people to retrieve the entire password.



**Note:** A password may be broken apart into multiple pieces at any point in its life cycle and also rejoined into one piece at any point in its life cycle.

### Compartmentalized Password Slices

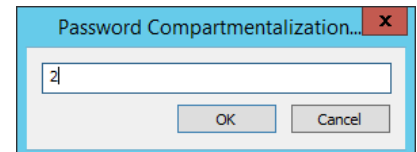
To setup password compartmentalization at job runtime, open the **Password Settings** tab and select the check box for **Generate compartmentalized password segments**, then define the number of segments. Do not set more segments than there are characters in the password. If the password is an odd number of characters, the remainder characters will go in the last segment.

Once a password has been compartmentalized, to undo the compartmentalization requires a job re-randomize or otherwise reset the password to a value where compartmentalized settings are not enabled on the job.

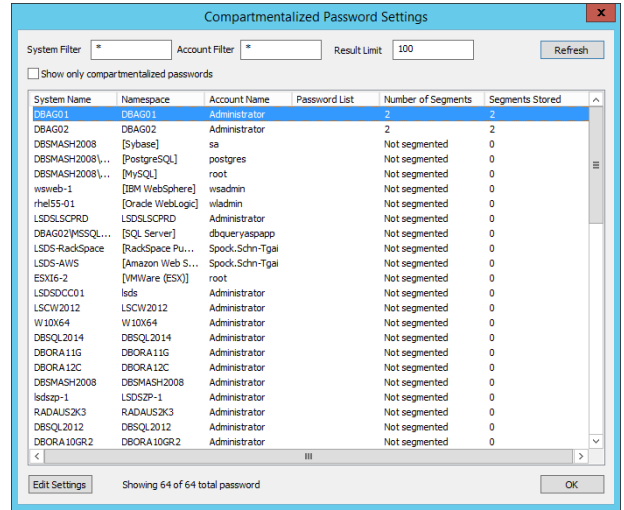
### Compartmentalized Password

To compartmentalize a password that is already stored in the secure password store, use the following steps:

1. From the management console, choose **Manage | Password Compartmentalization Settings**.
2. Select the account/password in question and click **Edit Settings** in the lower left corner of the dialog.
3. Enter the number of segments desired, then click **OK**.



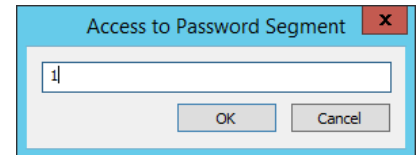
- If the password was not previously segmented, the password will be re-divided into that number of segments.



## Gaining Access to Compartmentalized Passwords

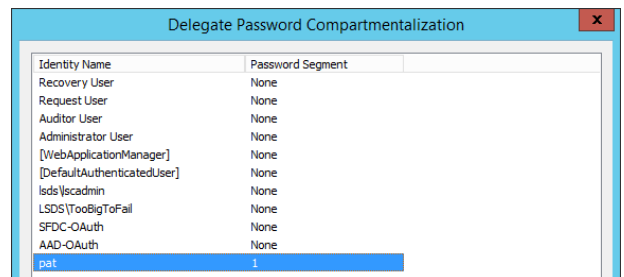
To gain access to a compartmentalized password, delegations must be configured for each segment. All Access identities have all access to the entire password without further management required.

- In the management console, go to **Delegation | Web Application Password Compartmentalization**.
- Select the identity to grant access to a segment to and click **Edit**. Note that the user still requires access to request or retrieve the password via normal delegations. This delegation is providing that access to a specific segment.
- Type in the number of the segment to which they should have access to, then click **OK**.



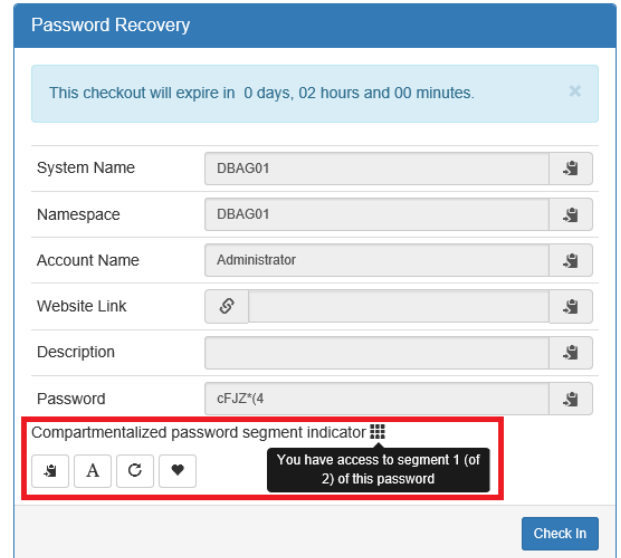
- The **Password Segment** column now indicates the password segment the identity will have access to. To remove the segment assignment, simply select the identity and click the **Remove** button.

When the user logs into the web application and opens the passwords area, as usual they will see all systems and accounts that they should have access to. However, the icons for segmented accounts will appear differently and other options that would normally be present, such as RDP or SSH will no longer appear, even if the user would otherwise have those permissions granted.



Once the user retrieves their segment, a note will indicate they are viewing a password segment and which segment that is.

When a user checks out their portion of a password, all the standard rules apply regarding checkout duration, password re-randomization, checkout to group, and so on, whether or not any other portions of the password have actually been checked out.




The screenshot shows the 'Password Recovery' interface. At the top, a blue header contains the title 'Password Recovery'. Below the header is a light blue notification bar stating 'This checkout will expire in 0 days, 02 hours and 00 minutes.' with a close icon. The main area contains several input fields: 'System Name' (DBAG01), 'Namespace' (DBAG01), 'Account Name' (Administrator), 'Website Link' (with a lock icon), 'Description' (empty), and 'Password' (cFJZ\*(4)). Below these fields is a section titled 'Compartmentalized password segment indicator' with a grid icon. This section contains a row of icons: a lock icon, the letter 'A', a refresh icon, and a heart icon. A tooltip points to the 'A' icon, displaying the text 'You have access to segment 1 (of 2) of this password'. At the bottom right of the interface is a blue 'Check In' button.

# Share Personal Passwords

This section describes sharing and accessing shared personal passwords.

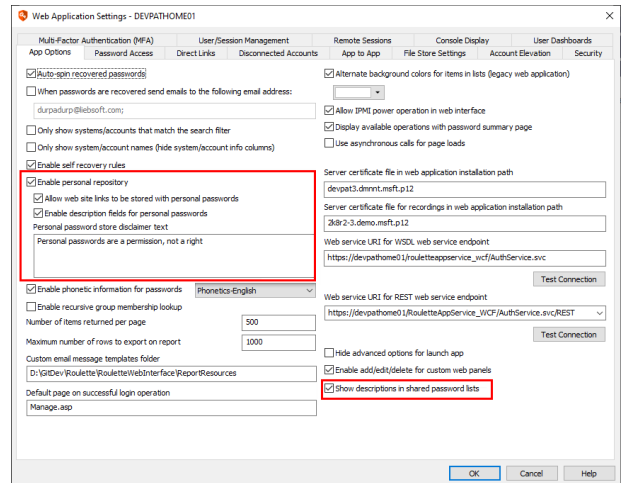
Passwords can be stored using the web application, and a link to a shared password may be sent to an unauthenticated user.

 **Note:** The recipient must be on the same intranet as the Privileged Identity web component.

## Share Personal Passwords

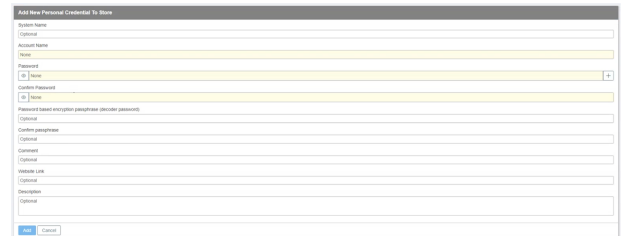
Enable Personal Repository in the management console:

1. From the **Actions** panel, click **Manage Web App**.
2. Double-click the instance name, or click **Edit** to modify the **Web Application Settings**.
3. On the **App Options** tab, enable the following options:
  - **Enable personal repository.**
  - **Allow web site links to be stored with personal passwords.**
  - **Enable description fields for personal passwords (optional).**
  - **Show descriptions in shared password lists (optional).** This option allows you to enter descriptions of items in shared password lists.
4. Click **OK** to save and update web application.



## Add a Password to the Personal Password Repository

1. Log into the web application.
2. Navigate to **Passwords > Personal**.
3. Click on the **+** button to the right of the **Filter** field at the top of the page to add a new password.
4. Enter the credential details. Required fields are:
  - **Account Name**
  - **Password**
  - **Confirm Password**
5. Click **Add**.

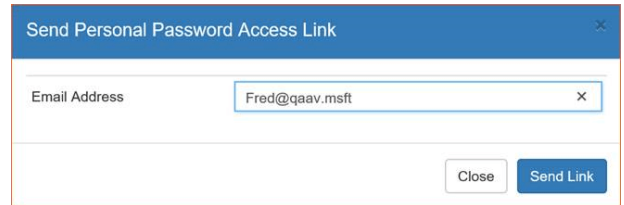


## Share a Personal Password

1. Log into the web application.
2. Navigate to **Passwords > Personal**.



3. Identify the password you wish to share, and then click the **Share** icon.
4. In the **Send Personal Password Access Link** dialog, enter the email recipient with whom you wish to share this password, and then click the **Send Link** button.



The dialog box has a blue header with the title "Send Personal Password Access Link" and a close button (X). Below the header is a text input field labeled "Email Address" containing the text "Fred@qaav.msft". At the bottom right of the dialog are two buttons: "Close" and "Send Link".

## Access a Shared Password

An email with a link to access the password is sent to the recipient. When the recipient clicks on the **Show Credential** link in the email, the personal password information that was shared with them will be displayed.



The dialog box has a blue header with the title "Shared Personal Password". It contains several fields with labels and values, and a "Close" button at the bottom right.

Label	Value
System Name	System1
Account Name	TestAdmin1
Comment	test account
Website Link	
Description	Optional
Password	••••

## Use Additional Features in the Web Application

This section describes additional functionality and settings for the web application outside of retrieving passwords, auditing, and application launching.

**i** For more information on those three topics, please see:

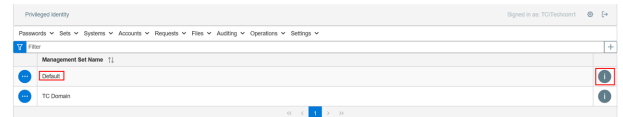
- ["Retrieve Passwords" on page 426](#)
- ["Audits and Alerts" on page 525](#)
- [Application Launcher and Session Recording](#)

## Configure Management Sets in the Web Application

A management set is a logical collection of systems and devices that you create as needed to organize system discovery and management tasks. There is no hard limit to the number of management sets you can create. A single system may appear in one or more management sets to address the necessary job and delegation requirements. Having a single system in one or more management sets will not consume multiple licenses when that one system is added by the exact same name.

Click **Sets > Management Sets** from the top menu in the web app to access the **Management Sets** page.

The first management set, called **Default**, is added automatically when Privileged Identity is installed. Initially it contains the system hosting the management console in it.



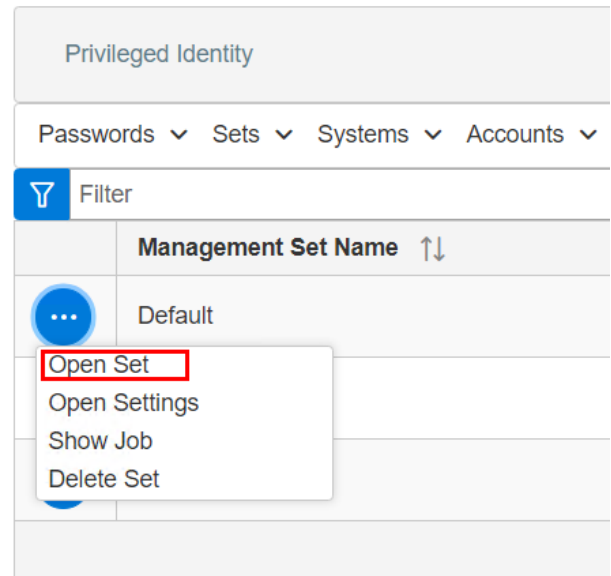
To view quick details for a management set, such as its name and comment, and the number of account stores in the set, click the **i** button for the management set.

### Add New Management Set

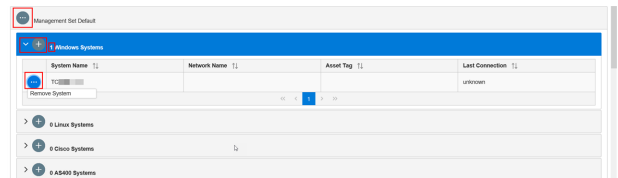
To add a new management set, click the **+** button in the top right corner of the **Management Sets** page. Enter a meaningful name for the management set, and then click **Save**. The management set is now listed. You must add systems to the set and configure its settings as described in the sections below.

## Add Systems to a Management Set

1. On the **Management Sets** page, click the **Actions** (ellipsis) button for the set, and then select **Open Set**.

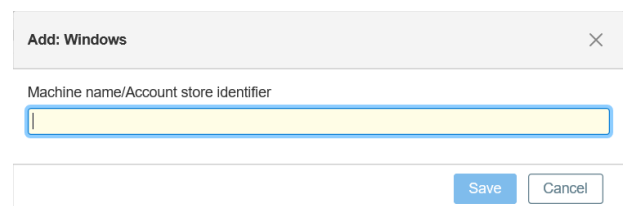


2. The types of systems that can be added to the set are listed. The number of each type of system already in the set is displayed next to the type of system. Click the **+** button next to the system type to add a system of that type.



**Tip:** Click the **arrow** to display details about the specific systems that are already in the set. Click the **ellipsis** for a specific system, and then select **Remove System** to remove it from the set. Click the **ellipsis** next to the management set name to open its settings.

3. Enter the name of the system or account store identifier, and then click **Save**.

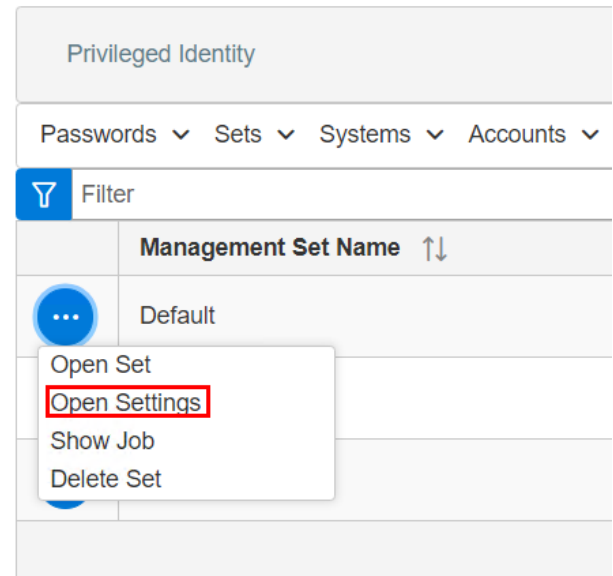


The screenshot shows a dialog box titled 'Add: Windows'. It contains a text input field labeled 'Machine name/Account store identifier'. Below the input field are two buttons: 'Save' and 'Cancel'.



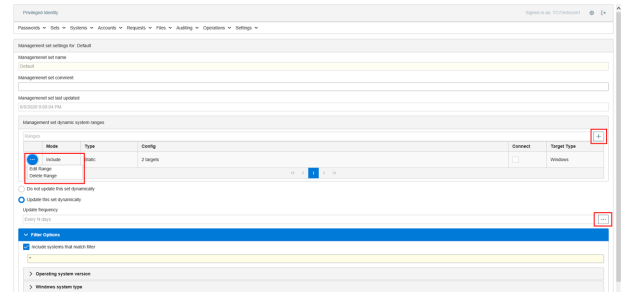
## Configure Settings for a Management Set

1. On the **Management Sets** page, click the **Actions** (ellipsis) button for the set, and then select **Open Settings**.



2. Configure the settings, and then click **Save**. Available settings are:

- **Management set name:** Shows the name of the management set. The name cannot be changed.
- **Management set comment:** This comment should help define what is in the management set, such as *London SharePoint Servers*.
- **Management set last updated:** Displays the date and time the set was last updated.
- **Management set dynamic system ranges:** Lists the inclusion or exclusion properties of the management set. Those properties are configured using the, **Edit Range** option from the **Actions** menu (ellipsis button). The columns in this section indicate a primary property of the management set attribute:
  - **Mode:** Defined as **Include** or **Exclude**, identifies if the property includes or excludes systems found with that query. Include statements are processed first from top to bottom. Exclude statements are processed from top to bottom. This option may be enabled or disabled, which allows you to enable or disable attributes of a management set without having to delete the attribute.
  - **Type:** Identifies the type of search being performed such as AD Query or Static.
  - **Config:** Lists properties such as the AD query or CMDB query being performed.
  - **Connect:** Identifies if the product attempts connections to target systems as part of the discovery process for inclusion in the discovery range.
  - **TargetType:** Identifies how a target found with this criteria is classified. Once a system is found and classified, the system classification does not change again without human intervention.
- **Update Frequency:** Defines the frequency with which this management set does(or does not) auto-update. The default time frame for a management set update job is to run every 30 days, indexed from the original time of management set's creation. Use the **Edit Job** (ellipsis) button on the dropdown to set additional job properties including:



- **Job Schedule:** Set the job to run as frequently as every hour.
- **Pre-Run Alerts:** Before the management set update runs, an email may be triggered to a defined email address or addresses (semicolon delimited list).
- **Pre/Post Run Steps:** Define additional scripts to run before or after the management set update runs.
- **Log:** The non-verbose log for the management set update job.
- **Filter Options:** Additional filter options for Windows systems being added to the management set based on name or OS type. We do not recommend you change these settings when querying from any source that already keeps track of this information, such as Active Directory. Configuring these options causes the product to create a secondary connection to the target to query its name and operating system values. Thus a query that could take 20 seconds from Active Directory can take several minutes or more for every management set update

## View and Edit Jobs for a Management Set

Every management set has a scheduled job associated with it. If the job is deleted, it is automatically recreated. These jobs may be viewed and edited by selecting **Show Job** from the **Actions** menu (ellipsis button) for the management set on the **Management Sets** page, or by clicking **Operations > Jobs** from the top menu.



*For more information on configuring jobs, please see:*

- *"Manage Jobs in the Web Application" on page 455*

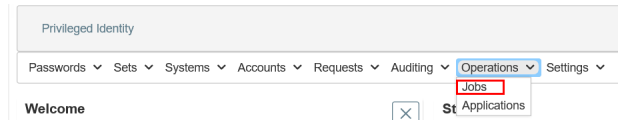
*For more detailed information on configuring management sets:*

- *"Enroll New Systems and Devices" on page 62*
- *"Management Set Introduction" on page 49*
- *"Create Management Sets and Enroll Systems" on page 63*
- *"Create Management Sets" on page 64*
- *"Manage Restricted and Orphaned Systems" on page 66*
- *"Configure Management Set Properties" on page 67*

# Manage Jobs in the Web Application

The web application provides an efficient method for creating and managing all of your change jobs. This section provides an overview for viewing, modifying, running, stopping, and deleting existing jobs, as well as for creating new change jobs in the web application.

1. In the web application, select **Operations > Jobs** from the menu.



2. To view details of a job, click the **i** button for a job in the grid to view to display a **Job Details** summary.

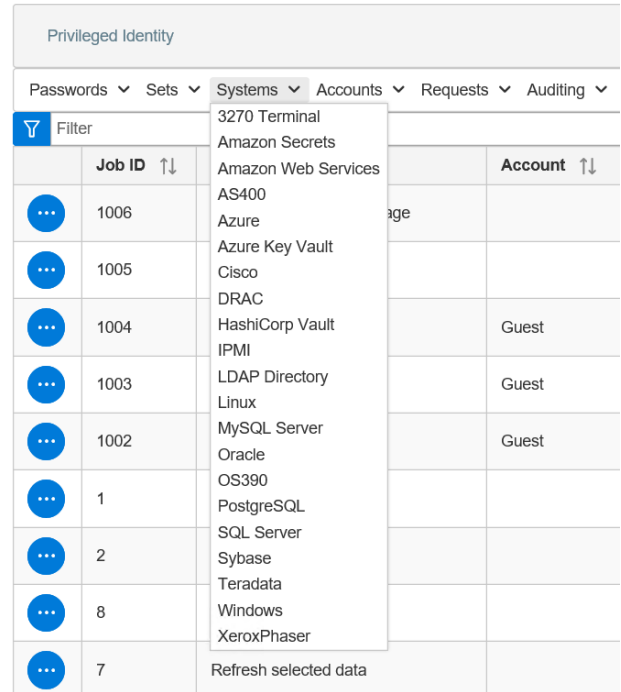
Attribute	Value
Job ID	1006
Job comment	
Job operation	Refresh and discover usage
Job result	Complete
Last run time	6/30/2020 3:40:12 PM
Next run time	6/30/2020 3:40:22 PM
Created by	
Creation time	6/30/2020 3:40:12 PM
Management set	TC Domain
Do not delete flag	false
Dynamic system list	false
Pull system list before run	false
Run job when new systems are added to set	false

3. To run, stop, disable, or delete a job, click the **ellipsis** button for a job in the grid, and then select **Run Job**, **Stop Job**, **Disable Job**, or **Delete Job**, as applicable. To edit a job, select **Job Details**, and then edit the options on each tab as desired.

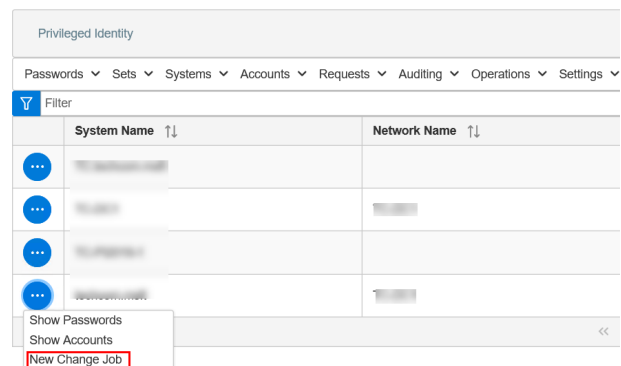
Job ID	Job Operation	Account	Next Run Time
1006	Refresh and discover usage		6/30/2020 3:40:22 PM
	Management set update		7/30/2020 2:00:00 AM
	Password change	Guest	6/28/2020 11:23:06 PM
1003	Password change	Guest	6/28/2020 3:34:07 PM

## Create a New Change Job

1. In the web application, select the **Systems** or **Accounts** dropdown, and then select the type of system or account from the list.



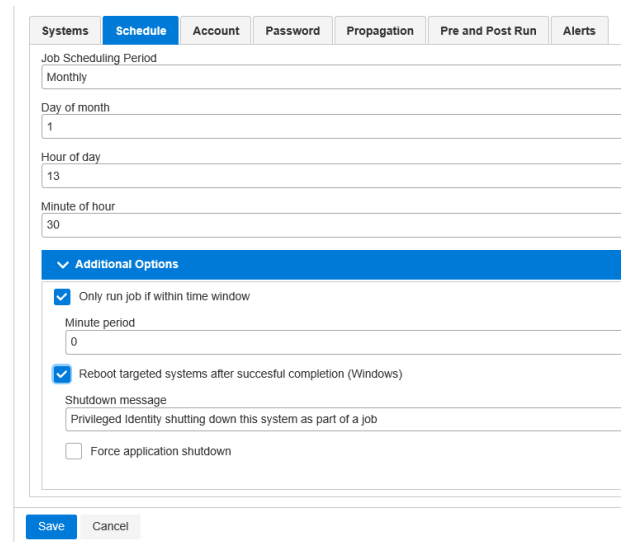
2. Click the **ellipsis** button for a system or account for which you want to create the new job, and then select **New Change Job**.



3. Select the **Schedule** tab, and then set the schedule to run the job using a 24-hour clock. Set additional options if, applicable. Jobs set to run immediately or interactively will run in the context of the logged-in user. All other scheduled jobs will be run by the deferred processor.

- **Job Scheduling Period:**

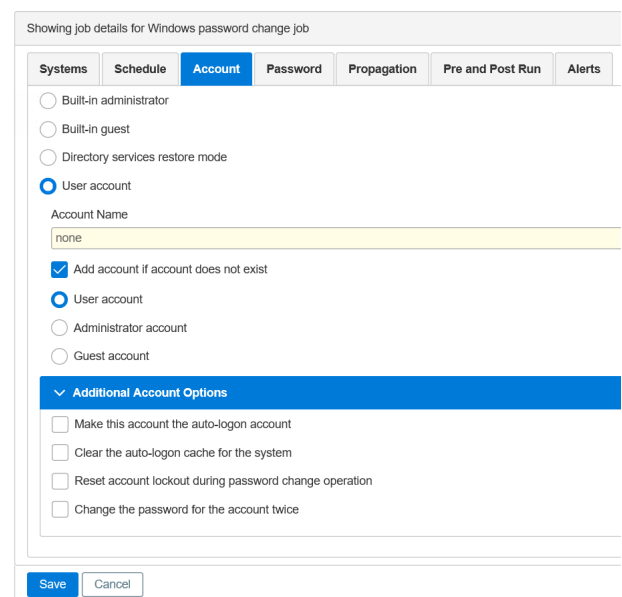
- **Immediately:** The job will run directly after you save the job using credentials of the currently logged-in user.
- **Hourly:** Set the minutes into the hour to run the job.
- **Daily:** Set the hour and minutes to run the job. This job will attempt to run every day at that time.
- **Weekly:** Set the day of the week, hour, and minutes to run the job. This job will attempt to run once per week on that day at that time.
- **Monthly:** Set the day of the month, hour, and minutes for the job to run. If a value greater than 28 is used, the job will be scheduled to run on the last day of the target month.
- **Yearly:** Set the month, day of the month, hour, and minutes to run the job. If a value greater than 28 is used, the job will be scheduled to run on the last day of the target month.
- **Once:** Set the month, day, hour, and minutes to run the job. This job will never run again except if you manually run it.
- **Every N days:** Set the period of days (for example, a 60 day interval), hour and minutes for the job to run. The timer will start from the this value is set.
- **Interactive Only:** The job will not be scheduled to run. You must manually run it. The job will run using the credentials of the currently logged-in user.
- **Every N hours:** Set the period of hours (for example, an 8 hour interval), and minutes into that hour for the job to run.




**Note:** To ensure your job will not run outside of the change window, enable the option to **Only run job if within time Window** and define the number of minutes where the window is available. This means a job set to run at 12:30 may be backed up depending on other jobs, but if it hasn't run within 90 minutes (2:00AM), the job should not run at all and will simply be rescheduled.

4. Select the **Account** tab, and then provide the target account settings. The options available vary depending on the type of system or account selected.

- **Built-in administrator account:** This option finds the built-in administrator account on local systems, member servers, and domains. This account's RID is 500. It does not matter if the account has been renamed.
  - If changing the built-in administrative account, the account can be renamed during the update process. To rename the account, select the **Rename account** option and supply the new name for the account. This feature is handy if the name of the administrator account has been changed on any of the selected systems but the name should be consistent across all systems. This feature is provided as an alternative to what is found in Active Directory Group Policy.
  - If the built-in administrator has been previously managed, and is now being renamed, the old name

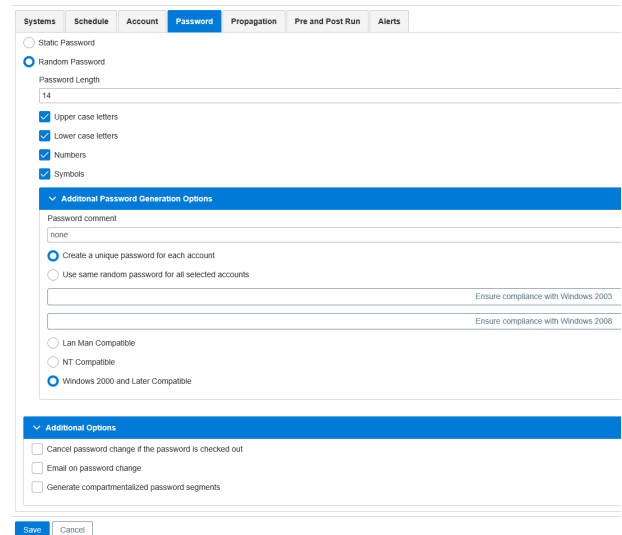


and old passwords will still be present in the password store. To remove the old password information associated with the old account name, select the option to **Remove stored password** for the old account name. This option will be unavailable if the account is not also being renamed.

- **Built-in guest account:** This option finds the built-in guest account on local systems, member servers, and domains. This account's RID is 501. It does not matter if the account has been renamed.
- **Directory services restore mode:** The directory services restore mode (DSRM) is a special password that is created on every Active Directory domain controller at install time. If working natively, the update process would require the use of NTDSUTIL. This process cannot be scripted. Every domain controller's DSRM password is unique to that server and should be changed regularly like any other password. Select this option to change the DSRM password.
- **User account:** Specify any account by name to update.
  - If the account is not found on one or more of the selected systems, the account can be added to those target systems by checking the **Add account if account does not exist** option.

5. Select the **Password** tab, and then provide either a static password or set the parameters to generate a random password:

- **Static password:** Define a static password. This is set once, when the job is run. The admin who inputs the job information will know what the password is. The password remains this value until it is reset with a new job. It never rolls to a new value following a password retrieval. Anyone who retrieves this password will then know the password until it is reset with a new job.
- **Random password:** Define password generation settings such as password length, allowed characters, and additional settings as follows:
  - Set the **Password Length**. This is the minimum and maximum length for the password.
  - Select each of the options for the types of characters the password must contain: **Upper case letters**, **Lower case letters**, **Numbers**, and **Symbols**.
  - Add a **Password Comment** to the password change job. The password comment will be visible in the website.
  - Select the **Create a unique password for each account** option to enforce the following:
    - If more than one system is included in the job, the named account on each system receives a unique password based on the password generation settings.
    - When the password for the account is retrieved using the web interface, the password is triggered for re-randomization in just two hours upon retrieval.
  - Select each of the compatibility options as required:
    - **Lan Man Compatible:** 14 character max, no lower case letters
    - **NT Compatible:** 14 character max, all character types allowed
    - **Windows 2000 and Later Compatible:** 127 character max, all character types allowed
  - Select the **Use the same random password for all selected accounts** option to enforce the following:
    - If more than one system is included in the job, the named account on each system will receive the exact same random password.

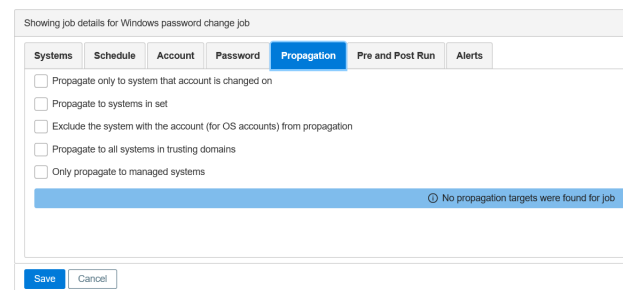


- When the password for the account is retrieved using the web interface, the password will NOT be triggered for re-randomization. This setting is ideal for service accounts where you wish to re-randomize accounts like root or local administrator following use, but do not wish to incur a service disruption following password retrieval of a service account.

- **Additional Options:** Select the following options, if desired.
  - **Cancel password change if the password is checked out:** For all system types, if the password is currently checked out and the password re-randomization interval occurs or the normal job schedule interval occurs, the job will not run and instead logs a failure and retries the job according to the global job retry policy.
  - **Email on password change:** Provide an email address to notify when a password has changed.
  - **Generate compartmentalized password segments:** Also referred to as *4 Eyes*. This requires the password to be divided into a number of segments (up to the number of characters in the password). This in turn requires this same number of people to obtain the entire password.

6. Select the **Propagation** tab, and then select the propagation options for the job, as desired:

- **Propagate only to system that account is changed on:** Used for a local password change where local resources use the local account; for example, scripts, services, etc.
- **Propagate to system in set:** (recommended) Used for distributed password changes where the target account exists in one location, such as Active Directory, and is used by services, processes, tasks, etc., on other systems. Use this setting to limit the propagation to a discrete list (management set) of systems.
- **Exclude the system with the account (for OS accounts) from propagation:** Enable this option to exclude examining the target password change system from the propagation scope if it would otherwise be included.
  - If this option is enabled, the **Exclude all domain controllers for Windows domain accounts** option is displayed. Enable this to exclude domain controllers from password propagation operations if they would otherwise be included in the scope.
- **Propagate to all system in trusting domains:** Enable this option to instruct Privileged Identity to enumerate all trusting domains and attempt discovery and propagation on each system. This not only requires full access to every system in the forest and other trusting domains, but will incur a huge time penalty, depending on the size of your enterprise.
- **Only propagate to managed systems:** Enable this option to set the propagation scope to every trusting system, as long as it appears in Privileged Identity.



7. Select the **Pre and Post Run** tab, and then select your desired options to run an application before or after a job runs.

Showing job details for Windows password change job

Systems Schedule Account Password Propagation **Pre and Post Run** Alerts

Run application before starting operation

Application path

Application arguments

Wait for application to exit

Abort operation if application returns non zero exit code

Run application after finishing operation

Application path


Application arguments

Wait for application to exit

Save Cancel


8. Select the **Alerts** tab, and then check the option to send an email alert before the job is scheduled to run, if desired. Configure the settings as follows:


- **Hours before job is scheduled to run:** The number of hours before the job runs that the email will be sent.

 **Note:** This requires the deferred processing service to be running and email settings to be configured in order for the alert to be sent.

- **Email address or mailing list to send alert:** Enter either a single email address or a semicolon-delimited list of email addresses to receive the alert.
- **Message:** Enter a custom message to send. Use HTML encoding for any special formatting, such as **<br>** for a line break. There are multiple variables that can be replaced in the custom message:

- **\$DateTimeLocal\$:** The current local time according to the zone processor when the alert was sent
- **\$DateTimeUTC\$:** The current local UTC time according to the zone processor when the alert was sent
- **\$AccountName\$:** The name of the target account if performing a password change job
- **\$RunTimeLocal\$:** The local time (non-UTC) that the job will be ran
- **\$JobID\$:** The ID of the job to be run
- **\$SystemsList\$:** The list of every system that is part of the job

 **Note:** Once the alert is performed, if you re-schedule the job no secondary alert will be sent. The alert flag is reset only after the job runs.

 For more detailed information on configuring job options, please see the following:

Systems Schedule Account Password Propagation Pre and Post Run **Alerts**

Send email alerts before job is schedule to run

Hours before job is scheduled to run

Email address to mailing list to send alert

Automated message from Privileged Identity at \$DateTimeLocal\$ (\$DateTimeUTC\$)<br><br>Account \$AccountName\$<br><br>Scheduled to be updated at \$RunTimeLocal\$ (\$RunTimeUTC\$)

Save Cancel



**i**

- *"Set the Job Schedule" on page 242*
- *"Configure Password Settings" on page 227*
- *"Work with Compartmentalized Passwords (Four Eyes)" on page 446*
- *"Use Pre and Post Run Steps to Run Scripts and Applications" on page 231*
- *"Manage Passwords and SSH Keys" on page 244*

# Elevate Accounts

Account elevation is a feature available for Windows, Linux, Unix, and Apple Mac systems. The logged in user account can be placed into a target group for a period of time, and then removed automatically when the time expires.

There are two types of elevation available to Privileged Identity:

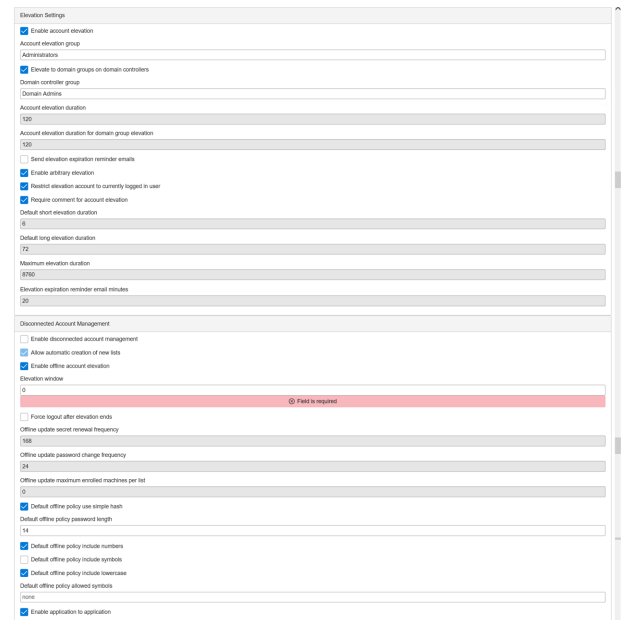
- Self service elevation allows a user to click a single link next to a single system (in the systems view).
- Arbitrary elevation allows a delegated user to place another user into any group on any system for an arbitrary period of time between now and some point in the future.

You can use account elevation from the web application or programmatically.

## Configure Account Elevation in the Web Application

1. Select **Settings > Site Settings** from the top menu.
2. Scroll down to the **Elevation Settings** section.
3. Check **Enable account elevation**, and then set the options for the elevation as required.

**i** For more information on elevation settings, please see "[Configure Site Settings](#)" on page 515.



## Configure Programmatic Account Elevation

Programmatic elevation is available only to users granted the global delegation **Elevate Any Account**.

- From PowerShell, call **New-LSJobAccountElevation**
- From SOAP, call **JobOps\_CreateAccountElevationJob**
- From REST, call **/REST/Job/WindowsElevation**

**i** For more information about account elevation in Linux systems, see "[Elevate Accounts for Linux Systems](#)" on page 468.

## Self-Service Elevation - Simple

The simple version of self-service elevation allows a delegated user the ability to elevate their own account into a pre-defined group on systems in a particular management set for a pre-defined period of time.

The target group and time period are defined in the web application settings. The management set or system is defined with global delegations, per management set delegations, or per system delegations.

To self-service elevate, the identity must have either of the following permissions:

- All Access
- Elevate Account Access and View Systems

### Elevate Account

1. Select **Systems > Windows**.

2. From **Account Type**, select either **Windows** or **Linux**.
3. From **System List**, select an available system or multiple systems.
4. Enter the **Account Name** to use for elevation.
5. Enter a **Group Name** to elevate into.
6. The **Elevation Duration (minutes)** field is prepopulated with the maximum time this elevation can last. You may leave this at its maximum or enter a smaller number.
7. Selection an **Elevation Time** or leave at the default of **Elevate Now**.
8. Enter a **Comment** to explain your reason for elevating this account.
9. If email expiration notices are enabled and SMTP settings are correctly set up, you can enter an **Expiration Email Notification** to be alerted when the elevation ends.
10. Click **Elevate**.
11. A message indicating an elevation job has been created displays.

**Privileged Identity**

Passwords ▾ Sets ▾ Systems ▾ Accounts ▾

**Elevate Account**

**Account Type**

**System List**  
  

☰
📄

**Account Name**

**Group Name**

**Elevation Duration (minutes)**

**Elevation Time**

**Comment**

Elevate
Cancel
★

## Self-Service Elevation - Advanced

In the advanced version of self-service elevation, a delegated user can select from a list of predefined groups associated with different management sets and can elevate their own account for a loosely predefined period of time.

Delegations are predefined in the management console and identify a target identity, target management set, target group, and maximum elevation duration.

To elevate, the identity must have the following permissions:

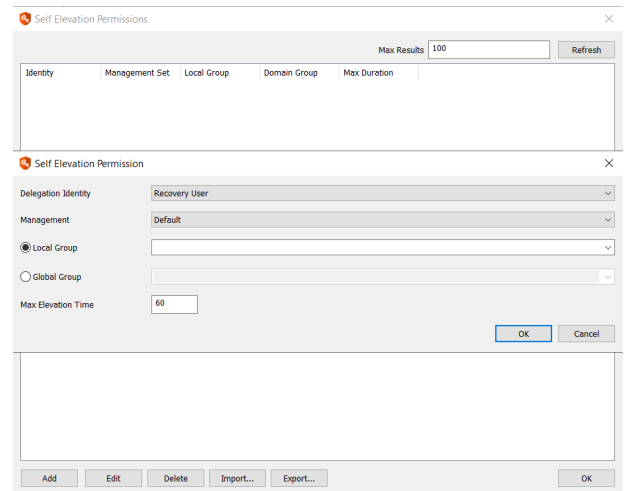
- **All Access** or **View Systems**
- A defined management set from which to view systems (if not granted **All Access**)
- Predefined self-elevation permissions

## Self-Elevation Permissions

You can manage self-elevation permissions in the management console or web application, either one at a time or by text import, or you can manage them programmatically.

### Manage Self-Elevation Permissions in the Management Console

1. Click **Delegation > Web Application Self-Elevation Permissions** from the top menu.
2. Click **Add** at the lower left of the dialog.
3. In the **Self Elevation Permission** dialog, enter:
  - **Delegation Identity:** Select an enrolled identity to which to grant this permission.
  - **Management Set:** Select a management set. The above identity will be able to elevate to Windows or Linux systems in this management set.
  - **Local Group/Global Group:** Set the scope of the target group and either type the simple name of the group or select a predefined group from the dropdown.
  - **Max Elevation Time:** Define the maximum time in minutes for which the user may elevate.
4. Click **OK**.




To manage Self-elevation permissions in the web app, please see "[Configure Delegation](#)" on page 490.

### Manage Self-Elevation Permissions with Text Import

1. Click **Delegation > Web Application Self-Elevation Permissions**.
2. Click **Import** at the bottom of the dialog.

3. You are prompted to confirm that you understand the required format. The expected format is:

```
identity,management_set,local_group,domain_group,time
```

For example:

```
admin,Web Servers,SQL Admins,,60
```

Click **OK**.

4. Browse to the appropriate text file, then click **Open**.
5. You are asked to map columns. Make any changes needed, then click **OK**.

## Perform Self-Elevation

Once you've defined self-elevation permissions, permitted users may elevate themselves to the predefined groups present on the systems in the target management sets. You can perform self-elevation from the web app or programmatically.

### Self-Elevate from the Web Application

When self-elevation is enabled for the user, the web app offers an **Operations > Self Elevation** menu item.



**Note:** If this option is not visible, either account elevation is not enabled, the user has no rights, or there are no self-elevation permissions defined.

1. In the web app, navigate to **Operations > Self Elevation** from the top menu.
2. From **Account Type**, select the platform target for elevation.
3. From **System List**, select an available system.
4. Enter the **Account Name** to use for elevation.
5. Enter a **Group Name** to elevate into.
6. The **Elevation Duration (minutes)** field is prepopulated with the maximum time this elevation can last. You may leave this at its maximum or enter a smaller number.
7. Selection an **Elevation Time** or leave at the default of **Elevate Now**.
8. Enter a **Comment** to explain your reason for elevating this account.
9. If email expiration notices are enabled and SMTP settings are correctly set up, you can enter an **Expiration Email Notification** to be alerted when the elevation ends.
10. Click **Elevate**.
11. A message indicating an elevation job has been created displays.

### Self-Elevate Programmatically

- **REST:** Job/WindowsElevation or Job/LinuxElevation

## Arbitrary Account Elevation

Arbitrary elevation allows a delegated user the ability to elevate any account into any group on any system for an arbitrary period of time.

To use arbitrary elevation, the identity must have either of the following permissions:

- All Access
- Elevate Any Account

### Elevate an Account

1. In the web app, select **Operations > Elevation** from the top menu.
2. From **Account Type**, select either **Windows** or **Linux**.
3. From **System List**, select an available system or multiple systems.
4. Enter the **Account Name** to use for elevation.
5. Enter a **Group Name** to elevate into.
6. The **Elevation Duration (minutes)** field is prepopulated with the maximum time this elevation can last. You may leave this at its maximum or enter a smaller number.
7. Selection an **Elevation Time** or leave at the default of **Elevate Now**.
8. Enter a **Comment** to explain your reason for elevating this account.
9. If email expiration notices are enabled and SMTP settings are correctly set up, you can enter an **Expiration Email Notification** to be alerted when the elevation ends.
10. Click **Elevate**.
11. A message indicating an elevation job has been created displays.

**Privileged Identity**

Passwords ▾ Sets ▾ Systems ▾ Accounts ▾

**Elevate Account**

**Account Type**

**System List**  


☰
📄

**Account Name**

**Group Name**

**Elevation Duration (minutes)**

**Elevation Time**

**Comment**

Elevate
Cancel
★

## Elevate Accounts for Linux Systems

In Privileged Identity versions 7.x and later, elevation happens against a group on the target Linux system using the commands below, edited in the **response.xml** file. Any customization to the group elevation on the target Linux system must be updated in the **response.xml** file for elevation to work.



**Note:** Linux Elevation in Privileged Identity versions 5.5.0 - 6.x requires elevation extensions. For more information, please see "[Linux Elevation in Privileged Identity Versions 5.5.0 - 6.x](#)" on page 469.

### Linux/Unix Configuration

First, you must configure the groups and memberships on target Linux systems. These are basic settings to test if elevation works against target groups and the commands are executed on the Linux system in question.

Description	Command
Add a new group	<code>sudo groupadd NAME-OF-THE-NEW-GROUP</code>
List users in a group	<code>groups NAME-OF-Group</code> or <code>groups NAME-OF-Account</code>
Add users to a group	<code>sudo usermod -G NAME-OF-Group NAME_OF_Account</code>
Remove a user from a group	<code>sudo gpasswd -d NAME_OF_ACCOUNT NAME_OF_GROUP</code>

### Privileged Identity Configuration

The default **response.xml** has the following commands:

- For elevation:

```
<StdIn>sudo usermod -G $(TargetGroup) $(TargetAccount)</StdIn>
```

```
<StdOut>#\| \$| \|></StdOut>
```

- For de-elevation:

```
<StdIn>sudo gpasswd -d $(TargetAccount) $(TargetGroup)</StdIn>
```

```
<StdOut>#\| \$| \|></StdOut>
```



**Note:** As with all password updates and elevation, the account used to run the job needs the required permissions to do so.





**Note:** The Web UI feature for Linux elevation is located under **Operations > Elevation** and requires permission to use this feature.

## Linux Elevation in Privileged Identity Versions 5.5.0 - 6.x

Account elevation for Linux systems uses a Privileged Identity Extension component to call upon the **ExtUnixSudoElevationTarget** to elevate users on the target Linux machine. When successful, it assigns Sudo privileges to the user. After de-elevation, the user does not have Sudo privileges.

The following configurations are required, and described in more details below.

- Linux Sudo setup and configuration
- SQL Server security login account
- Elevation extension file path and folder placement
- Privileged Identity elevation extension configuration
- Elevating with the web application
- PWC Extensions (Account Store, CLR), installed and configured during installation of the Privileged Identity Admin Console.

### Linux Sudo Setup and Configuration

The Linux/Unix environment must be configured to use Sudo. These instructions assume Sudo is configured on your Linux/Unix environment and focus on the user information, permissions, and files. We use the wheelhouse method for the user sudoers file.

In addition to Sudo, you also must have SSH running. Some distributions, such as Ubuntu, might not have SSH installed initially and it must be added.

### Install SSH on an Ubuntu System

If SSH is not already installed, configured, and running, enter the following commands from the command prompt to add it:

```
sudo apt install ssh
sudo apt install sshd
ps ax | grep sshd
sudo ssh start
sudo service ssh status
sudo initctl reload-configuration
sudo service ssh status
ssh start/running, process 1084
```



**Note:** These are generic commands for Ubuntu systems. Any other required commands must be researched and executed. These commands must be entered by a user that is in the sudoers file or has a root level account.

## Add a User to the SUDO Wheelhouse

These steps assume a user has been created but does not have a user sudoers file. If a user needs to be created, you can enter this command, entering the actual username in place of testuser:

```
sudo useradd testuser1
```

Update the visudo file, and create a user sudoers file:

1. Ensure the last line in the visudo file contains the following:  
**#includedir /etc/sudoers.d**
2. Enter the following commands to create the user sudoers file. Use the **bash** command to get to the root level. Enter the actual username in place of testuser1:
  - **sudo bash**
  - **# cd /etc/sudoers.d**
  - **/etc/sudoers.d# nano testuser1**
3. Add the following line: **!testuser1 ALL = (ALL) ALL**. The ! toggles on/off sudo for the user. Without the !, the user has sudo privileges with the above assignment.
4. Exit and save.
5. Enter the following: **/etc/sudoers.d# ls -l**. You should see the file you created.
6. Enter the following commands to assign the file the correct permissions:
  - **/etc/sudoers.d# chmod 440 testuser1**
  - **/etc/sudoers.d# visudo -c**
7. The results display the following:
  - **/etc/sudoers: parsed OK**
  - **/etc/sudoers.d/README: parsed OK**
  - **/etc/sudoers.d/testuser1: parsed OK**

## Add a SQL Security Login

Unlike the Windows elevation feature, the Unix/Linux Sudo elevation component is part of the Privileged Identity Extension framework. It uses the **PWC Extensions – Account Store (CLR)** COM wrapper that is created upon installation of Privileged Identity.

This COM wrapper uses the **Network Service** identity to run. Therefore, the machine account, where the elevation job is run from, must be added to the **Security Login** in SQL database and mapped to the database catalog. This is typically in the format of **Domain\MachineName\$**. This account must have the following rights:

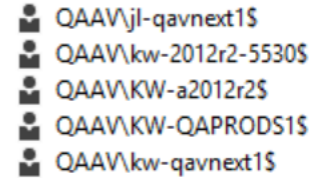
- DBO

or

- DB\_DDAdmin
- DB\_DataReader
- DB\_DataWriter

**Example:** `qaav\kw-qaprods1$` is the machine name of the zone processor host machine that is handling the elevation. This account is mapped to the database catalog with the **DBO** rights.

On the SQL server, go to **Security > Logins** and add a **New Login**. Add the machine name following the **Domain\MachineName\$** format. Then map this account to the application database for the Privileged Identity environment and assign the **DBO** rights.



**Note:** If using a zone processor, and it is not on the Privileged Identity host machine, you must create the **COM** object for **PWC Extension – Account Store (CLR)** on the zone processor host machine. You must also copy or install the supporting library files and register them.

## Extension File Path and Folder Placement

The Linux extension component looks for the assembly files under the following folder:

`~\Roulette\AccountStoreSupport\Extensions\UnixSudoElevationTarget\`

However, the extension folder is located under the `~\Roulette` program directory. Therefore, the entire extension folder must be copied to the `~\Roulette\AccountStoreSupport` folder. Follow these steps to copy the folder:

1. Shut down all running processes, specifically the `dhllhost.exe` for the **PWC Extension – Account Store (CLR)** if it is running.
2. Copy the entire extension folder (not just the contents) to `~\Roulette\AccountStoreSupport`.
3. Restart the Privileged Identity admin console.

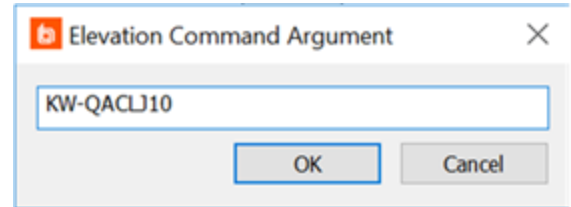
## Elevation Extensions Configuration — Admin Console

Although this feature uses the extension component for Privileged Identity, it is not configured under the **Account Store CLR** section of Privileged Identity. Instead, an elevation extension must be created and configured under the **Settings** section of Privileged Identity. Follow these steps to create and configure the elevation extension.

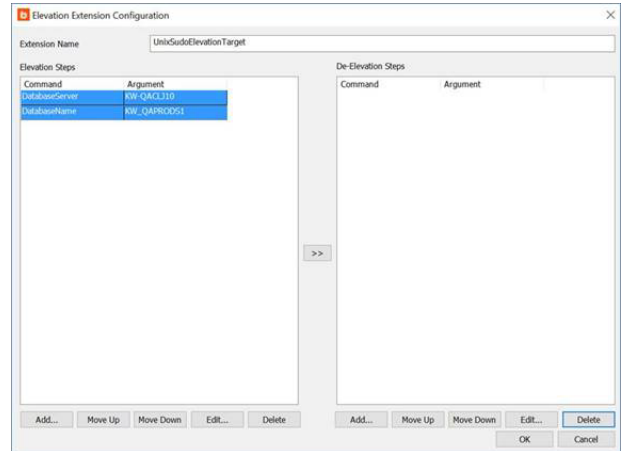
1. Open the Privileged Identity admin console.
2. Go to **Settings > Extension Components > Elevation Extensions**.
3. From the **Account Elevation Extension Modules** dialog, click **Add**.
4. Enter the following for **Extension Name: UnixSudoElevationTarget**. This must be entered exactly as spelled, and is case sensitive.
5. From the left side of the **Elevation Extension Configuration** dialog (**Elevation Steps** side), click **Add**.
6. For the **Elevation Command** pop-up, enter **DatabaseServer**, and then click **OK**.



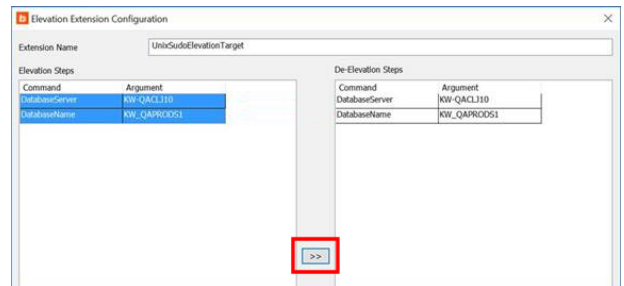
- In the Elevation Command Argument dialog, enter the database server name.



- From the **Elevation Steps** column, click **Add** again.
- Enter the following information for the above, respective dialogs: **DatabaseName** and **KW-Prods1**.
- Two entries display. Replace **KW-QACLJ10** with your SQL server, and **KW-Prods1** with your application database.



- Select both steps on the left and click the >> button in the middle to add the steps to the **De-Elevation Steps** column.
- Click **OK**.
- Click **OK** again to close the **Account Elevation Extension Modules**.



Any user with the **Grant All Access** or **Elevate Any Account** and the **View Systems** permissions can use this feature. **Grant All Access**, and **Elevate Any Account** are global settings and are in the **Delegation > Web Application Global Permissions**, but **View Systems** permissions can be set at any level down to **Web Application per Systems permissions**.



**Note:** If you have the **Grant All Access** permissions you do not need to set the other two permissions.

## Account Elevation — Web UI

- Log in to your Privileged Identity web application.
- Click **Operations > Elevation**.

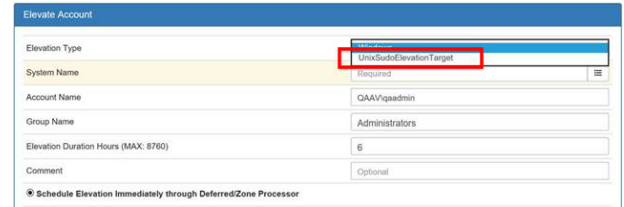
3. Enter the following information:

- **Elevation Type:** Select **UnixSudoElevationTarget**.
- **System Name:** Enter the Linux system under Privileged Identity management that you want to do the elevation against.
- **Account Name:** Enter the account name that was created with the sudoers file. This account does not need to be under Privileged Identity management.

4. Click **Elevate Account**.

5. A message states that the account has successfully been elevated.

6. You can verify that the account you elevated has SUDO permissions.



Elevate Account	
Elevation Type	UnixSudoElevationTarget
System Name	Required
Account Name	QAAVqadmin
Group Name	Administrators
Elevation Duration Hours (MAX: 8760)	6
Comment	Optional
<input checked="" type="checkbox"/> Schedule Elevation Immediately through Deferred/Zone Processor	

## Use the Secure File Store

The file store is an optional feature of Privileged Identity. It can be accessed via the web application or programmatically. The file store is used for secure storage of various types of data such as old password lists, certificates, or other important documents.

To use the file store, the option must be enabled by licensing, and the feature must be enabled in the web application settings.

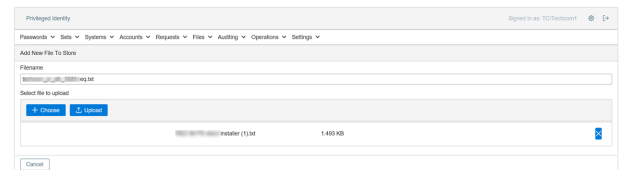
For a user to gain access to the file store, they must have either of the following permissions:

- All Access
- Access File Repository

Once access to the secure file store has been granted, to access a file, the owner must have granted permissions for the file to the target identity.

### View and Add Files

1. To view the file store, select **Files > View Files** from the top menu.
2. Files the user has access to are listed. If no files are present or no access is granted to at least one file, the page will display **No Stored Files**.
3. To add a file, click **+**.
4. Click **Choose**, and then select your file.
5. Click **Upload**.



### Grant Permissions to Other Identities

The identity who uploaded the file will be considered the owner of the file and effectively have full control of the file. To allow other identities access to the file, permissions need to be granted.

1. Click the **File Permissions** button for the particular file.
2. Select an identity from the **Identity Name** dropdown.
3. Grant the desired permissions:
  - **View**: View the file in the file store.
  - **Request**: Request access to the file. Another user must be granted the grant permission to grant or deny the request.
  - **Grant**: Grant or deny requests to download the file.
  - **Download**: Download the file. How the file opens will depend on the browser and IIS configuration for the particular file type. For example, Internet Explorer will simply open a text file in the browser while others may require you to download the file first. This behavior is not controlled by Privileged Identity.
  - **Update**: Allows replacing the file with a new file. The original file will be kept as a Stored Version, which can also be downloaded separately.
  - **Delete**: Allows deleting of the file and its stored versions.
  - **Delegate**: Allows configuring these permissions on this file.

## Download a File

Opening a file (or its stored versions) is simply a matter of having the Download permission on the file and clicking the **Download File** button. How the file opens will depend on the browser and IIS configuration for the particular file type. For example, Internet Explorer will simply open a text file in the browser while others may require you to download the file first. This behavior is not controlled by Privileged Identity.

## Access the File Store Programmatically

See the programmers reference for more information.

- From PowerShell, call...
  - **Add file to store** - Save-LSFileInStore.
  - **Delete a file from the store** - Remove-LSFileFromStore.
  - **Download file from store** - Get-LSFileFromStore.
  - **Request access to a file** - Set-LSFileRequest.
  - **Deny a file access request** - Deny-LSFileRequest.
  - **Grant a file access request** - Grant-LSFileRequest.
  - **Check in a file** - Set-LSCheckInFile.
  - **Get the list of your checked out files** - Get-LSListCheckedOutFiles.
  - **Get the list of ALL checked out files** - Get-LSListAllCheckedOutFiles.
  - **Get a list of all stored file versions** - Get-LSListFileVersions.
  - **Get a list of the current stored file versions** - Get-LSListMostRecentVersionOfFiles.
  - **Get a list of permissions on the file** - Get-LSListDelegationPermissionsOnFile.
  - **Output the permission list for the file to a text file** - Get-LSOutputAllPermissionsToFile.
  - **Add delegations** - Set-LSDelegationPermissionForIdentityOnFile.
  - **Remove Delegations** - Remove-LSDelegationPermissionForIdentityOnFile.
  - **Admin forcibly check a file in** - Set-LSForceFileCheckIn.
  - **Add a digital signature to the file** - Edit-AddSignatureToFile.
- From SOAP, call...
  - **Add file to store** - FileStoreOps\_StoredFile\_Save.
  - **Delete a file from the store** - FileStoreOps\_StoredFile\_Delete.
  - **Download file from store** - FileStoreOps\_StoredFile\_Checkout.
  - **Request access to a file** - FileStoreOps\_StoredFile\_RequestFileAccess.
  - **Deny a file access request** - FileStoreOps\_StoredFile\_DenyRequest.
  - **Grant a file access request** - FileStoreOps\_StoredFile\_GrantRequest.
  - **Check in a file** - FileStoreOps\_StoredFile\_CheckIn.
  - **Get the list of your checked out files** - FileStoreOps\_GetCheckedOutFileList.
  - **Get the list of ALL checked out files** - FileStoreOps\_GetAllCheckedOutFiles.
  - **Get a list of all stored file versions** - FileStoreOps\_GetListOfMostRecentVersionsOfFiles.
  - **Get a list of the current stored file versions** - FileStoreOps\_StoredFile\_GetVersions.

- **Get a list of permissions on the file** - DelegationOps\_StoredFile\_GetPermissions.
  - **Add/Remove delegations** - DelegationOps\_StoredFile\_SetPermissions.
  - **Admin forcibly check a file in** - FileStoreOps\_StoredFile\_ForceCheckIn.
  - **Add a digital signature to the file** - there is no web service equivalent for this function.
- From REST, call...
    - **Add/Update/Delete/CheckIn/CheckOut a File** - /REST/File.
    - **Request a file** - /REST/File/Request.
    - **Deny a file access request** - /REST/File/Request/Deny.
    - **Grant a file access request** - /REST/File/Request/Grant.
    - **Get a list of all stored file versions** - /REST/Files.
    - **Get a list of all file versions** - /REST/File/Versions.
    - **List all files checked out** - /REST/File/CheckedOut.
    - **List all files checked out to you** - /REST/File/CheckedOut/All.
    - **Add/Remove Delegations** - /REST/Delegation/File.
    - **Admin forcibly check a file in** - /REST/File/Force.
    - **Add a digital signature to the file** - there is no web service equivalent for this function.



## Manage File Store Settings

Operation and abilities of the file store are managed by the following settings. The file store is an optional feature of Privileged Identity, and these settings are available only if the license for this feature is installed.

- **Enable file store:** Enabling this option allows the upload, secure storage, delegated access, and access auditing of files within the web application.
- **When files are accessed send emails to the following address:** Any time a file is opened or checked out, this email address receives a notification.
- **Enable file check-out:** If this option is left disabled, any number of users may open the same file at the same time. With this option enabled, a file is checked out to a single user at any time.
- **Check-out window/extension interval:** Time in minutes that a user is guaranteed solitary access to a given file, blocking any other user from checking the file out and making changes to it.
- **Maximum check-out duration:** The maximum time in minutes that a user may have any single file checked out.
- **Maximum simultaneous check-outs:** The maximum number of files a single user may have checked out to them at any time.
- **Log all file check-outs / check-ins to system's event log:** Define a Windows event log server for file vaulting operations by providing the NetBIOS name of a target Windows computer. Events are written to the Application Log and have a source of Enterprise Random Password Manager. There are also event sinks for file store operations which provide more functionality and logging data.
- **Enable encryption for files in the store:** Turn on encryption for files stored in the file store. By default, this is not enabled due to encryption export restrictions that are specific to each country as applied to the encryption of data. This product encrypts files using the same methods used to encrypt the passwords it is storing. Please review country-specific laws on encrypting data before enabling this feature.



**Note:** When changing system-wide encryption keys, disable this encryption, change the encryption key for the system, update the web application, and then re-enable file store encryption.

- **Default file upload permissions:** These values define what permissions are assigned to a file that is uploaded into the secure file store. If this option is not enabled, when a user who belongs to multiple identities that are also granted access to the solution uploads a file, full control permissions will be granted to the user and all other identities the user belongs to. This can have the unintended side effect of unnecessarily granting access to secondary identities.
- **Limit file sizes for uploaded files in the store:** This is the maximum allowable size for file uploads. This size may still be limited by IIS settings which by default are more restrictive. If IIS is set to a lower value, the IIS value will take precedence.

## Web Application Settings

The web application has a number of settings that apply to individual users (and their sessions) as well as global settings that apply to all users.

## Session Settings

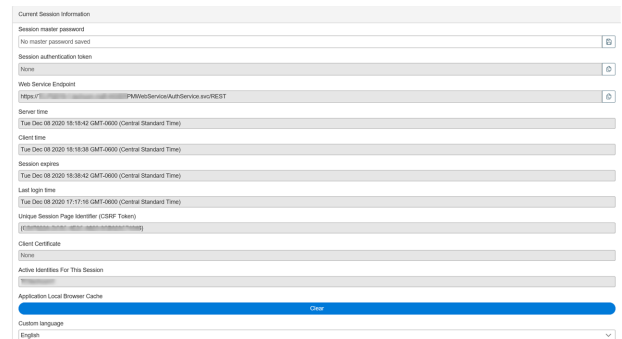
Every user has session settings. Session settings can be controlled by the user to affect their own current and future sessions.

The session settings are divided into seven sections:

## Current Session Information

When a user logs in, the login event is audited. The user can also see information about their logon session. This page is helpful for troubleshooting why a user has more or less permissions that they should and clearing about browser cache.

- **Session master password:**
- **Session Authentication Token:** The login token of the user. This token can be passed to the legacy SDK, PowerShell, or SOAP/REST interfaces for the authentication token. This value will be re-generated with each new login.
- **Web Service Endpoint:**
- **Server Time:** Local time on the web server the user has just attached to.
- **Client Time:** Current time for the system where the user's browser is.
- **Session Expires:** This is when the user's session expires if there is no further transactions.
- **Last Login Time:** The last time the user logged in.
- **Unique Session Page Identifier:** The page session identifier for this user's session.
- **Client Certificate:** If the user was logged in with a certificate, the certificate will be identified here.
- **Active Identities for this Session:** Identifies all identities (users, groups, roles, etc.) associated with the logged in user for this sessions.
- **Application Local Browser Cache:** Browsers can cache quite a bit of information and depending on how the browser reloads that information when revisiting or refreshing a page, a user may not see the information they expect to see. If any such issues arise, click this button to clear the browser local cache.
- **Custom Language:** The web application will default to display the language the user's browser is localized in if a pre-defined language exists. Otherwise, the user can force a language always be used for their session. Click the custom language link to set a language for the user. Choose from:
  - Arabic
  - Chinese Traditional
  - Chinese Simplified
  - Danish
  - Dutch
  - English
  - Finnish
  - French
  - German
  - Greek
  - Hebrew



The screenshot shows a 'Current Session Information' page with the following fields and values:

- Session master password: No master password saved
- Session authentication token: [Token]
- Web Service Endpoint: [Endpoint]
- Server time: Tue Dec 08 2020 16:18:42 GMT-0500 (Central Standard Time)
- Client time: Tue Dec 08 2020 16:18:38 GMT-0500 (Central Standard Time)
- Session expires: Tue Dec 08 2020 16:38:42 GMT-0500 (Central Standard Time)
- Last login time: Tue Dec 08 2020 17:17:16 GMT-0500 (Central Standard Time)
- Unique Session Page Identifier (CSRF Token): [Token]
- Client Certificate: [Certificate]
- Active Identities For This Session: [List of identities]
- Application Local Browser Cache: [Clear button]
- Custom language: English

- Hindi
- Hungarian
- Italian
- Japanese
- Korean
- Norwegian
- Polish
- Portuguese
- Russian
- Spanish
- Swedish
- Tagalog
- Turkish

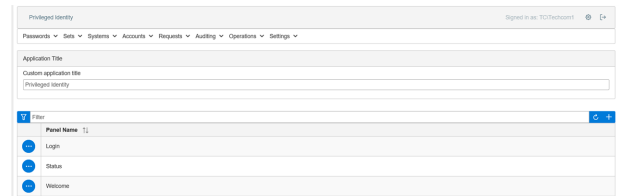
## Main Panel Configuration

Main panel configuration defines which charts and panels are available to users on their home page dashboard upon logging into the web application.

Every user can see web content panels. Web content panels are admin defined panels that can be used to display information, send messages, show graphics, etc. Panels can be created and edited from the web application by selecting **Settings > Web Content** from the menu. You must be an **All Access** user to customize web content.

Charts are available to users who have the **View Dashboards** global delegation.

There are six default panels displayed on the homepage dashboard:



- **Favorites:** Shows any items the user has added as a favorite item. These are essentially shortcuts to passwords or sessions the user refers to frequently. With a favorite, the user can simply click the favorite link and be brought right to the item in question without having to fully navigate through other menus first.

Click the **ellipsis** button to launch the favorite, rename it, or to delete it. Click the **X** button to remove the **Favorites** panel from the web app homepage.

- **Passwords:** The **Passwords** panel allows a user to search for managed passwords using a free text search. This panel can save the user time by allowing a partial match search for system name, account name, or name space to locate a managed password rather than having to navigate and filter.

Type in the [partial] name of a system, namespace, or account name and click the **filter** button on the right. A list of matching items will appear in the list. Use the action buttons to perform an operation such as a password checkout or session launch or click the name-link to be brought to the managed passwords page or relevant page for the account in question.

- **Shared Passwords:** The Shared Passwords panel allows a user to search for managed, stored, or personal passwords which have been shared with them by other users.
- **Personal Passwords:** The Personal Passwords panel allows a user to search for personal passwords stored in their Personal Password Store.
- **Status and Welcome:** These two panels are default example panels. The welcome panel contains links to other commonly used pages in the web application. Status simply displays a message about how you may edit these panels. These panels may be removed or edited as desired. They provide no functional value.

## User Settings

Every user can control their own user settings.

- **Cached User Email Address:** This field will get populated initially from the email field in Active Directory if available. If not, the field will be blank. The user may change the password on this page, or it may be defined during a password/access request.
- **AJAX Data Load Timeout:** Typically, this setting does not need to be changed. Default is 10000. This identifies the time in milliseconds for the data to load in the background before considering the page load a failure.
- **AJAX Data Page Limit:** Typically, this setting does not need to be changed. Default is 5. This identifies the number of data pages to query for and cache. This setting is multiplied by the number of AJAX Data Page Size to determine the number of total records returned and placed into user cache.
- **AJAX Data Page Retry Limit:** Default is 10. This identifies the number of retries to attempt a failed web service call before marking the call a total failure.
- **Hide Asset Tags:** Every system and account store can have an asset tag that may or may not be populated by an administrator. Enabling this option will show any assets tags for any systems the user has access to. This can be helpful when trying to retrieve the correct system password.
- **Hide Password Comments:** Every password change job can associate a comment with the password. Enabling this option will show that comment.
- **Hide System Information Columns:** When systems are refreshed, information like NetBIOS name or network name and IP address and other information can be captured. Enabling this option will not display this information to the user.

User Settings
Cached user email address
None
Cached user email address
30
AJAX data load timeout
30
AJAX data page limit
5
AJAX data page retry limit
10
<input type="checkbox"/> Hide asset tags
<input type="checkbox"/> Hide password comments
<input type="checkbox"/> Hide system information columns
<input type="checkbox"/> Show password comment column

## RDP Settings

Every user can modify their default RDP settings. Initial settings are based on the web application defined RDP settings. These settings are not related to the application launcher.

### Display

Display settings control how the RDP window is sized when an RDP session is launched.

RDP Settings	
<b>Display</b>	
Desktop Size	1280 x 768
Color Bit Depth	16
Display The Connection Bar When Fullscreen	<input checked="" type="checkbox"/>
<b>Local Resources</b>	
Remote Audio Playback	Play on this computer
Remote Audio Recording	Record from this computer

- **Desktop Size:** The default RDP Window size.
- **Color Bit Depth:** The default RDP color depth.
- **Display the Connection Bar When Full Screen:** When enabled, will display the RDP connection bar when in full screen.

### Local Resources

Determines how local desktop resources will be mapped into the RDP session.

- **Remote Audio Playback:** Determines how audio from the RDP target plays back through local speakers.
- **Remote Audio Recording:** Determines where RDP audio will be recorded if being recorded.
- **Apply Windows Key Combinations:** When key combinations are used, such as ALT+TAB, determines if the combination will be mapped to the local system or remote system.
- **Redirect Printers:** When enabled, will allow the remote system to print to locally attached printers.
- **Redirect Ports:** When enabled, will allow local port access from the remote server.
- **Redirect Smart Cards:** When enabled, will allow local smart card access from the remote server.
- **Redirect Clipboard:** When enabled, will allow local clipboard access from the remote server.
- **Plug N Play Devices:** Supported plug n' play devices will be accessible from the remote server.
- **Redirect Local Drives:** When enabled, local drives (e.g. C:, D:) will be available as a mapped drive on the remote server.

### Experience

- **Connection Speed:** Pre-determines the connection speed (and thus related settings) rather than permitting the client and server to negotiate the speed and related settings.
- **Allow Desktop Background:** The remote system's desktop wallpaper will be displayed over the RDP connection. Showing desktop background can slow performance. Disable to improve performance.
- **Show Windows Contents While Dragging:** When dragging a window, the dialog will be displayed if enabled. Disable to improve performance.
- **Disable Menu Animations:** Menus that have animations will show the animations when rendering. If the option is enabled, the menu will simply appear without animation. Disable to improve performance.
- **Disable Themes:** When enabled, the colors, rounded borders, etc. will not be used.
- **Enable Bitmap Caching:** When enabled, bitmap resources are locally stored on the client computer for reusing them later.

## Advanced

- **If server authentication fails:** If the server authentication fails, this setting determines what the RDP control will do.



## SSH Settings

Every user can modify their default SSH settings. Initial settings are based on the web application defined SSH settings. These settings are not related to the application launcher.

SSH Settings	
Connection Settings	
Enable SSH Console	<input checked="" type="checkbox"/>
Allow SSH To Any System	<input checked="" type="checkbox"/>
Allow Multiple SSH Sessions	<input checked="" type="checkbox"/>
Enable Key Timing Noise	<input type="checkbox"/>
Enable X11 Forwarding	<input type="checkbox"/>
Allow New Server	<input type="checkbox"/>
Enable Telnet Console	<input checked="" type="checkbox"/>

### Connection Settings

- **Enable SSH Console:** If disabled, the SSH Java applet for Linux/Unix targets will not be displayed.
- **Allow SSH To Any System:** Will permit the managed account to be used to connect to any system, if the target system permits it.
- **Allow Multiple SSH Sessions:** Will allow the launching of multiple SSH windows from the web client. If this box is cleared, the current Telnet session will be disconnected before the new session is established.
- **Enable Key Timing Noise:** Enable to create a random timing offset for key transfer (security).
- **Enable X11 Forwarding:** If X11 forwarding is enabled on the target host, this will enable the feature to function in the Java-based SSH session.
- **Allow New Server:** Enable to permit jumping from server to server from within the SSH session.
- **Enable Telnet Console:** If disabled, the Telnet Java applet for Linux/Unix targets will not be displayed.
- **Allow Telnet To Any Systems:** Will permit the managed account to be used to connect to any system, if the target system permits it.
- **Allow Multiple Telnet Sessions:** Will allow the launching of multiple Telnet windows from the web client. If this box is disabled, the current Telnet session will be disconnected before the new session is established.
- **SSH Type:** When set to Auto, the control will determine what the target supports and use that. Force a particular version if desired.
- **SSH Port:** The SSH target port
- **SSH Connection Timeout:** Initial connection timeout.
- **SSH Handshake Timeout:** Amount of time for the connection handshake to take place
- **SSH Key Exchange Timeout:** Amount of time for the key exchange to take place
- **SSH Proxy Type:** Both the SOCKS and HTTP proxy protocols can be used to traverse firewalls. SOCKS is usually used to create a raw TCP connection, and the HTTP proxy protocol can do the same with the CONNECT method. If a proxy is required, also supply the SSH Proxy Host, SSH Proxy Port, and SSH Proxy Timeout.
  - **SSH Proxy Host:** The name/IP of the SSH proxy host that must first be connected to for an SSH connection.
  - **SSH Proxy Passphrase:** The passphrase to use the SSH Proxy Host.
  - **SSH Proxy Port:** The target SSH port for the SSH connection when using an SSH Proxy Host.
  - **SSH Proxy Timeout:** The timeout for the SSH proxy connection in seconds.
- **Enable Public Key Encryption for SSH:** If SSH keys are configured in Privileged Identity, the Java-based SSH sessions may leverage keys to connect to the target systems.
- **Key location on client system:** The physical path on the client's workstation where the SSH keys are physically stored.
- **Allow Clients to Specify Private Key Paths:** Enable to allow the user to identify the public key path on their own system rather than relying on the globally configured option.

## Console Settings

- **Send on Backspace:** Character to send on BACKSPACE: BS (^h, 0x08), DEL (^?, 0x7f), or ERASE (^E[3~]).
- **Send on Delete:** Character to send on DELETE: BS (^h, 0x08), DEL (^?, 0x7f), or ERASE (^E[3~]).
- **Scrollbar Position:** Relative scrollbar position (none/left/right).
- **Mouse button to paste:** Click the **mouse** button to paste the copy buffer.
- **Terminal Type:** Name of terminal to emulate (xterm, linux, scoansi, att6386, sun, aixterm, vt220, vt100, ansi, vt52, xterm-color, linux-lat, at386, vt320, vt102 and tn6530-8).
- **Background Color:** Color of the background.
- **Cursor Color:** Color of the cursor.
- **Foreground Color:** Color of the foreground window.
- **Console Rows:** Number of rows to display in the terminal.
- **Console Columns:** Number of columns to display in the terminal.
- **Font Name:** The font name to use in the terminal.
- **Font Size:** Size of the font displayed in the terminal
- **Line space Delta:** Number of pixels to modify the line spacing with.
- **Line Buffer:** Number of lines to save in scroll back buffer.
- **Display ASCII:** Use ASCII Line-draw-characters instead of drawing.
- **Auto Linefeed:** Do auto-linefeed.
- **Auto Wrap:** Auto wrapping of line if output reaches edge of window.
- **Send <CR><NL> not <CR> for copy/paste:** Put <CR><NL> instead of <CR> at end of lines in copy/paste.
- **Copy On Mouse Select:** Copy directly on mouse-selection.
- **Send <CR><LF> not <CR><NUL>:** Send carriage returns as Telnet <CR><LF>.
- **Ignore NULL Inputs:** Ignore any null bytes in the data-stream.
- **Insert Mode Enabled:** Toggles insert mode.
- **Local Echo:** Enable local echo.
- **Local Page keys:** Use PgUp, PgDn, Home, End keys for local scroll or escape them.
- **Map Ctrl+Space to NULL:** Typically used for emacs.
- **Reposition Screen To Bottom On Input:** Reposition scroll-area to bottom on keyboard input.
- **Reposition Screen To bottom On Output:** Reposition scroll-area to bottom on output to screen.
- **Allow Resize:** Allow the window size to be changed or fixed.
- **Visible Cursor:** Toggles if cursor is visible or not.
- **Visual Bell:** Toggles if audible or visual bell will be used.

## Server Certificates

The server certificates section allows a user to download the certificates used by the web application server and media server if streaming sessions recorded from the application launcher. The certificates will be available if the certificates were deployed with the web application. If the certificates are self-signed or deployed from a non-trusted CA, this represents a way for users to quickly and easily gain access to the certificates the admin wishes for them to use.

Server Certificates	
Server Certificate	Not configured for download
Recording Server Certificate	Not configured for download

- **Server Certificate:** If a certificate was included in the web application deployment and defined in the global web site settings, the user may download the certificate from this link.
- **Recording Server Certificate:** If a certificate was included in the web application deployment and defined in the global web site settings, the user may download the certificate from this link.

## Web Services

When deploying the web application, the REST web service endpoint is required for the web application to fully function. In the context of a web application user, the web services section is a reference for users who may need to refer to these URLs for other projects.

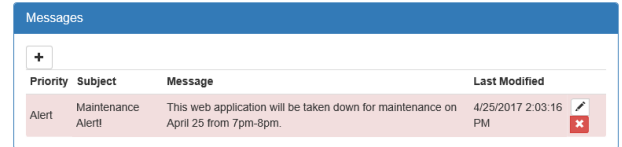
Web Services	
WSDL Web Service Endpoint	<a href="https://lsdlsicprd.lsdls.int/erpmwebservice/authservice.svc">https://lsdlsicprd.lsdls.int/erpmwebservice/authservice.svc</a>
REST Web Service Endpoint	<a href="https://lsdlsicprd.lsdls.int.443/ERPMWebService/AuthService.svc/REST">https://lsdlsicprd.lsdls.int.443/ERPMWebService/AuthService.svc/REST</a>



- **WSDL Web Service Endpoint:** Will list the the full URL to the SOAP-based web service if defined in the global web site settings. This is for user reference only.
- **REST Web Service Endpoint:** Will list the the full URL to the REST-based web service if defined in the global web site settings. This is for user reference only.

## Message Center

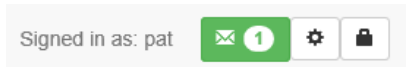
The message center is used by web application administrators to notify users of impending actions or add other notes the users should be aware of.

Click the **Add Message** button (+) to add a new message.



Messages			
+			
Priority	Subject	Message	Last Modified
Alert	Maintenance Alert	This web application will be taken down for maintenance on April 25 from 7pm-8pm.	4/25/2017 2:03:16 PM  

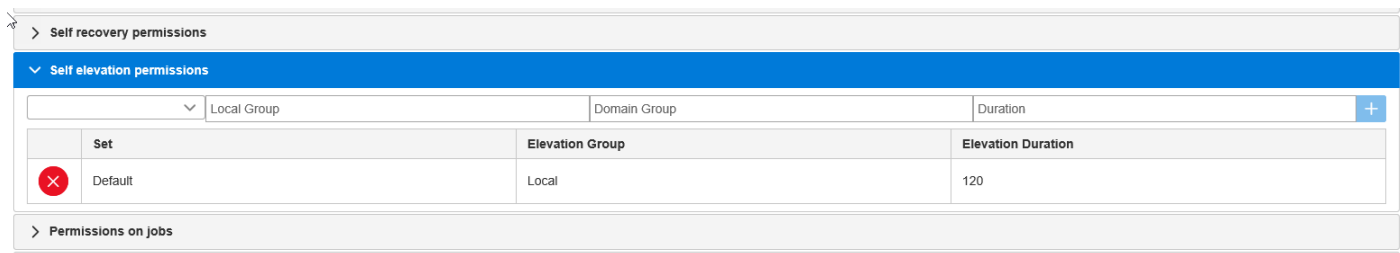
If the message center contains messages the user has not yet read, a green mail envelope indicating the number of new messages will appear in the top right corner of the web application.



# Configure Delegation

## Manage Self-Elevation Permissions in the Web Application

1. Click **Settings** on the top menu, and select **Delegation**.
2. Click the ellipsis at the start of the row for the **Identity Name** to which to grant this permission.
3. Click **Edit Identity** to change the permissions.
4. Click **Self elevation permissions** to expand the section. It may be necessary to scroll down to see the section.



Set	Elevation Group	Elevation Duration
Default	Local	120

5. Complete the information in the row.
  - **Management Set:** Enter or select the management set to which to grant this permission.
  - **Local Group/Global Group:** Enter the scope of the target group.
  - **Domain Group:** Enter the domain of the target group.
  - **Max Elevation Time:** Define the maximum time in minutes for which the user may elevate.
6. Click the blue plus sign (+) at the end of the row to add the permission.
7. Existing permissions display below the row for adding permissions. To remove an existing permission, click the red **X** at the start of the row.



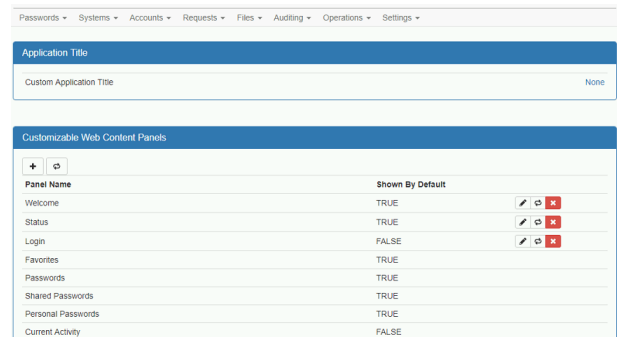
To manage self-elevation in the web application, please see "[Self-Service Elevation - Advanced](#)" on page 465.

## Customize Web Content

Web content can be customized by a user who has All Access.

The custom content page is divided into two sections:

- **Application Title:** Specify the name of the application that appears in the top left corner of the web application. When not defined (None), the application title will be Privileged Identity.
- **Customizable Web Content Panels:** This section identifies default and admin added web panels (available to all users). This section also identifies the charts available to users with the View Dashboards delegation. Panels that can be edited or removed will have editing icons to the right of them. The **Shown by Default** column will have a value of **True** or **False** for each panel. True indicates it will be added to a user's main page when they log in. The user can then remove the panel if desired from their own sessions.
- **Favorites:** Shows any items the user has added as a favorite item. These are essentially shortcuts to passwords or sessions the user refers to frequently. With a favorite, the user can simply click the favorite link and be brought right to the item in question without having to fully navigate through other menus first.



Click the **ellipsis** button to launch the favorite, rename it, or to delete it. Click the **X** button to remove the **Favorites** panel from the web app homepage.

- **Passwords:** The **Passwords** panel allows a user to search for managed passwords using a free text search. This panel can save the user time by allowing a partial match search for system name, account name, or name space to locate a managed password rather than having to navigate and filter.  
Type in the [partial] name of a system, namespace, or account name and click the **filter** button on the right. A list of matching items will appear in the list. Use the action buttons to perform an operation such as a password checkout or session launch or click the name-link to be brought to the managed passwords page or relevant page for the account in question.
- **Shared Passwords:** The Shared Passwords panel allows a user to search for managed, stored, or personal passwords which have been shared with them by other users.
- **Personal Passwords:** The Personal Passwords panel allows a user to search for personal passwords stored in their Personal Password Store.
- **Status and Welcome:** These two panels are default example panels. The welcome panel contains links to other commonly used pages in the web application. Status simply displays a message about how you may edit these panels. These panels may be removed or edited as desired. They provide no functional value.

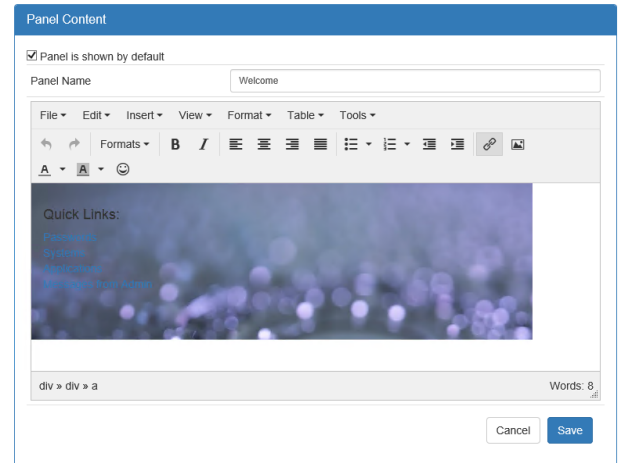
## Customize a Web Panel

Any panel created by an admin, can be viewed by any user who subsequently adds the panel. There are no panel or per panel delegations.

1. Edit an existing panel by clicking the **Edit** button (pencil) next to the panel. Add a new panel by clicking the **New Panel** button (+).
2. If desired, Enable **Panel is shown by default** to add make this panel shown automatically to new user sessions. Users who have logged in previously will need to add the panel by enabling it in their user settings.
3. Supply a name for the panel in the **Panel Name** field. This name will be visible to all users.

4. Supply/Edit your content.
5. Click **Save** when done.

This is a basic content management system. Use the WYSIWIG editor (what you see is what you get) to add text, format text, insert links, etc. If you prefer to edit the HTML source code directly, go to **Tools <> Source Code** and write HTML code.





## Message Templates

Privileged Identity ships with a number of predefined message templates that are used when certain things occur, such as an account elevation, password recovery alert, etc. Templates can be modified from either the web application or the management console under **Settings > Message Templates**.

Message templates are stored in the program's data store. There is no need to distribute message templates with versions of Privileged Identity after v5.4.0.

When editing the messages, the message opens in the built-in CMS editor.

 For more information, please see "[Customize Web Content](#)" on page 491.

## Message Templates

Variables available to all templates except **Status Reports**, include:

- **\$ApplicationName\$**: Inserts the name of the application.
- **\$ApplicationVersion\$**: Inserts the working version of the application.
- **\$LocalSystemName\$**: Inserts the name of the Privileged Identity host that generated the message.

## Account Elevation Jobs

The following variables are available to account elevation jobs:

- **\$ApplicationName\$**: Inserts the name of the application.
- **\$ApplicationVersion\$**: Inserts the working version of the application.
- **\$FullAccountName\$**: The full account name of the identity being manipulated.
- **\$ElevationGroup\$**: The target group to add the identity to.
- **\$ElevationSystem\$**: The target server for the job.
- **\$ElevationExpirationMinutes\$**: The number of minutes until the identity is removed from the target group.

Following are the account elevation message templates:

- **Account Elevation Expiration Alert**: This message is sent when an account elevation is going to expire.
- **Account Elevation Success Alert**: This message is sent when the account elevation job succeeds.
- **Account Elevation Failure Alert**: This message is sent when the account elevation job fails.
- **Account De-elevation Success Alert**: This message is sent when the account elevation job successfully removes the user from the elevation group.
- **Account De-elevation Failure Alert**: This message is sent when the account elevation job fails to remove the user from the elevation group.

## SMS Token Alert Emails

Extra variables available to SMS Token Alert Emails:

- **\$TokenCode\$**: The token code the user requires for OATH MFA.

Following are the SMS token alert email message templates:

- **Token Code Email Message**: The email message sent to a user containing their OATH pin code.
- **Token Code SMS Message**: The SMS message sent to a user containing their OATH pin code.

## Password and SSH Key Action Alerts

Extra variables available to password and SSH key action alert emails:

- **\$FullAccountName\$**: The full account name of the identity being manipulated.
- **\$Comment\$**: The recovery or request comment.
- **\$ADDisplayName\$**: The display name of the account in Active Directory, if available.

Following are the password action message templates:

- **Password Checkout Expiration Alert**: Message sent to user who checked out a password, prior to the check-out expiring. This template also uses the following variables:
  - **\$ExpirationMinutes\$**: Number of minutes until the password check-out expires.
- **Password Recovery Alert**: Message sent to when a password is retrieved. This template also uses the following variables:
  - **\$RecoveryTime\$**: When the password was recovered.
  - **\$RecoveredBy\$**: Who the password was recovered by.
- **Password Request Alert**: Message sent to the request granter when a request is made for a password or SSH key. This template also uses the following variables:
  - **\$RequestType\$**: Indicates if the request is an incident or change.
  - **\$RequestedAccountSystem\$**: The name of the target system.
  - **\$RequestedAccountNamespace\$**: The namespace of the requested target account.
  - **\$RequestedAccountName\$**: The name of the requested account.
  - **\$RequestTime\$**: The time when the user is requesting access. This time is *now* or some point in the future.
  - **\$RequestTimeUTC\$**: The UTC time when the user is requesting access. This time is *now* or some point in the future as displayed in UTC.
  - **\$RequestingUser\$**: The name of the identity requesting access.
  - **\$RequestingUserIP\$**: The IP address of the user requesting access.
- **Password Request Denied Message**: Message sent to the requester when their password or SSH key request is denied. This template also uses the following variables:
  - **\$RequestDenyUsername\$**: The user who denied the request.
  - **\$RequestDenyTime\$**: The time when the request was denied in local time.
  - **\$RequestDenyTimeUTC\$**: The time when the request was denied in UTC time.
- **Password Request Granted Message**: Message sent to the requester when their password or SSH key request is granted. This template also uses the following variables:
  - **\$RequestGrantWindow\$**: The amount of time the requesting user has to retrieve the password.
  - **\$RequestGrantTime\$**: The time the request was granted in local time.
  - **\$RequestGrantTimeUTC\$ UTC**: The time the request was granted in UTC time.

- **\$RequestGranterUsername\$**: The name of the user who granted the request.

## Status Reports

- **Web Application Compatibility Test**: This page is used via the interactive management console during a web application status test (**Check Status** button in **Manage Web App** dialog). Variables include:
  - **[%INSERT\_DATE%]**: The date the report was run.
  - **[%INSERT\_SERVER\_COMPATIBILITY\_STATUS%]**: Returns *Passed* or *Failed* when checking the target web server Remote IIS, ADSI, and COM accessibility.
  - **[%INSERT\_SERVER\_COMPATIBILITY\_DATA%]**: Returns status information for required remote interfaces.
  - **[%INSERT\_WEB\_APPLICATION\_STATUS%]**: Returns *Passed* or *Failed* for web application settings matching the querying console.
  - **[%INSERT\_WEB\_APPLICATION\_SETTINGS\_DATA%]**: Returns application data such as DB and encryption configuration settings matching the console.
  - **[%INSERT\_COM\_COMPONENT\_STATUS%]**: Returns *Passed* or *Failed* for COM Component Settings availability.
  - **[%INSERT\_COM\_COMPONENT\_DATA%]**: Returns COM Component settings information.
  - **[%INSERT\_IIS\_SITE\_STATUS%]**: Returns *Passed* or *Failed* for target IIS requirements such as ASP.
  - **[%INSERT\_IIS\_SITE\_DATA%]**: Returns IIS Site/Virtual Directory information.
  - **[%INSERT\_DASHBOARD\_STATUS%]**: Returns *Passed* or *Failed* for .NET Dashboard components.
  - **[%INSERT\_DASHBOARD\_DATA%]**: Returns .NET Chart control assembly information.
- **Password Status Report**: This template is the password status report for testing the viability of stored passwords. It is used when a password is tested from the management console stored managed passwords dialog or when a password verification job is run. Variables include:
  - **[%INSERT\_DATE%]**: The date the report was run.
  - **[%INSERT\_GROUP\_NAME%]**: The name of the target management set.
  - **[%INSERT\_PASSWORD\_DATA%]**: The password test results.
- **Compliance Report**: The template for compliance reports.
  - **[%INSERT\_COMPANY%]**: Inserts the company name and related information. This is always *BeyondTrust*.
  - **[%INSERT\_REPORT\_TYPE%]**: Identifies the type of report being generated.
  - **[%INSERT\_REPORT\_DATA%]**: Inserts the compliance report data.
  - **[%INSERT\_ADMIN%]**: Identifies the admin who generated the report.
  - **[%INSERT\_DATE%]**: Identifies the date the report was generated.
- **Admin Report**: The default administrative report template. This shows job activity status for every password change job, ever.
  - **[%INSERT\_JOB\_ACTIVITY%]**: Inserts the job activity report data.

## Remote Applications

Remote applications can be added, edited, or deleted by a user who has All Access permissions in the web application.




*For more information, please see the [Application Launcher & Session Recording guide](#).*


Remote Application Label	Target	Arguments	Run Remote
Amazon AWS	Login_AWS.vbs	https://console.aws.amazon.com/console/home	FALSE
APC PDU	Login_APC_PDU.vbs	http://\$(RemoteAccessTarget_TargetName)/logon.htm	FALSE


## Live Activity

The live activity page can be used by a user with All Access to see what managed or shared passwords and files are currently checked out. This page also allows the All Access user to forcibly check the item back in, thus making the item available to other users. This feature is useful to all users who cannot complete their work when the resource (password or file) has been checked out by another user, who is unavailable to check the resource back in.

Click the **Force Check In** button (red exclamation mark) to forcibly check the item back in.

Checked Out Managed Passwords				
System	Namespace	Account	Checked Out By	ExpirationTime
DBAG01	DBAG01	Administrator	Isds\lscadmin	4/25/2017 5:42:24 PM 

Checked Out Shared Passwords				
Password List	System	Account	Checked Out By	ExpirationTime
OATH Seed Values	OATHSeedValue	robert	Isds\lscadmin	4/25/2017 5:42:48 PM 

Checked Out Files		
File Name	Checked Out By	ExpirationTime
LSC Certificate for Web-1.cer	Isds\lscadmin	4/25/2017 5:43:07 PM 

## Unlock Locked Out Accounts

An account that is locked out can be unlocked within the web application.

1. Click **Settings** on the top menu, and select **Lockout**.
2. A list shows all locked out accounts by **User**, with the number of **Failed Attempts** and **Lockout Time** for each user.
3. Click the blue lock icon at the start of a row to unlock that account.



*To set or change the lockout policy in the web application, please see "[Configure Site Settings](#)" on page 515.*

## Configure Event Sinks in the Web Application

Privileged Identity features an extensible eventing system that allows an administrator to receive alerts or take action when events occur. Many actions can trigger messages that can be forwarded to other systems (for example, ticketing or SIEM systems) using the Event Sink system.

An event sink, sometimes called a *listener*, is a piece of code that defines how a server or computer is to handle given events. Event sinks are often used in spam filters to trigger actions in response to the receipt of an email message with defined characteristics or certain types of attachments. The destination of data handled by such a program is also sometimes called an event sink.

The purpose of the Event Sink system is to provide alerts and a framework for data feeds or integrations to third party systems. It was also designed with its own limitations in mind, and supports things like the arbitrary program invocation. In the case of Privileged Identity, actions such as password check-outs, password randomizations, failed propagations, and other events can trigger an event sink.

Each event sink is a registered listener that takes a specific action when one or more events occur. Each event sink can be configured to respond to a single event, a range of events, or multiple ranges of events. The action that an event sink takes is determined by the configuration of that specific event sink. Event sinks can be created, configured, and deleted from the management console or from the web application. The files are located in a directory specified by the application settings. By default, it is the program installation directory.

Once the event server is started, components of the application (console, web application COM object, deferred processors, zone processors, and so on) pass events to the event server to be processed asynchronously. When the event server receives an event, it checks the ranges of each registered event sink object to see if the event should trigger the sink. The event server triggers every event sink that is listening for a specific event.

Each event sink contains an output type. The output type determines what action is taken when an event is processed by the event sink. All events that match the event filter settings are sent to the specified output type.

Event sink event IDs are grouped into sections based on the type of operation. Groups of operations each contain 1000 possible events, although not all 1000 event IDs are used in each range.



**Note:** Some event sinks, while visible and selectable from the UI, are not used or reported by the Privileged Identity. These event sinks are noted with the following prefix: **-- Not Available --**.

If an event sink has been replaced by another event sink, the new event sink is listed in the event sink description. Otherwise, the deprecated event sink is no longer functional.

The instructions in the following sections describe how to configure event sinks from the web application.

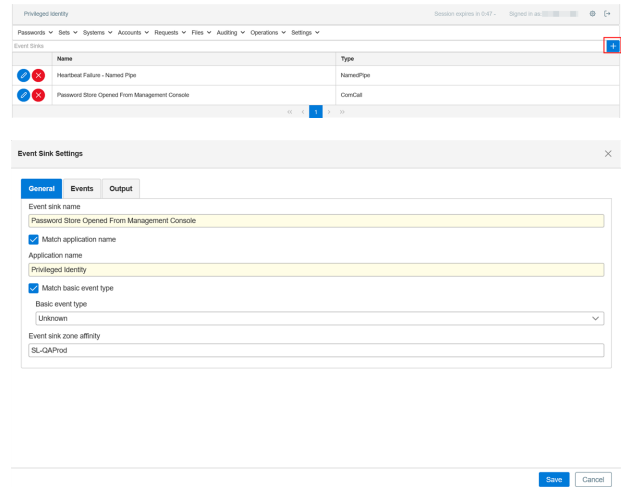


For more information, please see the following:

- For configuring event sink output types, "[Configure Event Sink Output](#)" on page 544
- For specific events, "[Event Sink Events List](#)" on page 536
- For configuring event sinks from the management console, "[Configure Event Sinks in the Management Console](#)" on page 542

## Create an Event Sink

1. In the web application, navigate to **Settings > Event Sinks** from the top menu.
2. Click the **+** button in the top-right corner of the page.



3. On the **General** tab, supply the following:

- **Event sink name:** This name cannot be changed once created.
- **Match application name:** When this option is checked, any event sink triggered by this application fires any event sink possible for this component. **Privileged Identity** is the only valid application name.
- **Match basic event type:** When this option is checked, it triggers any event from a specific scope. Valid scopes are:
  - **Unknown:** There is no other predefined event sink.
  - **Generic Success:** Triggered by any success message.
  - **Generic Failure:** Triggered by any failure message.
  - **Trace:** Triggered by debug specific messages when debugging is enabled for the system.
  - **Job Processing Start:** Triggered when any job begins.
  - **Job Processing Status Update:** Triggered by any updates during a job run.
  - **Job Processing Complete, Success:** Triggers when any job successfully completes.
  - **Job Processing Complete, Failure:** Triggers when any job fails to complete successfully.
  - **Operation Notification, General:** Any operation that sends a notifications and that notification has a status message, including success and failure.
  - **Operation Notification, Success:** Any operation that sends notifications and that notification has a success message.
  - **Operation Notification, Failure:** Any operation that sends a notification and that notification has a failure message.
  - **Operation Status Update:** Any operation that has a status message.
  - **Operation Failure:** The operation failed.
  - **Application Component Status Update:** Any update messages from any component.
  - **Application Component Internal Error:** Any component used during a job encountered an error.

The filters are cumulative, meaning the event sink processes the event only if it matches all of the filter settings. For example, if the event must match the Application Name and an event ID range is also provided, then the event must match the Application Name and must also fall into the event range.

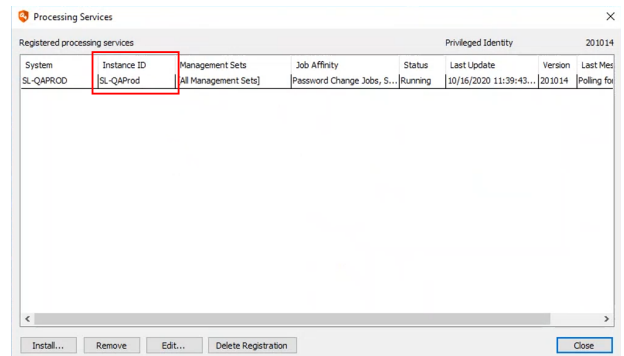
If filtering for events that match the application name, only events generated by the application with the same name are processed by the event sink. By default, the name of the application in the filter field is the same as the name of the application which creates the dialog. All the components of an application use the same application name for the purposes of identifying events (web application, deferred processors, zone processors, etc).

If filtering for a specific event type, only events that match the type are processed by the event sink. Event types include:

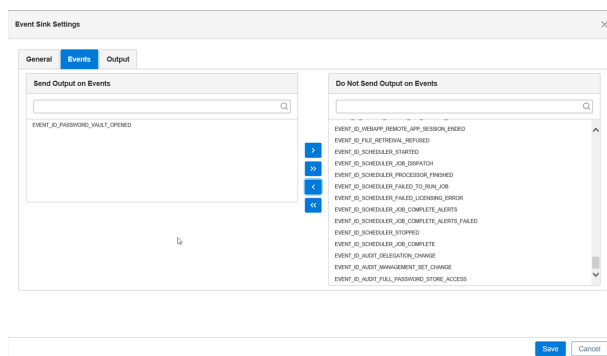


- **Success:** Indicates that an operation completed.
  - **Failure:** Indicates that an operation failed.
  - **Error:** Indicates that an error occurred during the course of operation.
  - **Debug Trace:** Indicates that the event was triggered as a debug diagnostic step.
  - **Status Update:** Indicates that the event was triggered to provide additional information during the course of a normal operation.
  - **Unknown:** Indicates an unknown type of event.
- **Event sink zone affinity:** In a case where an event sink is processed by a server that resides in an environment where not all Privileged Identity processors have access to that server, you may want to configure the event sink so that it processes by a specific zone processor in a zone where all integration components can access the server processing the event. This ensures the processor can reach the target to process the event and send the event output. Enter the ZoneID to configure the event sink to use the zone processor in that specific zone.

**Note:** The ZoneID entered must exactly match the Instance ID for the zone processor, as shown on the **Processing Services** window in the management console.



4. Select the



**Even**

**ts** tab, and then select the specific events you wish to add to the event sink. Use the arrows to add and remove them individually or all at once.

5. Select the **Output** tab to configure the setting for the output type.
6. Click **Save** to save the event sink.

## Configure Event Sink Output

The event sink listener configuration controls the action taken when an event is processed by the event sink. Currently, an event sink can do any of the following:

- **Log File:** Write the message to an output file. The only additional argument to specify here is the name of the file that you want to write the messages to. If the file does not exist, it is automatically created when the first event is processed.
- **Set Registry Value:** Write the message to a value in the registry. Specify the system, base registry key, registry path, and registry value to write the event message. Specify an event notification name.
- **Named Pipe:** Write the message to a named pipe. Specify the name of the named pipe to write the event message data to. The resulting event message is sent as text over the pipe.

- **COM Call:** Call a function on a COM object and pass the message as an argument. Specify a COM Program ID and a COM method name on that interface. The resulting event message is passed as a BSTR argument to the function.
- **Send Email:** Send an email containing the message. Specify an SMTP email configuration profile as well as a semicolon delimited list of message recipients. By default, if the SMTP email settings for the application are configured already, then the SMTP configuration exists with the configuration name **Default**.

If an alternate email configuration is required for the event sink, then create a different configuration in the registry by creating a new key under the location **HKLM\Software\Lieberman\SmtSettings** and copy the values from default or change the values as desired. Emails can be sent to single users or to distribution lists using this method.

- **Event Log:** Write the event to an application event log. Specify the target windows server, event type, and event ID.
- **Syslog:** Write the event to a syslog compatible system. Specify the target syslog server, and proper format as syslog format (no options), ArcSight CEF format, or QRadar LEEF format.
- **MSMQ:** Write the event to a Microsoft Message Queue. Specify the target MSMQ server and queue.
- **Run Specific Application:** Run any local program on the application or web server. Define the path on the host web or application server and any required parameters.
- There are also a number of default integrations you can select from the list such as ArcSight, QRadar, BMC Remedy, Microsoft System Center Service Manager, ServiceNow, and others.

Each output type by default receives only the raw event message when an event is processed. Use a transformation to modify the format or information that is sent before it is sent to the output type. To use a transform, select **Custom transform file** from the **Output data format** list, and then specify the path to the **Transform file**. The same transform file can be leveraged for multiple event sinks. A transform file can be created and modified directly from the dialog. The default transform editor used to open the transform files can be configured through the Transform Editor options. The default transform editor is Windows Notepad, located at the default location. The file type and format of the event messages using transform files can also be set. For example, to send HTML file email alerts to an email list, specify an HTML transform file with the correct formatting for an HTML email message, and the recipients will receive HTML encoded email messages in the same format as the transform file when the event corresponding to the event sink occurs.

The following sections describe how to configure specific output types.

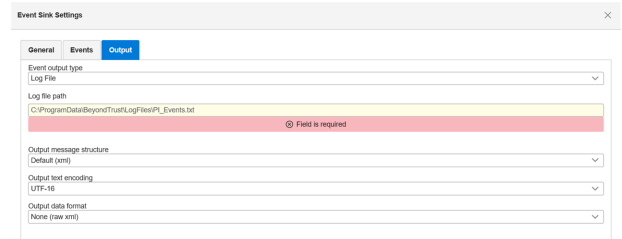


*For more information on transform file formatting and replaceable arguments, please see "Event Sink Transform Files" on page 568.*

## Configure Log File Event Output Type

The event sink log file outputs the event information to a flat text file at the chosen location. If the file does not exist the first time the event sink is triggered, it is created automatically.

Select **Log File** for the **Event output type**, and then specify the path and the name of the log file to generate, or leave the default of **c:\ProgramData\BeyondTrust\LogFiles\PI\_Events.txt**. Select the **Output message structure**, **Output text encoding**, and **Output data format** from the available lists or leave as default.



*For more information on the message structure, text encoding, and data format of the output, please see "[Event Data Message Format](#)" on page 566.*

## Configure Set Registry Value Event Output Type

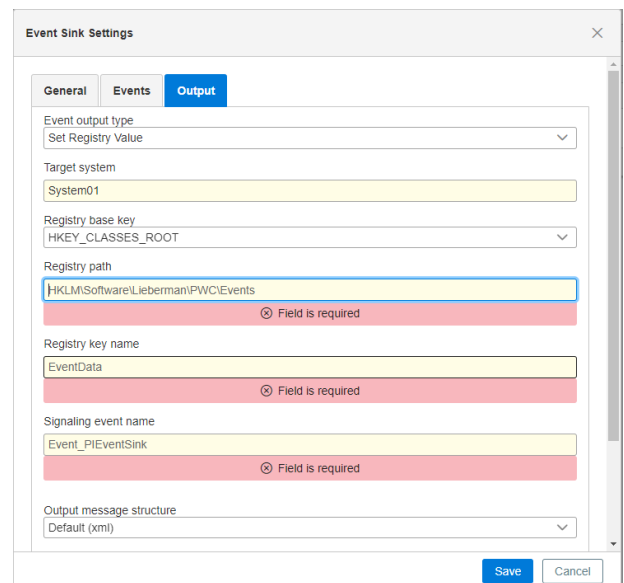
The event sink registry value outputs the event information to a registry path value on the target server. If the file registry value does not exist the first time the event sink is triggered, it is created automatically.

The process works as follows:

1. An event is configured to output to a target Windows host registry.
2. The sink then writes the event sink information to the target system at the registry location specified for the sink.
3. Pending successful write of the registry value, Privileged Identity sends a windows signaling event (e.g clicking a button in a dialog signals OnClick), so that if there is a separate listing application listening for that signaling event, that application can then take additional action. If there is nothing listening for the signaling event on the target system, you may leave the field blank or supply any random name. The OpenEvent function is called on the remote system followed by set event.

Select **Set Registry Value** for the **Event output type**, and then specify the following values:

- **Target system:** Name of the machine which the registry write occurs. This requires access to the remote registry feature (remote registry service must be running). Input is the target server name.
- **Registry base key:** The hive key to write to such as, HKEY\_LOCAL\_MACHINE, HKEY\_USERS, or HKEY\_CLASSES\_ROOT.
- **Registry path:** The path within the base key to write the registry values such as, Software\EventSinks. As Privileged Identity is currently a 32 bit application, The Software registry key is virtual. This means an input of Software\EventSinks is automatically written by the target Windows system as Software\WoW6432Node\EventSinks.
- **Registry key name:** Is a string value located within the registry path such as: PasswordCheckoutData.
- **Signaling event name:** Is the event name sent to the Windows eventing system (not the event logs).



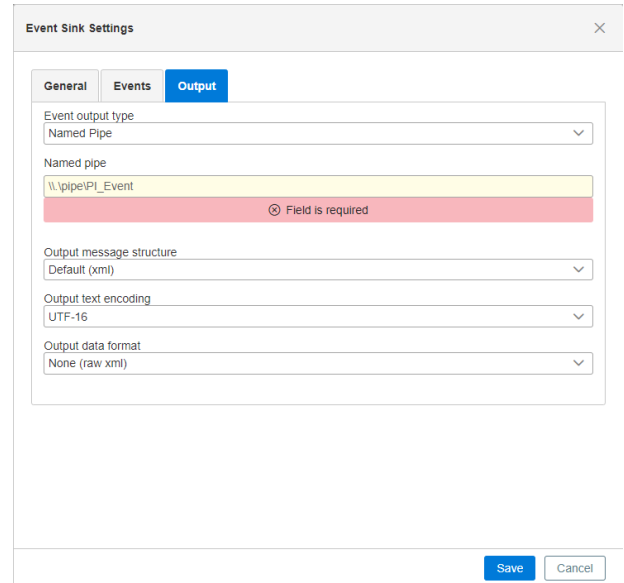

For more information, please see the following:

- [OpenEventA function \(synchapi.h\) at https://msdn.microsoft.com/en-us/library/windows/desktop/ms684305\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/desktop/ms684305(v=vs.85).aspx)
- For information on the message structure, text encoding, and data format of the output, "[Event Data Message Format](#)" on page 566

## Configure Named Pipe Event Output Type

The **Named Pipe** event sink outputs the event information to a specified named pipe call.

Select the **Event output type** as **Named Pipe** and specify its path, and then select the message structure, text encoding, and data format for its event output.



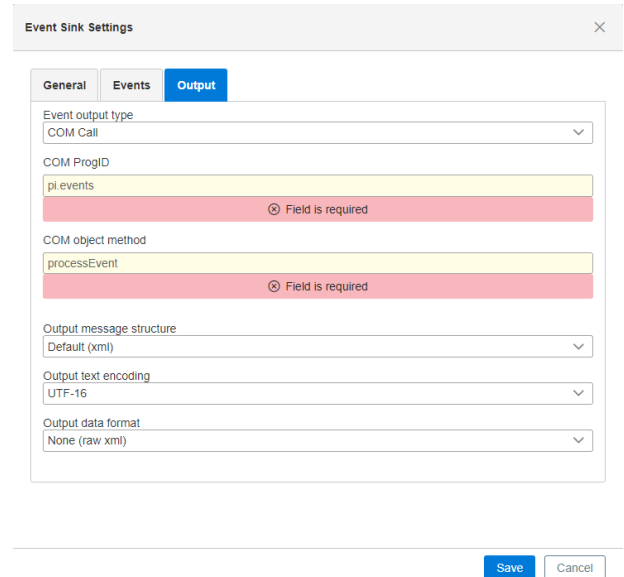
- i** For more information, please see the following:
- For information on the message structure, text encoding, and data format of the output, "[Event Data Message Format](#)" on page 566
  - For more information on how to construct and use named pipes, please refer to Microsoft or other applicable vendor documentation.

## Configure COM Call Event Output Type

The **COM Call** event sink outputs the event information to a specified COM object by calling a method specific to the COM object. The COM object can take any possible action, such as feeding information to a help desk ticketing system or calling other methods or scripts.

Select **COM Call** for the **Event output type**, and then specify the following values:

- **COM ProgID:** The registered name of the COM object.
- **COM object Method:** The method or routine as specified by the COM object.




For more information on the message structure, text encoding, and data format of the output, please see "[Event Data Message Format](#)" on page 566.

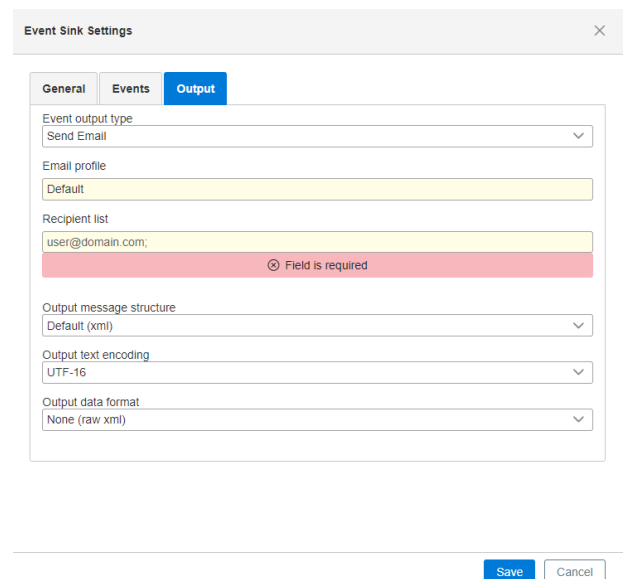
## Configure Send Email Event Output Type

The **Send Email** event sink outputs the event information to the specified email recipients via an email.

An SMTP email configuration profile must be specified as at least one email recipient. By default, if SMTP email settings for the application have already been configured, the SMTP configuration is named **Default**.

Email output types simply write the event message in its entirety to the body of an email or an email attachment.

To format the data using a transform file, select **CustomTransform File** from the **Output data format** list, and then specify the path to the transform file. The same transform file can be leveraged for multiple event sinks. Transform files format the data and included information within the log file.



For the email output, if a transform file is not used, the raw event information is attached to a blank email. A sample transformation file is shown here:

```
<html>
  <head>
    <title>Test</title>
  </head>
  <body>
    Email for password recovery from the web application
    %Message%
    Login Name %sLoginName%
    Checked out account: %ContextVariable:sSystemName%\%ContextVariable:sAccountName%
  </body>
</html>
```



For more information, please see the following:

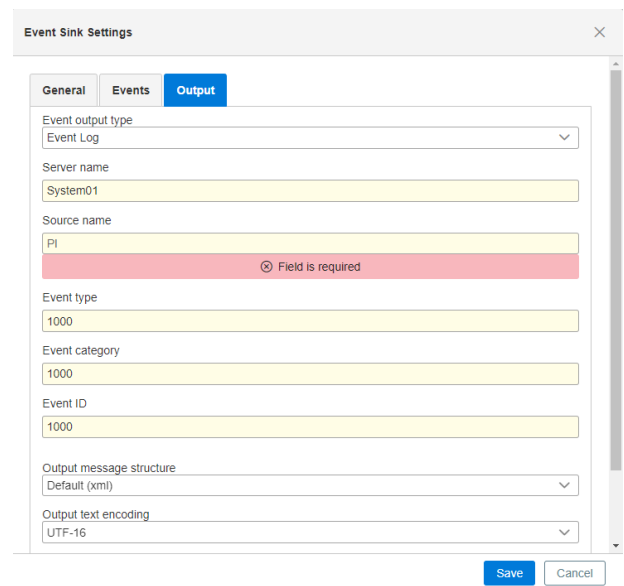
- For transform file formatting and replaceable arguments, "[Event Sink Transform Files](#)" on page 568
- For message structure, text encoding, and data format of the output, "[Event Data Message Format](#)" on page 566

## Configure Event Log Event Output Type

The **Event Log** event sink outputs the event information in its entirety to the application log of the target Windows server.

Select **Event Log** for the **Event output type**, and then specify the following values:

- **Server name:** Name of the system used as the Windows event log server.
- **Source name:** The event source that is written to the event log.
- **Event Type:** Specifies if the type is information, warning, or failure. Event types 0, 1, and 2 are for Information, Error, and Warning, respectively.
- **Event Category:** Specifies the category of the event, which is useful for filters in the event log. Categories are defined by the admin for the sake of information organization.
- **Event ID:** The event ID as it should appear in the event logs for this particular event sink.



The screenshot shows the 'Event Sink Settings' dialog box with the 'Output' tab selected. The 'Event output type' is set to 'Event Log'. The 'Server name' is 'System01'. The 'Source name' is 'PI', with a red error message 'Field is required' below it. The 'Event type' is '1000', 'Event category' is '1000', and 'Event ID' is '1000'. The 'Output message structure' is 'Default (xml)' and 'Output text encoding' is 'UTF-16'. 'Save' and 'Cancel' buttons are at the bottom right.



For more information on the message structure, text encoding, and data format of the output, please see "[Event Data Message Format](#)" on page 566.

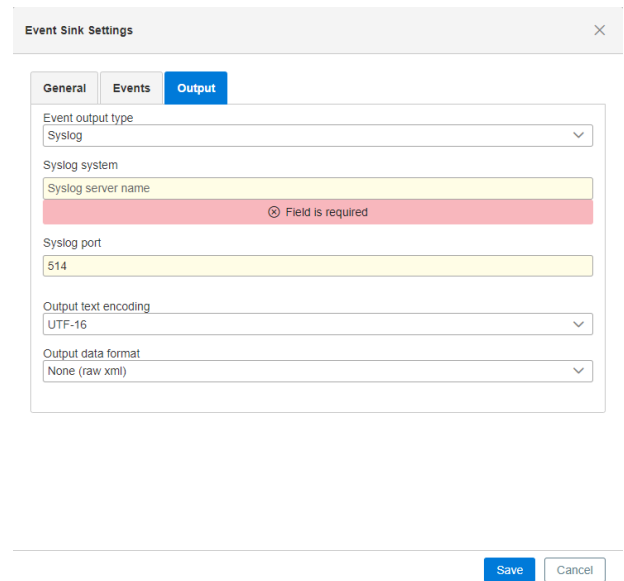
## Configure Syslog Compatible Event Output Types

Both **Syslog** and **IP Address and Port** event sinks output the event information to a target server in a syslog compatible format. The messages are written to the default messages log. **Syslog** uses the historical syslog service and **IP Address and Port** uses the reliable syslog service and functions over TCP or UDP.

### Syslog

Select **Syslog** for the **Event output type**, and then specify the following values:

- **Syslog system:** Name of the server to be used as the syslog server.
- **Syslog port:** The default syslog port is **514** and sends via UDP.

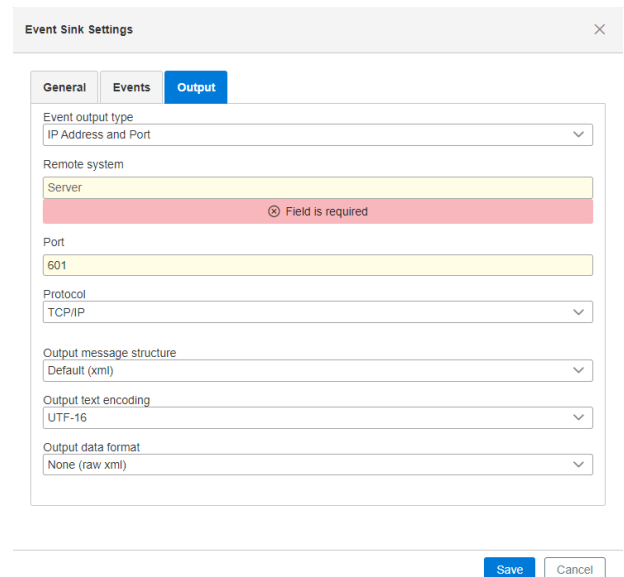


The screenshot shows the 'Event Sink Settings' dialog with the 'Output' tab selected. The 'Event output type' is set to 'Syslog'. The 'Syslog system' section has a 'Syslog server name' field with a red error message 'Field is required'. The 'Syslog port' is set to '514'. The 'Output text encoding' is 'UTF-16' and the 'Output data format' is 'None (raw xml)'. 'Save' and 'Cancel' buttons are at the bottom right.

### IP Address and Port (Reliable Syslog)

Select **IP Address and Port** for the **Event output type**, and then specify the following values:

- **Remote system:** Name of the system used as the reliable syslog server.
- **Port:** The default reliable syslog port is **601** for both TCP and UDP.
- **Protocol:** Choose between **TCP/IP** or **UDP**.



The screenshot shows the 'Event Sink Settings' dialog with the 'Output' tab selected. The 'Event output type' is set to 'IP Address and Port'. The 'Remote system' section has a 'Server' field with a red error message 'Field is required'. The 'Port' is set to '601' and the 'Protocol' is 'TCP/IP'. The 'Output message structure' is 'Default (xml)', 'Output text encoding' is 'UTF-16', and 'Output data format' is 'None (raw xml)'. 'Save' and 'Cancel' buttons are at the bottom right.

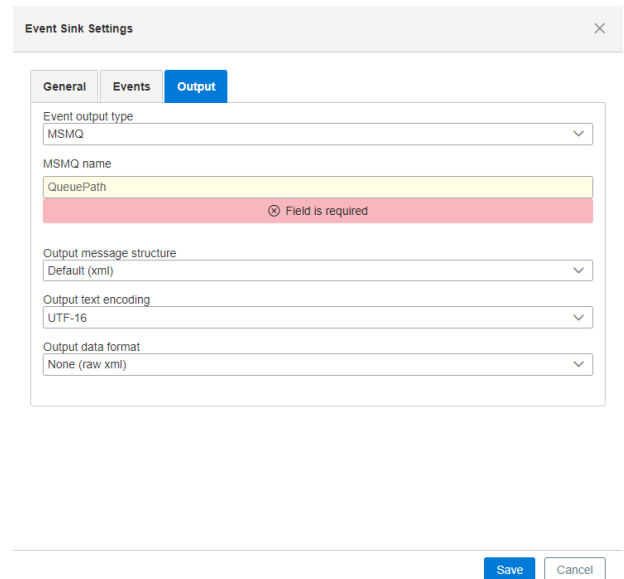


**i** For more information about message structure, text encoding, and data format of the output, please see "[Event Data Message Format](#)" on page 566.

## Configure MSMQ Event Output Type

The Microsoft Message Queue (**MSMQ**) event sink outputs the event information to a target server in a Microsoft Message Queue compatible format.

Select **MSMQ** for the **Event output type**, and then specify the name and queue of the MSMQ. Proper format is: **ServerName\QueueName**.



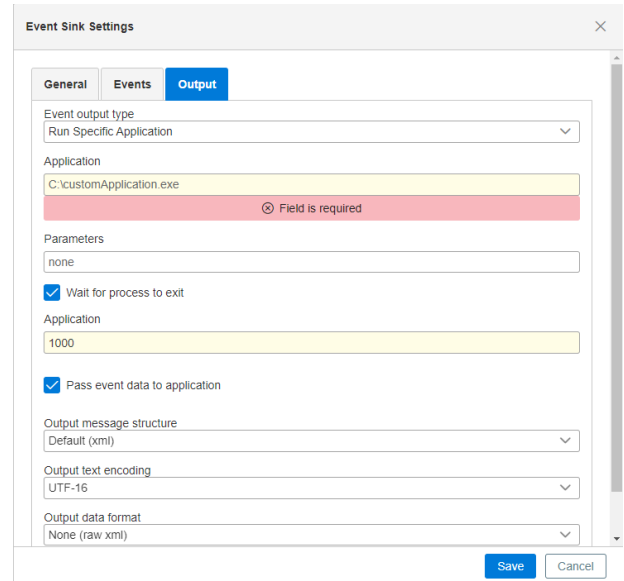
The screenshot shows the 'Event Sink Settings' dialog box with the 'Output' tab selected. The 'Event output type' is set to 'MSMQ'. The 'MSMQ name' field is highlighted in yellow and contains the text 'QueuePath', with a red error message below it stating 'Field is required'. The 'Output message structure' is set to 'Default (xml)', 'Output text encoding' is 'UTF-16', and 'Output data format' is 'None (raw xml)'. At the bottom right, there are 'Save' and 'Cancel' buttons.

**i** For more information about message structure, text encoding, and data format of the output, please see "[Event Data Message Format](#)" on page 566.

## Configure Run Specific Application Event Output Type

Select the **Event output type** as **Run Specific Application**, and then specify the following:

- **Application:** Specify the path to the application to run. The path is a local path relative to the system processing the event sink. For example, if the web server is to process the event sink, provide a local path on the web server.
- **Parameters:** The parameters are any additional parameters that would be provided following the executable name, such as additional paths, passwords, etc.
- **Wait for process to exit:** Causes the Privileged Identity to wait for a return code from the application.
- **Pass event data to application:** The event sink data is passed to the specified process for further action as **stdin**. The event being run receives the raw event data stream.



As a test of passing the event data to the process, set the following parameters:

- Application to run: **CMD**
- Parameters: **/C more > c:\text.txt**.

This runs the **more** command and pipe the event sink output data stream to the specified output file of **c:\text.txt** (formatted for this manual).

```
<Event CompactMode="1"
sEventType="OpResult"
dwBasicEventType="8"
dwAppSpecificEventID="2035"
sEventID="EVENT_ID_PASSWORD_VAULT_OPENED"
sOriginatingApplicationName
="Privileged Identity Console"
sOriginatingApplicationComponent=""
sOriginatingApplicationVersion="5.5.2.1" sOrigina
tingSystem="LSDSLSCPRD"
sOriginatingAccount="lsds\lscadmin"
dtPostTime="2017-03-05T10:35:43"
sMessage="Privileged Identity Console (running as user lsds\lscadmin) on system LSDSLSCPRD;
opened the password vault."/>
```



For more information on the message structure, text encoding, and data format of the output, please see ["Event Data Message Format"](#) on page 566.

## Configure Pre-Built Ticketing and Logging Event Output Types

Privileged Identity ships with the following prebuilt integrations for ticketing systems, which you may select from the **Event output type** list to configure. When the triggers configured in the event sink occur, a ticket is created within the configured incident system.

- **BMC Remedy (COM)**
- **HP Service Manager (COM)**
- **Jira (COM)**
- **OTRS (COM)**
- **CA Service Desk (COM)**
- **ServiceNow (COM)**
- **System Center (COM)**

Privileged Identity ships with following prebuilt integrations for logging systems, which you may select from the **Event output type** list to configure. Syslog output types simply write the event message in its entirety to a syslog server. These messages are written to the default messages log.

- **Syslog to ArcSight (CEF)**: Select this if the target syslog server is ArcSight. This preconfigures the output format to syslog CEF and cannot be changed. The default syslog port is **514** and sends via UDP.
- **Sylog to QRadar (LEEF)**: Select this if the target syslog server is QRadar. This preconfigures the output format to syslog LEEF and cannot be changed. The default syslog port is **514** and sends via UDP.



*For more information on the message structure, text encoding, and data format of the output, please see "[Event Data Message Format](#)" on page 566.*

# Configure Propagation Types in the Web Application

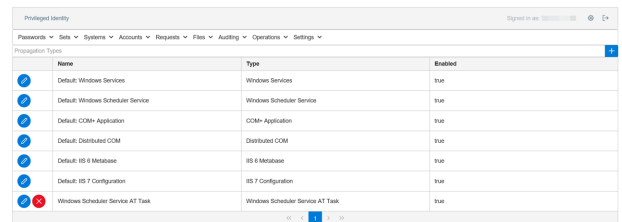
Propagation configuration settings define how Privileged Identity searches for credential references, and updates those references when the source credential is changed. For example, if you're using an account in a configuration file to connect to a database as part of an application running, when Privileged Identity updates the account password, it updates the password within that configuration file automatically as part of the password update.

To view, modify, and add propagation types in the web application, click **Settings > Propagation** from the top menu.

Default and user-defined propagation types are listed.

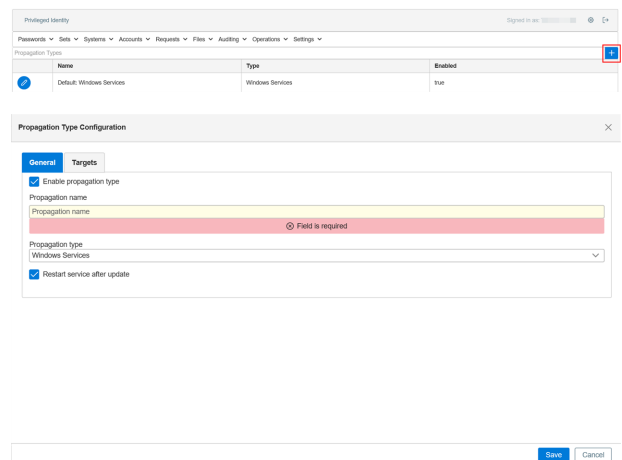
You can enable and disable the default propagation types and modify its targets but you cannot delete the default propagation types. Click the **Edit** (pencil icon) button for the default propagation type to enable or disable it or change its targets.

You can modify or delete (as indicated by the red **X**) user-defined propagation types.



## Add New Propagation Type

1. Click the **+** button above the list of propagation types.
2. Enter a name for the propagation.
3. Select a **Propagation Type** from the list. The following types can be added:



- **Windows Services:** Affects the identity used to logon for Windows services. Default has no configurations. Windows Services attempts to identify if each Windows service is clustered; if so, various cluster APIs are leveraged. If the service is not clustered, the service control manager (SCM) APIs are leveraged. Each Windows service is stopped and evaluated for dependencies. If any dependent services are found, they are also stopped. Each Windows service and any dependant service is then restarted in the correct order. User-added propagation allows configuration of service auto-restart functionality.
- **Windows Scheduler Task:** Affects credentials used to run Windows Scheduled tasks. There are no configuration options for this propagation type. Use **Settings > Program Options > Performance** in the management console to enable a timeout case. The timeout case uses a performance check to determine how long it will take to enumerate the tasks on a target system and if that timeout will be exceeded. If the timeout would be exceeded, this operation is skipped.
- **Windows Scheduler AT Service Task:** Affects the identity used for AT tasks (deprecated after Windows Server 2012). There are no configuration options for this propagation type.
- **COM + Application:** Affects COM Application Identities. There are no configuration options for this propagation type.
- **Distributed COM:** Affects DCOM application RunAs identities. There are no configuration options for this propagation type. Use **Settings > Program Options > Performance** in the management console to enable a timeout case. The timeout case uses a performance check to determine how long it will take to enumerate the DCOM applications on a target system and if that timeout will be exceeded. If the timeout would be exceeded, this operation is skipped.

- **IIS6 Metabase:** For IIS6 (Server 2003), affects users configured to run application pools, configured as the anonymous account for a web site or virtual directory, and the account configured for network access. There are no configuration options for this propagation type. If a target system is detected as Windows Server 2008 or later, this operation is skipped.
- **IIS7 Configuration:** For IIS7 and later (Server 2008 and later), affects users configured to run application pools, configured as the anonymous account for a web site or virtual directory, and the account configured for network access. There are no configuration options for this propagation type. If a target system is detected as Windows Server 2003 or earlier, this operation is skipped.
- **Run Process:** Configure the path to a program or script to run (on the host system or target system), configuration of that program runs on the host or target system and who the program runs as and if any files need to be copied to the target system (Windows only).
- **.NET Configuration Files:** List accounts configured in the Connection Strings component of ASP.NET under IIS 7 or later. This searches for the following elements: User, UserID, and UID. There are no configuration options for this propagation type.
- **System Center Operations Manager:** Affects credentials configured as RunAs identities within Microsoft System Center Operations Manager (SCOM). Use of this propagation requires copying the correct SCOM SDK binary files (all files from the SDK binaries directory of the SCOM host to the installation directory of the management console and/or zone processor. There are no configuration options for this propagation type.
- **File Search and Replace:** Configure the path to a file that is parsed for password replacement using the proper RegEx expression.
- **SharePoint:** There are no configuration options for this propagation type. Use **Settings > Program Options > General** to specify the SharePoint admin port that should be used for management. If you are attempting to manage multiple SharePoint farms, all need to be on the same port. This propagation element deploys a temporary service to the target machine to perform the propagation which self-terminates and removes itself.
- **IBM WebSphere Application Server:** Configure the use of SSL and the target port. This propagation type is typically not used as the functionality has been supplanted by directly managing the WebSphere server using the IBM WebSphere node.
- **Oracle WebLogic Server:** Configure the use of SSL and the target port. This propagation type is typically not used as the functionality has been supplanted by directly managing the WebLogic server using the Oracle WebLogic node.
- **SAP Server:** Configure the use of the Netweaver gateway or direct connect and related information. This propagation type is typically not used as the functionality has been supplanted by directly managing the SAP system using the SAP node.
- **SQL Server:** Lists accounts under the Credentials node as seen in SQL Management Studio. The target SQL database instances must be defined.
- **Windows Logon Cache:** Affects identities stored in the Windows Logon Cache (cached logons). There are no configuration options for this propagation type.
- **Windows Auto Logon:** Updates the registry setting for the configured auto-logon account on a Windows system. If the system is not configured to use auto-logon or if the configured auto-logon account is not the same as the account being updated, the propagation will have no effect.
- **SQL Server Reporting Services:** Affects the account for the specific SQL Reporting Services instance (SSRS). There are no configuration options for this propagation type.

4. Select the **Targets** tab and check the following options as applicable:

- **Run against Windows targets:** Enables this propagation type to be used when propagating to Windows targets.
- **Restrict to systems in a management set:** Restricts this propagation type to limit to a specific list of machines as found in a particular management set. Consider a service account used for windows services, tasks, in shell scripts, and also for SharePoint. By placing the SharePoint server in a management set designated for the SharePoint servers, as well as in the target management set for propagation, and restricting this propagation to the SharePoint servers management set, when the propagation runs against all of those systems, the SharePoint propagation will only be attempted against the SharePoint servers and not on any other system. If the restriction is not set, the SharePoint propagation would be

attempted against every system, including Linux machines, even if they don't actually run SharePoint. This can cause a job that could run in a few minutes to take considerably longer and could lead to other problems such as misleading failure notifications.

5. Click **Save**.

**i** For more information on configuring specific propagation types, please see "*Configure Propagation Settings*" on page 235.

## Configure Site Settings

Site settings is available to All Access users. This page lists all settings for the web application. Some settings can be specified and some cannot. Settings mentioned in this guide which are not visible in the web application are not enabled by your licensing.

### Version Information

This section lists the versions of the web application and web service components. No settings can be changed.

- **Web Page Version:** This version should match the COM Object, Web Service, and Web Service COM versions. This version number should also match the management console.
- **Com Object Version:** This version should match the Web Page, Web Service, and Web Service COM versions. This version number should also match the management console.
- **Web Service Version:** This version should match the COM Object, Web Service, and Web Page versions. This version number should also match the management console.
- **Web Service COM Object Version:** This version should match the COM Object, Web Page, and Web Service COM versions. This version number should also match the management console.
- **JQuery Version:** The JQuery Version. Contact BeyondTrust for more information.
- **JQueryUI Version:** The JQueryUI version. Contact BeyondTrust for more information.
- **Bootstrap Version:** The boot strap version. Contact BeyondTrust for more information.
- **DevExpress Version:** The DevExpress version. Contact BeyondTrust for more information.
- **TinyMCE Version:** The TinyMCE version. Contact BeyondTrust for more information.
- **jscolor Version:** The jsColorVersion. Contact BeyondTrust for more information.

### Database Settings

This section lists the database configuration for this web application. This should match the settings in your management console. No settings can be changed.

- **Database Server:** Name of the database server hosting the program database.
- **Provider Type:** SQL Server or Oracle.
- **Authentication mode:** Integrated (uses COM object credentials) or explicit (uses a defined database credential).
- **Database name:** Name of the database file on the database server, not present for Oracle.
- **SQL Account Name:** Integrated (uses COM object credentials) or the name of the explicit database account.
- **SQL Account Password:** Integrated (uses COM object credentials) or explicit if using an explicit database account.

### Email Configuration

The Email Settings are used to send email alerts when passwords are recovered or requested. These settings are controlled from the management console. If the email settings are updated, the web site(s) using the same Email Profile Name will automatically reflect the changes. If the changes to the mail profile are made from the management console or a different web application, the new settings will not be used until the COM application is restarted (will happen after several minutes of web site inactivity). Settings made at the web application will affect that web application immediately.

## Email Server Settings

- **Email Server Address:** Name of the mail server used to send emails.
- **Email Profile Name:** Default is Default. Multiple profiles can be created and manipulated for different scenarios.
- **Email Profile Description:** Default is Default Email Profile. This is a description of the email profile.
- **Organization:** The friendly name of the organization using this email profile.
- **From Name:** Name of the email sender.
- **From Address:** Email address this email should come from.
- **Reply to Address:** Email address any replies will be sent to.
- **Return Receipt Address:** Email address to send failed email send notifications to.
- **Read Receipt Address:** Email address to send "email read" notifications to.
- **Mail port:** The port used for the SMTP connection.
- **Server Timeout:** Timeout in seconds before SMTP gives up trying to send an email.
- **Custom Header Name:** The name of the custom mail MIME header. Note, this is not a subject name for the email.
- **Custom Header Value:** The custom MIME header code. Do not modify this code if you are not well versed in writing email MIME headers.
- **Message Priority:** The priority (High, low) for the message.
- **Message Sensitivity:** Helps define the nature of the content of the email message.
- **Message Importance:** Define how important this email message should be to the recipient.
- **Enable Log File:** Enable to turn on SMTP logging.
- **Log File Path:** If Enable Log File is enabled, define the physical path on the web server to log SMTP mail messages.
- **Enable Event Log Logging:** If enabled, will write SMTP log messages to the Windows application log of the host web server.

## Authentication Settings

- **Server Authentication Method:** Set the email server authentication method.
- **Server Requires Authentication:** Enable of the email server requires any form of user-based authentication to send email messages.
- **Authentication Username:** The username to authenticate to the email server with.
- **Authentication Password:** The password of the Authentication Username.

## Encryption Settings

- **Enable Email Signing:** Enables email signing for sent emails. Requires a signing certificate be configured.
- **Signing Cert Store Type:** Defines if the signing certificate comes from a Windows Certificate store or file.
- **Signing Cert Filename:** If Signing Cert Store Type is set to Signing File, identifies the path to the signing certificate.
- **Signing Cert Store Name:** If Signing Cert Store Type is set to Certificate, identifies the Windows Certificate Store name.
- **Signing Cert Password:** The password for the signing certificate, if required.
- **Signing Algorithm:** The signing algorithm to use.
- **Attach Signing Cert:** Attaches the signing certificate public key to outbound emails.



- **Enable Cached Signing Cert:** Loads the signing certificate into web application cache rather than calling from the store for each email signing attempt.
- **Enable Email Encryption:** Enables encrypted emails. This will require an encryption certificate.
- **Encryption Cert Store Type:** Defines if the encryption certificate comes from a Windows Certificate store or file.
- **Encryption Cert Filename:** If Encryption Cert Store Type is set to Encryption File, identifies the path to the encryption certificate.
- **Encryption Cert Store Name:** If Encryption Cert Store Type is set to Certificate, identifies the Windows Certificate Store name.
- **Encryption Cert Password:** The password for the encryption certificate, if required.
- **Encryption Algorithm:** The encryption algorithm to use.
- **Enable Cached Encryption Cert:** Loads the encryption certificate into web application cache rather than calling from the store for each email encryption attempt.
- **Enable Client SSL Cert:** Enables SMTP with SSL/TLS.
- **Client SSL Cert Store Type:** Defines if the SSL certificate comes from a Windows Certificate store or file.
- **SSL Client Cert Filename:** If SSL Cert Store Type is set to SSL File, identifies the path to the SSL certificate
- **SSL Client Cert Store Name:** If SSL Cert Store Type is set to SSL File, identifies the path to the SSL certificate.
- **SSL Client Cert Password:** The password for the SSL certificate, if required.
- **Enable Cached Client SSL Cert:** Loads the SSL certificate into web application cache rather than calling from the store for each email SMTP SSL/TLS attempt.

## OAuth Settings

- **OAuth Client ID:** The ID of the OAuth client that was assigned when the application was registered with the authorization server.
- **OAuth Client Secret:** The secret value for the client when the application was registered.
- **OAuth Server Auth URL:** The URL of the authorization server.
- **OAuth Server Token URL:** The URL used to obtain the access token.
- **OAuth Authorization Scope:** (Optional) The scope request or response parameter used during authorization. If the scope is not set, the authorization server will use the default access scope for your application as determined by the server. To request a specific access scope set this property to a space separated list of strings as defined by the authorization server.
- **OAuth Authorization:** Provide an authentication value if required by the authorization server.
- **Hide Browser Response:** When enabled, this will suppress the confirmation pop-up that indicates if authorization was successful.

## Password Recovery

Password Recovery Settings indicate what the constraints are to checkout a password and what will happen when a password is checked out.

- **Require Recovery Comment:** Enabled or disabled.
- **Require Check-in Comment:** Enabled or disabled.
- **Send Recovery Emails:** Enabled or disabled.
- **Send Recovery Emails To:** When passwords are recovered in the web interface, the alias will receive notifications for ALL password recoveries.
- **Auto-Roll Recovered Passwords:** Enabled or disabled. When a password is recovered that was set as a UNIQUE RANDOM password, the password may be set for re-randomization.

- **Password Check-Out Enabled:** Enabled or disabled. If enabled, users will be limited in time, and the number of passwords that they can have at any moment in time.
- **Check-Out Expiration Time:** Maximum amount of time in minutes a password can be checked out for.
- **Check-Out Extension Duration:** Initial amount of time in minutes a password will be checked out for. Also the amount of time granted per extension. This setting affects Windows systems only.
- **Max Simultaneous Check-Outs:** If password checkout is enabled, this is the maximum number of passwords an individual may check out at any moment in time.
- **Block Check-In if Account in Use:** Enabled or disabled. If enabled will cause the web site to attempt to see if the target account is logged into the system and if so will block the password from being checked back in.
- **Log if Account is in use to Event Log on System:** Enabled or disabled.
- **Log if Password Check-In to Event Log on System:** Enabled or disabled.
- **Log Password Check-Outs to Event Log on System:** Enabled or disabled.
- **Enable Personal Password Store-** enabled or disabled. If enabled, any user that can log in can store their own passwords in the password store.
- **Request Timeout:** Time in minutes a request for a password is valid for, after which it automatically expires.
- **Request Grant Timeout:** Time in minutes where once a request is granted, will the grant be valid for, after which it automatically expires.
- **Allow Users to Request Password for Future Check-out:** Enabled or disabled. Determines if a user can make a request for a password in the future in addition to immediately.
- **Time Window for Users to Request Passwords in the future:** The number of minutes in the future for which a user may request a password in the future.
- **Allow User to Check-Out Passwords to Groups:** Allows user to check out a password then make the same account/password available to members of another group that the original user also belongs to.
- **Force Search for Systems and Accounts:** Enabled or disabled. Determines if a user will be able to see a list of systems or will have to type in the name of the system and account they are attempting to recover.
- **Hide Information Columns for Systems and Accounts:** Enabled or disabled. Determines if information such as last status, IP address, etc, will be visible on the Systems view.
- **Block Password Checkout if Password Spin Fails:** Enabled or disabled. Determined if the password can be checked out of the parent job no longer exists.
- **Self Recovery:** Enabled or disabled. Enables or disables self recovery feature.
- **Phonetic Guide Locale:** Will vary based on Phonetics file chosen during web application setup.
- **Enable Phonetic Guide for Passwords:** Enabled or disabled.
- **Enable web site links for personal passwords:** Enabled or disabled.
- **Enable descriptions for personal passwords:** Enabled or disabled.

## Disconnected Account Management Settings

Disconnected account management settings are used to control the password policy for disconnected systems.

- **Enable Disconnected Account Management:** Enables the disconnected account management update feature.
- **Allow Automatic Tenant Creation:** Allows new tenants to automatically enroll. If a tenant ID does not exist and this option is enabled, when a new client checks in with a new tenant ID, a new tenant will be created with that ID.
- **Tenant Default Machine Max:** Maximum number of machines that a tenant can create.
- **Tenant Default Secret Update Frequency (in hours):** The frequency of the password change seed value change.

- **Tenant Default Derived Password Update Frequency (in hours):** The frequency of the password change.
- **Tenant Default Use Simple Derived Password (Hash):** When enabled, The first character will become a question mark and value containing uppercase, lower case, and numbers will be created. The length will be 31 characters. This password will be a derived value from the client secret.
- **Tenant Default Derived Password Length:** The length of the password to set.
- **Tenant Default Derived Password Allow Numbers:** Allows numbers to be used in the password.
- **Tenant Default Derived Password Allow Symbols:** Allows symbols to be used in the password.
- **Tenant Default Derived Password Allow Lowercase:** Allows lower case letters to be used in the password.
- **Tenant Default Derived Password Symbol Set:** Defines the symbols that can be used in a password if symbols are allowed.

## Ticketing Systems

Ticketing systems can be used for additional validation of the user prior to retrieving a password.

- **Require Ticket Number for Recovery:** Enabled or disabled.
- **Verify Ticket Number with Remedy:** Will attempt to verify the ticket number with Remedy.
- **Verify Ticket Number with SCSM:** Will attempt to verify the ticket number with Microsoft Service Manager.
- **Verify Ticket Number with HPSM:** Will attempt to verify the ticket number with HP Service Manager.
- **Verify Ticket Number with ServiceNow:** Will attempt to verify the ticket number with ServiceNow.
- **Verify Ticket Number with JIRA:** will attempt to verify the ticket number with JIRA.
- **Verify Ticket Number with OTRS:** Will attempt to verify the ticket number with OTRS.
- **Verify Ticket Number with CA Service Desk:** Will attempt to verify the ticket number with Service Desk.

## Account Elevation

- **Account Elevation:** Enabled or disabled.
- **Account Elevation Group:** The group (globally defined) that a user with **Elevate Account Access** can be added to on a target system.
- **Account Elevation to Domain Global Groups:** Enabled or disabled.
- **Account Elevation to Domain Global Group:** The group (globally defined) that a user with **Elevate Account Access** can be added to on a target system if the target system is a domain controller.
- **Local Account Elevation Timeout:** Time in minutes the user being elevated will remain in the target group.
- **Global Account Elevation Timeout:** Time in minutes the user being elevated will remain in the target group.
- **Enable Arbitrary Account Elevation:** Enabled or disabled.
- **Enable Elevation Expiration Emails:** Enabled or disabled.
- **Elevation Expiration Reminder Time:** Number of hours prior to automatic removal from a group that a user will be notified of that removal.
- **Require Comment for Self Account Elevation:** If enabled, the user must supply a comment when elevating their own account.
- **Default Short Term Account Elevation:** For arbitrary elevations, amount of time to elevate an account when "short" duration is selected.
- **Default Long Term Account Elevation:** For arbitrary elevations, amount of time to elevate an account when "long" duration is selected.
- **Maximum Account Elevation Time:** The maximum amount of time the user may be placed into an elevated group.

- **Elevation jobs are not owned by the creating user:** Enabled or disabled.
- **Hide job list in the elevation page:** Enabled or disabled.
- **Limit pre-selection list of elevation groups:** Enabled or disabled. This option does not affect permissions — only the visible options.
- **Require comment for self-service elevation:** Enabled or disabled.

## Security Settings

- **Session Timeout:** Time in minutes before login token expires.
- **Recalculate Session Permissions:** Time in minutes before a user's delegation rights will be recalculated. Set to higher if rights rarely if ever change. Controlled through the registry at HKLM\Software\Wow6432Node\Lieberman\PWCWebComponent\WebApplicationSettings\m\_iSessionRightsRefreshTimeoutMinutes.
- **Enable recursive group membership lookups:** Enabled or disabled. Determines if the web interface will recursively enumerate group memberships when a user tries to log on.
- **Disable Copy Button for Displayed Passwords:** Enabled or disabled. Determines if the copy button in the web interface is visible and functional when retrieving stored passwords.
- **Allow users to Edit Stored Random Passwords:** Enabled or disabled. For managed random passwords, determines if a user can edit the stored value for a password.
- **Hide Recovered Passwords After Timeout:** Enabled or disabled.
- **Display Time Before Passwords are Hidden:** Time in seconds before automatic navigation from password recovery page in web application.
- **Hide All Authenticator List In Login Screen:** If enabled, the login screen will not display an authenticators list. Users must be able to authenticate using their UPN name.
- **Strip Links to Non Local Resources:** If enabled, URLs will be presented as text rather than hyper-links.
- **Force RSA SecureID for Web Logins:** Enabled or disabled. Ignored if MFA is not enabled for the web site.
- **Enable RSA SecureID for Web Logins:** Enabled or disabled.
- **Enable Simple Usernames for RSA SecureID:** Enabled or disabled.
- **Enable OATH token authentication for web logins:** Enabled or disabled.
- **Allow Default Authenticated User Access:** Enabled or disabled.
- **Enable Integrated Windows Authentication:** Enabled or disabled.
- **Auto-Login Integrated Accounts:** Enabled or disabled. Ignored if Enable Integrated Windows Authentication is set to disabled.
- **Require Secure Cookies:** Enabled or disabled.
- **Limit Data Stored in Cookies:** Enabled or disabled.
- **Embed Unique Session Identifier within Page Requests:** Enabled or disabled.
- **Generate Unique String per Page Request:** Enabled or disabled.
- **Force Logout on Page Error:** Enabled or disabled.
- **Prevent Users From Granting Their Own Password Requests:** Enabled or disabled.
- **Allow Client Certificate Authentication:** Allows use of user-based certificates for authentication. Requires IIS also be configured to use SSL and accept or require client certificates.
- **Bypass Login Challenge for Client Certificate Identities:** If enabled and using certificate authentication, user will not be required to also enter a username and password when accessing the web site.
- **Enable Frequent Request Redirection:** If enabled, the web site will redirect the offending user to an alternate location to avoid service outage.

- **Frequent Request Redirection Threshold:** Is the number of requests per second coming from a specific source that will be tolerated before redirecting the user to an alternate location.
- **Enable Account Lockout:** If enabled, an account will be locked out of the web application for a period of time if a the threshold of failed login attempts in a defined time window is met.
- **Account Lockout Duration:** The number of minutes an account will be locked out if a the threshold of failed login attempts in a defined time window is met.
- **Account Lockout Reset Counter After:** The time window, in minutes, for which failed login attempts are tracked.
- **Account Lockout Threshold:** The number of failed attempts within the time window that an account may try to login unsuccessfully before being locked out.
- **Escape All Password Input:** Attempts to escape all password input to avoid certain types of hacking events.
- **Server Certificate URL:** The url for a client to download the certificate used by the web server.
- **Recording Server Certificate URL:** The url for a client to download the certificate used by the media server for recorded session playback.
- Enable VeriClouds Credential Testing will pass the user's login name, when specified as an email address, to your VeriCloud service provider to validate if the user's credentials have been found on the internet and potentially compromised. This service requires a separate support contract with Vericlouds. For more information go to <https://www.vericlouds.com/>. If the username (email address) and password are found by VeriClouds, a prompt will appear to the user indicating as such.
- **VeriClouds API Key:** input your VeriClouds API key.
- **VeriClouds API Secret:** input your Vericlouds API secret.
- **Prevent User Login If Password Is Known By VeriClouds:** When enabled, if the password the user supplies (when encrypted) matches the encrypted password value returned by VeriClouds, the user will not be allowed to login. The user will need to change their password to a new value that cannot be matched by the VeriClouds returned value.

## File Store

- **File Storing Enabled:** Enabled or disabled. It is enabled as part of the web site global configuration.
- **Alert on File Store Operations:** Enabled or disabled. Will generate an email to globally defined email address if enabled.
- **File Store Operation Alert Email:** The email address that file vault operation alerts will be sent to.
- **File Check-Out Enabled:** Enabled or disabled. If enabled, users will be limited in time, and the number of files that they can have at any moment in time.
- **File Check-Out Expiration Time:** Maximum amount of time in minutes a file can be checked out for.
- **File Check-Out Duration:** Initial amount of time in minutes a file will be checked out for. Also the amount of time granted per extension.
- **Max Simultaneous File Check-Outs:** If file checkout is enabled, this is the maximum number of files an individual may check out at any moment in time.
- **Log File Check-Outs to Event Log on System:** Disabled if set to None. Enabled if a particular system name is present.
- **Log File Check-Ins to Event Log on System:** Disabled if set to None. Enabled if a particular system name is present.
- **Files Encrypted in Store:** Enabled or disabled.
- **Add Permissions to Files For Group Memberships:** If enabled, the following permissions will may applied to all secondary identities associated with the user: View File, Request File, Change Delegation, Delete File.
  - Default Permission View File
  - Default Permission Request File
  - Default Permission Change Delegation
  - Default Permission Delete File

- **Limit File Store File Size:** Enabled or disabled.
- **File Store Size Limit:** If Limit Store File Size is enabled, this setting determines the maximum file size in kilobytes that Privileged Identity will permit for a file upload. Note-if IIS settings are more restrictive, IIS will limit the maximum file upload size.

## Sessions

Web Console SSH/Telnet Settings pertain to the remote sessions configuration for SSH and Telnet configurations. These settings may be overridden by user specific settings.

- **Enable RDP Connections:** If enabled, permits RDP connections to be made from the web application via the ActiveX control.
- **RDP Gateway Enabled:** If enabled allows defining an RDP gateway.
- **RDP Gateway Server:** Enabled and defined, is the RDP Gateway server that clients will be directed through when launching the ActiveX-based RDP control.
- **Allow RDP Connections To Any System:** Allows the user to specify the target system to connect to with the selected account when using the ActiveX-based RDP control.
- **Allow Multiple RDP Sessions:** Allows the client to initiate multiple, simultaneous RDP sessions when using the ActiveX-based RDP control.
- **Open RDP Sessions Maximized:** Will open ActiveX-based RDP sessions at Full Screen.
- **Generate And Download RDP Files:** Will download RDP files for the user to download and connect with rather than launching directly from the web site.
- **Enable SSH Console:** If disabled, the SSH java applet for Linux/Unix targets will not be displayed.
- **Allow SSH Connections To Any System:** Will permit the managed account to be used to connect to any system, if the target system permits it.
- **Allow Multiple SSH Sessions:** Will allow the launching of multiple SSH windows from the web site. If this box is disabled, the current telnet session will be disconnected before the new session is established.
- **Enable Key Timing Noise:** Enable to create a random timing offset for key transfer (security).
- **Enable X11 Forwarding:** If X11 forwarding is enabled on the target host, this will enable the feature to function in the Java-based SSH session.
- Use Thick Terminal Services Client
- **Allow New Server:** Enable to permit jumping from server to server from within the SSH session.
- **Enable Telnet Console:** If disabled, the Telnet java applet for Linux/Unix targets will not be displayed.
- **Allow Telnet To Any Systems:** Will permit the managed account to be used to connect to any system, if the target system permits it.
- **Allow Multiple Telnet Sessions:** Will allow the launching of multiple telnet windows from the web site. If this box is disabled, the current telnet session will be disconnected before the new session is established.
- **SSH Proxy Host:** Disabled if set to None. Enabled if an SSH Proxy Host name is present.
- Private Key Passphrase
- **SSH Compatibility Type:** When set to Auto, the control will determine what the target supports and use that. Force a particular version if desired.
- **SSH Port:** The SSH target port
- **SSH Connection Timeout:** Initial connection timeout.
- **SSH Handshake Timeout:** Amount of time for the connection handshake to take place
- **SSH Key Exchange Timeout:** Amount of time for the key exchange to take place



- **SSH Proxy Type:** Both the SOCKS and HTTP proxy protocols can be used to traverse firewalls. SOCKS is usually used to create a raw TCP connection, and the HTTP proxy protocol can do the same with the CONNECT method. If a proxy is required, also supply the SSH Proxy Host, SSH Proxy Port, and SSH Proxy Timeout.
  - **SSH Proxy Host:** The name/IP of the SSH proxy host that must first be connected to for an SSH connection.
  - **SSH Proxy Passphrase:** The passphrase to use the SSH Proxy Host.
  - **SSH Proxy Port:** The target SSH port for the SSH connection when using an SSH Proxy Host.
  - **SSH Proxy Timeout:** The timeout for the SSH proxy connection in seconds.
- **SSH Compression Level:** 0 (for none) or the level of compression.
- **Enable Private Key:** If SSH keys are configured in the solution, the Java-based SSH sessions may leverage keys to connect to the target systems.
- **Default Client Key Path:** Default location to look for keys on client systems if SSH keys are allowed for remote connections using a user's set of keys.
- **Allow Client Key Paths:** Enable to allow the user to identify the public key path on their own system rather than relying on the globally configured option.
- **Hide Advanced Settings for Launch App:** If enabled, advanced application launcher settings will be unavailable to the user.

## Languages

The languages section identifies all language files currently available to Privileged Identity.

## Other Settings

These are global settings that apply to all traffic with the web application.

- **Delay Page Load Data:** If enabled, begins loading pages as soon as any information is available rather than waiting for all information to be available.
- **Load All Password Options When Page Loads:** If disabled, all possible actions that can be taken against an account or system will be displayed to the user when the page is load. Disabling this setting can adversely affect performance for users who are not granted all access.
- **Items Per Page:** The number of rows returned on each page load. More rows means slower loading, but less paging. Users can override the global value by choosing Settings > User Session Settings, and setting a custom Page Size value in the User Settings section.
- **Auto-Login Integrated Accounts:** To be used in conjunction with Enable Integrated Windows Authentication from Security Settings, will automatically log a user into the web application when the user may use integrated Windows Authentication.
- **Privileged User Management Integration Enabled:** Enabled or disabled.
- **Privileged User Management Service Address:** The name of the PUM server.
- **Privileged User Management Access Account:** The name of the account that will be used to access the PUM server.
- **Privileged User Management Configuration File:** The name and path of the configuration file used to successfully communicate with the PUM server.
- **Enable Session Recording Integration:** Enable the ObservelT integration in the web application to playback and search through ObservelT recorded sessions.
- **Session Recording Application Address:** The address of the ObservelT integration page.
- **Session Playback Address:** The web URL for streaming media services used by the application launcher with built-in session recording.
- **Default Site Theme:** The default theme of the site. Users may override this setting with their own theme.

- **Enable User Dashboards:** Enables user dashboards.
- **Maximum Number of Rows to Export in Reports:** Determines the maximum number of rows to return in a report. More rows means larger files.
- **Default Page After Successful Login:** The page to load after a user successfully logs into the web site.
- **Web Service WSDL Endpoint URI:** The full URL to the SOAP-based web service.
- **Web Service REST Endpoint URI:** The full URL to the REST-based web service.



# Audits and Alerts

## Charts

Privileged Identity provides a number of charts that will display a variety of information about password age, account usage, and more. The charts are available to anyone granted **All Access** or **View Dashboards** permissions.

Chart data can be viewed from the web application or programmatically.

## Retrieve Chart Data from the Web Application

To enable a chart, select **Settings > Session Settings** from the top menu. Under **Main Panel Configuration**, check the box for the chart you wish to add. You can select all charts to add at once if desired.

## Password Data

- **Account Usage Discovered Instances:** A pie chart that describes account usage discovery by identifying how many usages are discovered for various propagation types.
- **Account Usage Password Coverage:** A histogram that describes account usage and divides the information based on managed passwords vs unmanaged passwords.
- **Password Access Activity:** A logarithmic chart showing access to stored/managed passwords over time. Click on a bar to drill down to the auditing data.
- **Password Access Times:** A logarithmic pie chart showing access to stored/managed passwords over time indicating the periods of highest activity. Click on a section of the chart to drill down to the auditing data.
- **Password Age Histogram (Linux root):** A histogram showing the number of stored Linux/Unix root account passwords. Click on a bar to drill down the account page filtered for root accounts to show the managed password ages of the root accounts.
- **Password Age Histogram (Linux):** A histogram showing the number of stored Linux/Unix account passwords. Click on a bar to drill down the account page filtered for Linux/Unix accounts to show the managed password ages of the accounts.
- **Password Age Histogram (Oracle):** A histogram showing the number of stored Oracle accounts indicating when they were last managed. Click on a bar to drill down the account page filtered for Oracle accounts.
- **Password Age Histogram (SQL sa):** A histogram showing the number of stored MS SQL sa account passwords. Click on a bar to drill down the account page filtered for root accounts to show the managed password ages of the sa accounts.
- **Password Age Histogram (SQL):** A histogram showing the number of stored MS SQL account passwords. Click on a bar to drill down the account page filtered for MS SQL accounts to show the managed password ages of the accounts.
- **Password Age Histogram (Stored):** A histogram showing the number of stored account passwords and when they were last set. Click on a bar to drill down the accounts page.
- **Password Age Histogram (Windows Admin):** A histogram showing the number of stored Windows Administrator account passwords. Click on a bar to drill down the account page filtered for root accounts to show the managed password ages of the Windows Administrator accounts.
- **Password Age Histogram (Windows):** A histogram showing the number of stored Windows account passwords. Click on a bar to drill down the account page filtered for root accounts to show the managed password ages of the Windows accounts.
- **Password Ages:** A chart identifying the relative password ages of all stored passwords.
- **Password Usage Summary:** A histogram summarizing password access by method. Click on a bar to drill down to the auditing data.

## Activity Data

- **User Activity:** A graph showing user activity. Click on a bar to drill down to the auditing data.
- **User Login Times:** A pie chart indicating periods of highest login activity.
- **Web Activity:** This chart shows web activity by operation over time. Click on a bar to drill down to the auditing data.

## Performance Data

- **Web Performance:** This chart shows web application calls and response times. The App Operations Metrics database must be configured for this chart to show data.
- **Web Service Performance:** This chart shows web service calls and response times. The App Operations Metrics database must be configured for this chart to show data.

## Job Data

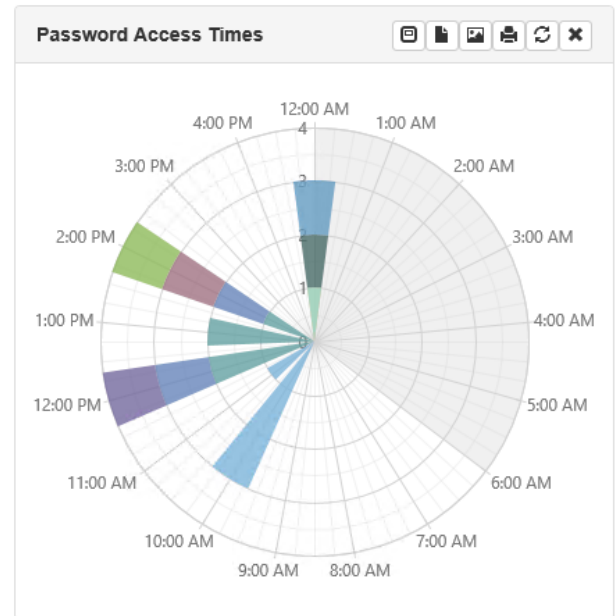
- **Job Account Elevations:** This chart indicates the number and types of account elevation jobs and job run durations.
- **Job Activity Summary:** This chart indicates the number and types of jobs and job run durations.
- **Job Completion Status:** This chart indicates the number of jobs that have ran and their success and failure rates.
- **Job Password Changes:** This chart indicates the number of password change jobs that have ran and their success and failure rates.

## Management Set Data

- **Management Set Target Count:** This chart shows management sets by size.
- **Management Set Target Types:** Indicates the number of size of managed targets by type.

All charts have options across the title bar.

- **Pop out:** Load the chart in a new tab to obtain a larger view.
- **Download...:** Download the chart data as a PDF or PNG file.
- **Print:** Print the chart.
- **Refresh:** Refresh the chart data.
- **Close:** Remove the panel from the user's session.



## Retrieve Chart Data Programmatically

Chart data is rendered via data feed from the web service. The web service is a prerequisite for installation and use of the web application.

To programmatically retrieve chart data use the REST interface: **/REST/Reports/ChartData**.

## Compliance Reports

There are a number of compliance reports built in to Privileged Identity designed to meet the most typical auditor-based requests regarding access to privileged accounts.

Compliance reports are available in both the management console and web application. Data capture jobs for compliance reports must be created in the management console.

The data in the compliance report database is not critical and serves no functional purpose beyond report generation. If this database should be lost or otherwise damaged, simply create a new compliance database and recreate new a new data capture.

### Create the Compliance Snapshot Database

1. From the **Actions** pane in the management console, click **Compliance**.
2. Click **Yes** to create the compliance database.
3. Specify the necessary database host settings and target database on the host. Specify a default schema (DBO is recommended).
4. Click **OK**.

### Update the Web Application to use the Compliance Snapshot Database

To make the data available to the web application, the web application must be updated.

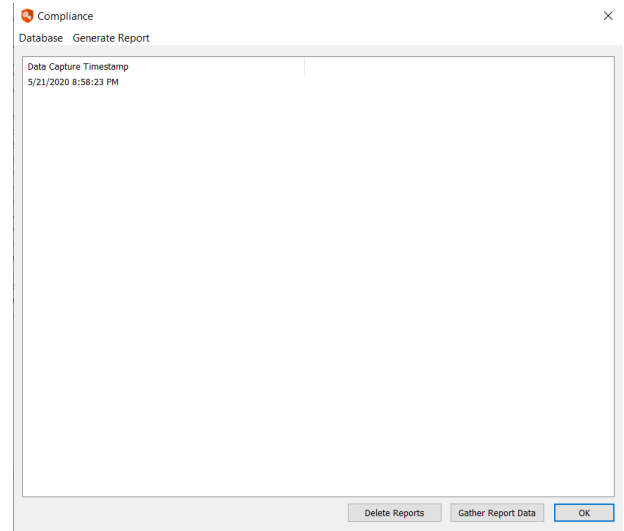
1. From the **Actions** pane in the management console, click **Manage Web App**.
2. Double click the target web instance to edit it.
3. Select **Yes** to overwrite web application settings.
4. Click **OK** to close the **Web Application Settings** dialog. This will cause the management console to re-push data store configuration which includes the compliance database configuration. The web server COM application will restart as part of this process.

### Create Compliance Report Data Capture Jobs

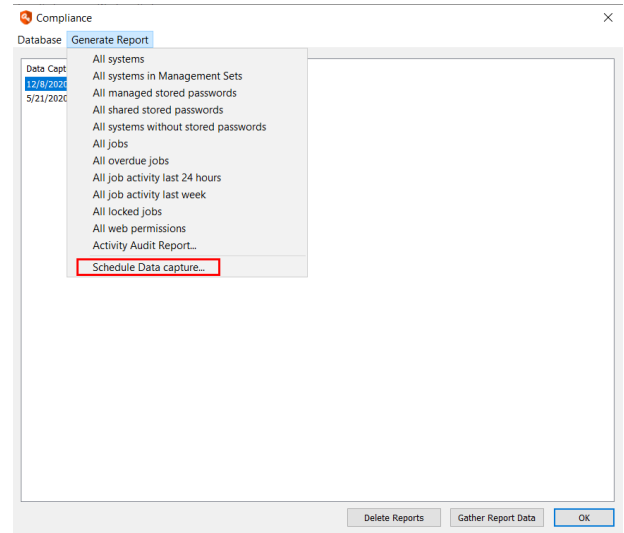
Once the compliance snapshot database has been created, create jobs to capture the reporting data.

Once the data capture occurs, the reports will be generated for the snapshot in multiple formats which will be compressed and stored in the program data store and made available through the web application.

1. From the **Actions** pane in the management console, click **Compliance**.
2. To capture the data now, click **Gather Report Data**.
3. To delete a snapshot, select it, and then click **Delete Reports**.

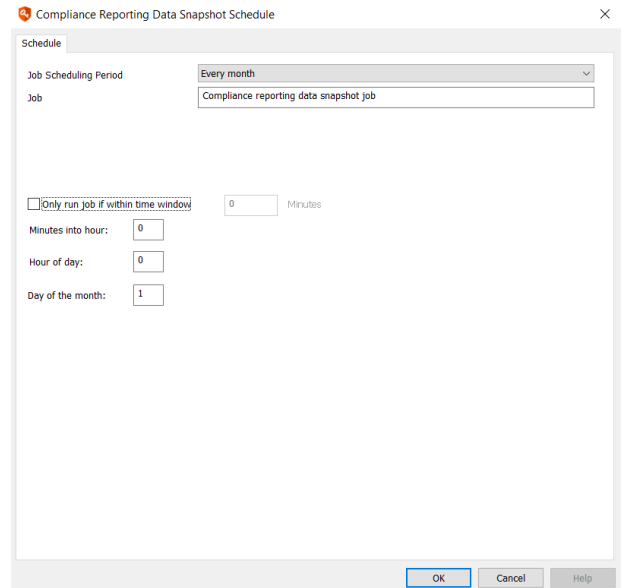


4. To schedule a data capture, select **Generate Report > Schedule Data Capture** from the menu.



5. Select the **Job Scheduling Period** and set a time frame to run the job, if desired, and then click **OK**.

**i** For more information on scheduling jobs, please see *"Set the Job Schedule"* on page 242.

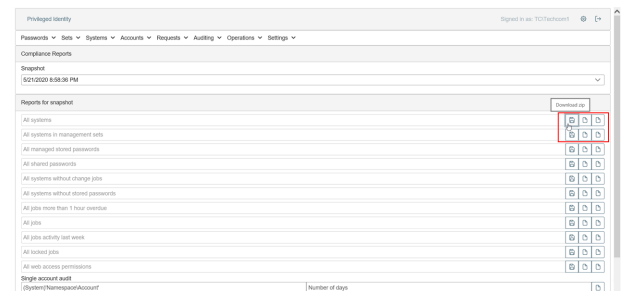


## View Compliance Reports

Compliance reports are available to users delegated **All Access** or **View Web Activity Logs** permissions.

1. Log into the web application.
2. Select **Auditing > Compliance Reports** from the menu.
3. Select a compliance report from the **Snapshot** list.
4. Click one of the download buttons for the report, according to which format you wish to download. Report formats include Zip, HTML, and CVS. Available reports include:

- **All Systems:** All systems in Privileged Identity and their last known status and other system information. Regardless of association with a management set.
- **All Systems in management sets:** All systems that are present in a management set.
- **All managed stored passwords:** Last managed time for every stored/managed password in the secure password store.
- **All shared passwords:** Last set time for credentials in shared credential lists.
- **All systems without change jobs:** All systems in Privileged Identity that are not associated with a password change job.
- **All systems without stored passwords:** All systems in Privileged Identity that have no stored passwords.
- **All jobs more than 1 hour overdue:** Lists jobs that are more than 1 hour past their scheduled run time.
- **All jobs:** List the job status for all jobs in Privileged Identity.
- **All jobs activity last week:** Lists all job status for jobs run in the last week.
- **All locked jobs:** Lists all jobs that are in a locked state. Locked jobs are either currently running or experiencing an issue and will not be run again without manual intervention.
- **All web access permissions:** Lists all global delegation permissions.



- **Single account audit:** Supply the name of a specific managed account to see all of the activity surrounding that account for a particular number of days. The expected name format is: **(SystemName)'NameSpace\AccountName'**.
- **Single system audit:** Supply the name of a specific system to see all of the activity surrounding that system for a particular number of days.
- **Single user audit:** Supply the name of an identity to see all activity performed by that identity for a particular number of days.



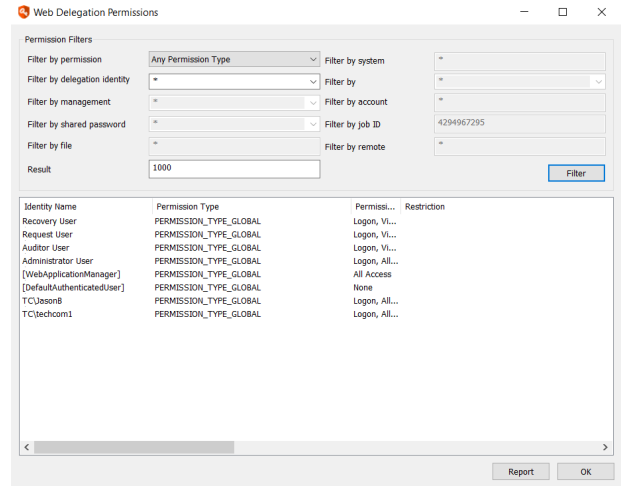
*For more information on available namespaces, please see "[Namespace Values](#)" on page 589.*

# Delegations Reporting

You can report on any and all permissions at any level.

## Create a Permissions Report

1. In the management console, select **Delegation > Delegation Permissions** from the top menu to open the **Web Delegation Permissions** dialog.
2. Choose the appropriate permission filter from the **Filter by permission** list.
3. The default selection is **Any Permission Type**. As necessary, filter by any of the other values.
4. Once the filter parameters are set, click the **Filter** button to view the result set.
5. Click the **Report** button to open the **Report Generator** dialog.



For more information on configuring the report generator dialog, please see **"Report Generator"** on page 533.

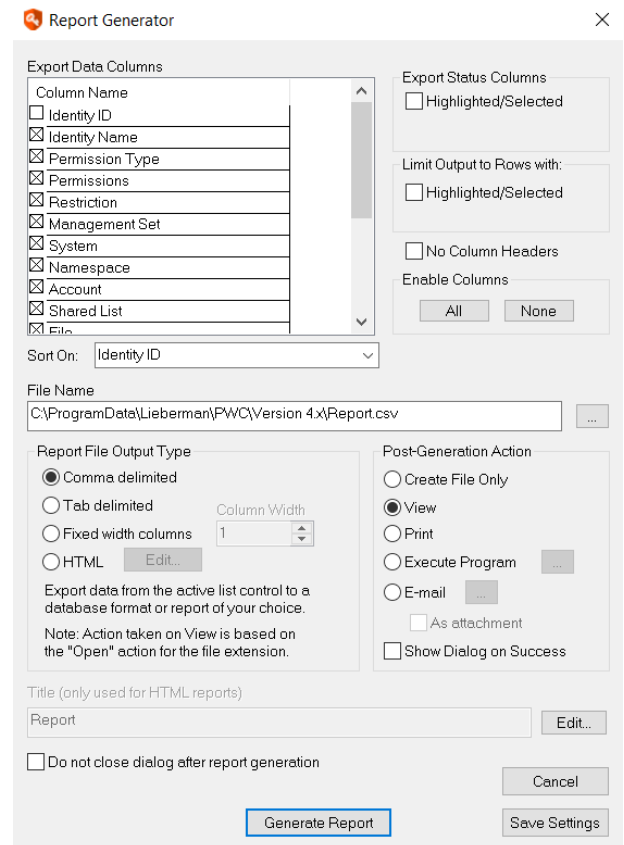


# Report Generator

In the management console, most dialogs have a report or generate report button. This will capture the contents of the current dialog and place it into a report document. The report may be CSV, HTML, tab, or fixed width delimited. There is no scheduling option for these types of reports.

The **Report Generator** dialog has the following settings:

- **Export Data Columns:** This is the list of columns present in the report. This list will change based on the dialog being reported on. Columns with an X in it will be included in the report.
- **Export Status Columns:** If enabled, the value will indicate if the object was highlighted when the report was generated.
- **Limit Output to Rows with:** If enabled, will limit the output to only the highlighted elements on the dialog.
- **No Column Headers:** If enabled, will not include the column names as the first row of the report.
- **Enable Columns:** Click **All** to enable all export data columns or **None** to turn off all data export columns.
- **Sort On:** Select the column name to sort the report on.
- **File Name:** The path the the reports location, including file name.
- **Report File Output Type:** The type of file to create. Large HTML reports will require a browser to open and are larger than their CSV or tab delimited counterparts.
  - For **HTML**, click **edit** next to the HTML report type to format the HTML report generation settings.
- **Post Generation Action:** Choose an action to perform once the report is generated.
  - **Create File Only** or **View** are the actions most typically used.
  - **Print** will require a printer be installed.
  - Select **Execute Program** to run a program after the report is generated.
  - Select **E-mail** to email the report to users.
    - If the **As Attachment** option is not selected, the body of the email will be the report.



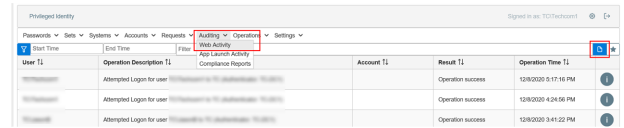
Click **Save Settings** to save your settings as the default settings.

Click **Generate Report** once you are satisfied with all of your settings.

# Audit Web Activity

## View Web Activity Using the Web Application

You can view the web activity logs from **Auditing > Web Activity** in the web application. All operations that occurred in the web application or web service, or delegation changes made in the management console that affect the web application are listed. **Web Activity** logs are available to users delegated **All Access** or **View Web Activity Logs** permissions.



You can sort the logs by clicking the desired column header and filter the logs by using the filters at the top of the page. You can save the filter by clicking the **Add Favorite** button next to the **Filter** box.

You can export the logs by clicking the **Export CSV** button next to the **Filter** box.

Web activity logs can also be viewed in the web application or programmatically.

## View Web Activity Logs Programmatically

Please see the programmers reference for more information.

- From Powershell, call `Get-LSListWebAuditLogs`.
- From SOAP, call `LoggingOps_GetWebAuditLog`.
- From REST, call `/REST/Logs/WebLogs`.

## Alerts and Integration Using Event Sinks

Privileged Identity features an extensible eventing system that allows an administrator to receive alerts or take action when events occur. Many actions can trigger messages that can be forwarded to other systems (for example, ticketing or SIEM systems) using the Event Sink system.

An event sink, sometimes called a *listener*, is a piece of code that defines how a server or computer is to handle given events. Event sinks are often used in spam filters to trigger actions in response to the receipt of an email message with defined characteristics or certain types of attachments. The destination of data handled by such a program is also sometimes called an event sink.

The purpose of the Event Sink system is to provide alerts and a framework for data feeds or integrations to third party systems. It was also designed with its own limitations in mind, and supports things like the arbitrary program invocation. In the case of Privileged Identity, actions such as password check-outs, password randomizations, failed propagations, and other events can trigger an event sink.

Each event sink is a registered listener that takes a specific action when one or more events occur. Each event sink can be configured to respond to a single event, a range of events, or multiple ranges of events. The action that an event sink takes is determined by the configuration of that specific event sink. Event sinks can be created, configured, and deleted from the management console or from the web application. The files are located in a directory specified by the application settings. By default, it is the program installation directory.

Once the event server is started, components of the application (console, web application COM object, deferred processors, zone processors, and so on) pass events to the event server to be processed asynchronously. When the event server receives an event, it checks the ranges of each registered event sink object to see if the event should trigger the sink. The event server triggers every event sink that is listening for a specific event.

Each event sink contains an output type. The output type determines what action is taken when an event is processed by the event sink. All events that match the event filter settings are sent to the specified output type.



*For more information on configuring event sink output types, please see "[Configure Event Sink Output](#)" on page 544.*

## Event Sink Events List

Event sink event IDs are grouped into sections based on the type of operation. Groups of operations each contain 1000 possible events, although not all 1000 event IDs are used in each range.



**Note:** Some event sinks, while visible and selectable from the UI, are not used or reported by the Privileged Identity. These event sinks are noted with the following prefix: -- **Not Available** --.

If an event sink has been replaced by another event sink, the new event sink is listed in the event sink description. Otherwise, the deprecated event sink is no longer functional.

Below is a list of event IDs with their corresponding descriptions.

### Unknown operations

- **0 - Event ID unknown:** Serves as a catch-all for when a condition is triggered for which there is no event ID data.

### Console/Interactive operations start at 1000.

- **1000 - EVENT\_ID\_PASSWORD\_RECOVERY\_MAIL\_ALERT** - Password was recovered and an alert email generated.
- **1001 - EVENT\_ID\_JOB\_FAILED\_TO\_LOCK** - Job lock request was issued, but job could not be locked; job will not run.
- **1002 - EVENT\_ID\_JOB\_RESET\_FOR\_RUN** - Job rescheduled to run again.
- **1003 - EVENT\_ID\_JOB\_CONTINUE\_PARTIAL\_RUN** - Multi-part job is only partially run; system set dynamic update configured to run on new systems.
- **1004 - EVENT\_ID\_JOB\_CANCELING\_RUN** - Job was canceled.
- **1005 - EVENT\_ID\_JOB\_STARTING\_TRUST\_UPDATE** - Determine domain trust relationships initiated.
- **1006 - EVENT\_ID\_JOB\_TRUST\_UPDATE\_OPERATION** - Determine domain trust relationships status complete.
- **1007 - EVENT\_ID\_JOB\_STARTING\_DYNAMIC\_GROUP\_UPDATE** - System set update initiated.
- **1008 - EVENT\_ID\_JOB\_DYNAMIC\_GROUP\_UPDATE\_OPERATION** - System set update status complete.
- **1009 - EVENT\_ID\_JOB\_STARTING\_ADMIN\_ACTIVITY\_REPORT** - Admin activity report initiated.
- **1010 - EVENT\_ID\_JOB\_ADMIN\_ACTIVITY\_REPORT\_OPERATION** - Admin activity report status complete.
- **1011 - EVENT\_ID\_JOB\_PASSWORD\_STATUS\_REPORT\_OPERATION** - Password verification report status complete. It is triggered after the report is successfully emailed.
- **1012 - EVENT\_ID\_SYSTEM\_RESTRICTED** - System was added to the global restricted systems list.
- **1013 - EVENT\_ID\_JOB\_LAUNCHING\_THREADS** - Job [id] launched a new thread.
- **1014 - EVENT\_ID\_JOB\_COULD\_NOT\_CONNECT\_TO\_SYSTEM** - Job [id] failed to connect to a system included in the job.
- **1015 - EVENT\_ID\_CONSOLE\_STARTED** - Management console was launched.
- **1016 - EVENT\_ID\_JOB\_COMPLIANCE\_DATABASE\_SNAPSHOT** - Compliance data snapshot was initiated from the management console.
- **1017 - EVENT\_ID\_JOB\_MISSED\_RUN\_RESCHEDULED** - Past due job was rescheduled to run ASAP (FIFO).
- **1018 - EVENT\_ID\_JOB\_MISSED\_RUN\_FINISHED** - Past due job finished running.
- **1019 - EVENT\_ID\_JOB\_ORPHANED\_LOCK\_CORRUPTED** - Fires when a locked job is found for which the original process that locked it no longer exists (for example, a deferred processor crash or a job prematurely stopped by an administrator).

- **1020 - EVENT\_ID\_JOB\_ORPHANED\_UNLOCKED** - When an orphaned job is found (per event sink 1019) the lock may be cleared automatically or by hand to provide further job processing. When the lock is removed, this event sink is triggered.
- **1021 - EVENT\_ID\_ENCRYPTION\_CHANGED** - Fires when a change to the encryption settings is made.
- **1022 - EVENT\_ID\_DATASTORE\_DATABASE\_DELETED** - This event is triggered when the management console is used to delete a database from the data store configuration dialog.

## Password operations start at 2000

- **2000 - EVENT\_ID\_PASSWORD\_ACCESS\_GRANTED** - Permission check for password recovery succeeds.
- **2001 - EVENT\_ID\_PASSWORD\_ACCESS\_REFUSED** - Permission check for password recovery fails.
- **2002 - EVENT\_ID\_PASSWORD\_CHECKED\_OUT** - Password recovery using RECOVERY rights was attempted.
- **2003 - EVENT\_ID\_PASSWORD\_CHECKED\_IN** - Password that was recovered has been checked in by the user.
- **2004 - EVENT\_ID\_PASSWORD\_CHECKOUT\_EXPIRED** - Recovered password checkout duration has expired.
- **2005 - EVENT\_ID\_PASSWORD\_RETRIEVED** - Password recovery was successful (Not by GRANT).
- **2006 - EVENT\_ID\_PASSWORD\_REQUESTED** - Password request was created.
- **2007 - EVENT\_ID\_PASSWORD\_REQUEST\_GRANTED** - Password request has been granted.
- **2008 - EVENT\_ID\_PASSWORD\_REQUEST\_DENIED** - Password request has been denied.
- **2009 - -- Not available -- EVENT\_ID\_PASSWORD\_RECOVERED\_FOR\_RDP** - RDP session has been initiated using the auto-RDP feature. This is now captured in event 3019.
- **2010 - EVENT\_ID\_JOB\_GENERATED\_RANDOM\_PASSWORD** - Random password was generated.
- **2011 - EVENT\_ID\_JOB\_STARTING\_PASSWORD\_STATUS\_REPORT** - Password Verification report initiated.
- **2012 - EVENT\_ID\_JOB\_FAILED\_PASSWORD\_STATUS\_CHECK\_FOR\_ACCOUNT** - Password verification failed for [ACCOUNT].
- **2013 - EVENT\_ID\_JOB\_STARTING\_PASSWORD\_CHANGE\_ON\_SYSTEM** - Password change job launched against [SYSTEM].
- **2014 - EVENT\_ID\_JOB\_FAILED\_LINUX\_PASSWORD\_UPDATE** - Linux password change job encountered a failure.
- **2015 - EVENT\_ID\_JOB\_SUCCESS\_LINUX\_PASSWORD\_UPDATE** - Linux password change job completed successfully.
- **2016 - EVENT\_ID\_JOB\_FAILED\_CISCO\_PASSWORD\_UPDATE** - Cisco password change job encountered a failure.
- **2017 - EVENT\_ID\_JOB\_SUCCESS\_CISCO\_PASSWORD\_UPDATE** - Cisco password change job completed successfully.
- **2018 - EVENT\_ID\_JOB\_FAILED\_MYSQL\_PASSWORD\_UPDATE** - MySQL password change job encountered a failure.
- **2019 - EVENT\_ID\_JOB\_SUCCESS\_MYSQL\_PASSWORD\_UPDATE** - MySQL password change job completed successfully.
- **2020 - EVENT\_ID\_JOB\_FAILED\_ORACLE\_PASSWORD\_UPDATE** - Oracle password change job encountered a failure.
- **2021 - EVENT\_ID\_JOB\_SUCCESS\_ORACLE\_PASSWORD\_UPDATE** - Oracle password change job completed successfully.
- **2022 - EVENT\_ID\_JOB\_FAILED\_WINDOWS\_PASSWORD\_UPDATE** - Windows password change job encountered a failure.
- **2023 - EVENT\_ID\_JOB\_SUCCESS\_WINDOWS\_PASSWORD\_UPDATE** - Windows password change job completed successfully.
- **2024 - EVENT\_ID\_JOB\_FAILED\_SQL\_PASSWORD\_UPDATE** - MS SQL password change job encountered a failure.
- **2025 - EVENT\_ID\_JOB\_SUCCESS\_SQL\_PASSWORD\_UPDATE** - MS SQL password change job completed successfully.
- **2026 - EVENT\_ID\_JOB\_FAILED\_AS400\_PASSWORD\_UPDATE** - AS400 password change job encountered a failure.
- **2027 - EVENT\_ID\_JOB\_SUCCESS\_AS400\_PASSWORD\_UPDATE** - AS400 password change job completed successfully.
- **2028 - EVENT\_ID\_JOB\_PROPAGATING\_TO\_SHAREPOINT\_SERVICES** - Propagation steps for Microsoft Office SharePoint Server.

- **2029** - -- Not available -- **EVENT\_ID\_JOB\_PROPAGATING\_TO\_TASKS** - Propagation steps for Windows tasks status.
- **2030** - -- Not available -- **EVENT\_ID\_JOB\_PROPAGATING\_TO\_COMPLUS** - Propagation steps for Windows COM status.
- **2031** - -- Not available -- **EVENT\_ID\_JOB\_PROPAGATING\_TO\_DCOM** - Propagation steps for Windows DCOM status.
- **2032** - -- Not available -- **EVENT\_ID\_JOB\_PROPAGATING\_TO\_IIS** - Propagation steps for Windows IIS status.
- **2033** - -- Not available -- **EVENT\_ID\_JOB\_PROPAGATING\_TO\_CUSTOM** - Propagation steps for custom propagation definitions status.
- **2034** - **EVENT\_ID\_JOB\_PROPAGATING** - Propagation steps for any propagation steps status.
- **2035** - **EVENT\_ID\_PASSWORD\_VAULT\_OPENED** - Stored password dialog was opened successfully from the management console.
- **2036** - **EVENT\_ID\_JOB\_FAILED\_CUSTOM\_ACCOUNT\_STORE\_PASSWORD\_UPDATE** - External password update failed.
- **2037** - **EVENT\_ID\_JOB\_SUCCESS\_CUSTOM\_ACCOUNT\_STORE\_PASSWORD\_UPDATE** - External password update succeeded.
- **2038** - **EVENT\_ID\_JOB\_STARTING\_ACCOUNT\_ELEVATION\_JOB** - Account elevation job initiated.
- **2039** - **EVENT\_ID\_JOB\_FAILED\_LDAP\_PASSWORD\_UPDATE** - LDAP directory account password update encountered a failure.
- **2040** - **EVENT\_ID\_JOB\_SUCCESS\_LDAP\_PASSWORD\_UPDATE** - LDAP directory account password update completed successfully.
- **2041** - **EVENT\_ID\_JOB\_FAILED\_SYBASE\_PASSWORD\_UPDATE** - Sybase password change job encountered a failure.
- **2042** - **EVENT\_ID\_JOB\_SUCCESS\_SYBASE\_PASSWORD\_UPDATE** - Sybase password change job completed successfully.
- **2043** - **EVENT\_ID\_PASSWORD\_RECOVERED\_BY\_GRANT** - Password recovery using REQUEST rights was attempted.
- **2044** - -- Not available -- **EVENT\_ID\_PASSWORD\_RECOVERED\_FOR\_TERMINAL\_SERVICES** - RDP session has been initiated using the auto-RDP feature.
- **2045** - -- Not available -- **EVENT\_ID\_PASSWORD\_RECOVERED\_BY\_CLIENT\_AGENT** - Password was recovered via SDK resources.
- **2046** - **EVENT\_ID\_JOB\_FAILED\_OS390\_PASSWORD\_UPDATE** - OS390 password change job encountered a failure.
- **2047** - **EVENT\_ID\_JOB\_SUCCESS\_OS390\_PASSWORD\_UPDATE** - OS390 password change job completed successfully.
- **2048** - **EVENT\_ID\_JOB\_DISCOVERY** - A discovery job was initiated.
- **2049** - **EVENT\_ID\_JOB\_FAILED\_IPMI\_PASSWORD\_UPDATE** - IPMI password change job encountered a failure.
- **2050** - **EVENT\_ID\_JOB\_SUCCESS\_IPMI\_PASSWORD\_UPDATE** - IPMI password change job completed successfully.
- **2051** - **EVENT\_ID\_JOB\_ACCOUNT\_ELEVATED** - An account was successfully elevated on the target system.
- **2052** - **EVENT\_ID\_JOB\_ACCOUNT\_ELEVATION\_FAILED** - An account failed to be elevated on the target system.
- **2053** - **EVENT\_ID\_JOB\_ACCOUNT\_ELEVATION\_DEELEVATED** - An account was successfully removed from the previous elevation.
- **2054** - **EVENT\_ID\_JOB\_ACCOUNT\_ELEVATION\_DEELEVATED\_FAILED** - An account failed to be removed from the previous elevation.
- **2055** - **EVENT\_ID\_JOB\_FAILED\_3270\_PASSWORD\_UPDATE** - TN3270 password change job success.
- **2056** - **EVENT\_ID\_JOB\_SUCCESS\_3270\_PASSWORD\_UPDATE** - TN3270 password change job failure.
- **2057** - **EVENT\_ID\_SHARED\_CREDENTIAL\_LIST\_ADDED\_ACCOUNT** - A new account password was added to a shared credential list.
- **2058** - **EVENT\_ID\_SHARED\_CREDENTIAL\_LIST\_EDITED\_ACCOUNT** - An account password was edited in a shared credential list.
- **2059** - **EVENT\_ID\_SHARED\_CREDENTIAL\_LIST\_REMOVED\_ACCOUNT** - An account was removed from a shared credential list.

- **2060 - EVENT\_ID\_SHARED\_CREDENTIAL\_LIST\_CHANGED\_PERMISSIONS** - Permissions were changed on a shared credential list.
- **2061 - EVENT\_ID\_SHARED\_CREDENTIAL\_LIST\_CREATED\_LIST** - A new shared credential was created.
- **2062 - EVENT\_ID\_SHARED\_CREDENTIAL\_LIST\_DELETED\_LIST** - An existing shared credential was deleted.
- **2063 - EVENT\_ID\_JOB\_STARTING\_SSH\_KEY\_UPDATE\_ON\_SYSTEM** - SSH key update job initiated.
- **2064 - EVENT\_ID\_JOB\_SUCCESS\_SSH\_KEY\_UPDATE** - The SSH key update succeeded.
- **2065 - EVENT\_ID\_JOB\_FAILED\_SSH\_KEY\_UPDATE** - The SSH key update failed.
- **2066 - EVENT\_ID\_JOB\_FAILED\_POSTGRESQL\_PASSWORD\_UPDATE** - PostgreSQL password update failed.
- **2067 - EVENT\_ID\_JOB\_SUCCESS\_POSTGRESQL\_PASSWORD\_UPDATE** - PostgreSQL password update succeeded.
- **2068 - EVENT\_ID\_JOB\_FAILED\_TERADATA\_PASSWORD\_UPDATE** - Teradata password update failed.
- **2069 - EVENT\_ID\_JOB\_SUCCESS\_TERADATA\_PASSWORD\_UPDATE** - Teradata password update succeeded.
- **2070 - EVENT\_ID\_JOB\_FAILED\_XEROX\_PHASER\_PASSWORD\_UPDATE** - Xerox Phaser password update failed.
- **2071 - EVENT\_ID\_JOB\_SUCCESS\_XEROX\_PHASER\_PASSWORD\_UPDATE** - Xerox Phaser password update succeeded.
- **2072 - EVENT\_ID\_JOB\_FAILED\_PASSWORD\_CHANGE\_ACCOUNT\_CHECKED\_OUT** - The password change job did not run because the account was currently checked out.

## Web application options start at 3000

- **3000 - -- Not available -- EVENT\_ID\_WEBAPP\_FAILED\_PERMISSIONS\_CHECK** - Login failed because account does not have permissions.
- **3001 - EVENT\_ID\_WEBAPP\_INVALID\_AUTH\_TOKEN** - Bad authentication token was passed for attempted login.
- **3002 - - - Not available, migrated to event ID 3010 and 3011 -- EVENT\_ID\_WEBAPP\_PERMISSION\_NOT\_GRANTED** - Valid login attempting to perform operation they don't have permissions for.
- **3003 - EVENT\_ID\_WEBAPP\_DATABASE\_CONNECTION\_FAILURE** - COM object failed to connect to the data store.
- **3004 - EVENT\_ID\_WEBAPP\_REMEDY\_VERIFICATION\_FAILED** - Ticket verification required as a web app option and verification failed
- **3005 - EVENT\_ID\_WEBAPP\_REMEDY\_LIBRARY\_LOAD\_FAILED** - The integration DLL did not get loaded.
- **3006 - EVENT\_ID\_WEBAPP\_REMEDY\_VERIFICATION\_SUCCESS** - Ticket verification required as a web app option and verification succeeded.
- **3007 - EVENT\_ID\_WEBAPP\_SCSM\_VERIFICATION\_FAILED** - Ticket verification required as a web app option and verification failed.
- **3008 - EVENT\_ID\_WEBAPP\_SCSM\_LIBRARY\_LOAD\_FAILED** - The integration DLL did not get loaded.
- **3009 - EVENT\_ID\_WEBAPP\_SCSM\_VERIFICATION\_SUCCESS** - Ticket verification required as a web app option and verification succeeded
- **3010 - EVENT\_ID\_WEBAPP\_PASSWORD\_RETRIEVAL\_FAILURE\_PERMISSIONS** - Password retrieval via client SDK failed.
- **3011 - EVENT\_ID\_WEBAPP\_PASSWORD\_RETRIEVAL\_SUCCESS** - Password retrieval via client SDK succeeded.
- **3012 - EVENT\_ID\_WEBAPP\_GENERIC\_ERROR** - web site encountered an error.
- **3013 - EVENT\_ID\_WEBAPP\_GENERIC\_MESSAGE** - web site encountered an event that caused logging.
- **3014 - EVENT\_ID\_WEBAPP\_PUM\_COMMAND\_SENT** - A command was sent to the PUM provider.
- **3015 - EVENT\_ID\_WEBAPP\_PUM\_COMMAND\_ERROR** - The command sent to the PUM provider generated a communications error.
- **3016 - EVENT\_ID\_WEBAPP\_LOGIN** - An identity logs into the web site.



- **3017 - EVENT\_ID\_WEBAPP\_GENERIC\_EVENT** - Covers generic events not covered by other specific events when there is no specific target account.
- **3018 - EVENT\_ID\_WEBAPP\_PASSWORD\_CHECKIN** - A password is checked in via the web interface.
- **3019 - EVENT\_ID\_WEBAPP\_GENERAL\_ACCOUNT\_OP** - Add a password, spin a password, enroll a system, or add an account via the web site. Also covers other general access denied errors not covered by a specific event ID where there is a specific target account.
- **3020 - EVENT\_ID\_WEBAPP\_LOGOUT** - An identity logs out of the web site.
- **3021 - EVENT\_ID\_WEBAPP\_HPSM\_LIBRARY\_LOAD\_FAILED** - The integration DLL did not get loaded.
- **3022 - EVENT\_ID\_WEBAPP\_HPSM\_VERIFICATION\_FAILED** - Ticket verification required as a web app option and verification failed.
- **3023 - EVENT\_ID\_WEBAPP\_HPSM\_VERIFICATION\_SUCCESS** - Ticket verification required as a web app option and verification succeeded.
- **3024 - EVENT\_ID\_WEBAPP\_PASSWORD\_CHANGE** - A stored password was changed directly from the web site.
- **3025 - EVENT\_ID\_WEBAPP\_SERVICENOW\_LIBRARY\_LOAD\_FAILED** - The integration DLL did not get loaded.
- **3026 - EVENT\_ID\_WEBAPP\_SERVICENOW\_VERIFICATION\_FAILED** - Ticket verification required as a web app option and verification failed.
- **3027 - EVENT\_ID\_WEBAPP\_SERVICENOW\_VERIFICATION\_SUCCESS** - Ticket verification required as a web app option and verification succeeded.
- **3028 - EVENT\_ID\_WEBAPP\_JIRA\_LIBRARY\_LOAD\_FAILED** - The integration DLL did not get loaded.
- **3029 - EVENT\_ID\_WEBAPP\_JIRA\_VERIFICATION\_FAILED** - Ticket verification required as a web app option and verification failed.
- **3030 - EVENT\_ID\_WEBAPP\_JIRA\_VERIFICATION\_SUCCESS** - Ticket verification required as a web app option and verification succeeded.
- **3031 - EVENT\_ID\_WEBAPP\_OTRS\_LIBRARY\_LOAD\_FAILED** - The integration DLL did not get loaded.
- **3032 - EVENT\_ID\_WEBAPP\_OTRS\_VERIFICATION\_FAILED** - Ticket verification required as a web app option and verification failed.
- **3033 - EVENT\_ID\_WEBAPP\_OTRS\_VERIFICATION\_SUCCESS** - Ticket verification required as a web app option and verification succeeded.
- **3034 - EVENT\_ID\_WEBAPP\_CASERVICEDESK\_LIBRARY\_LOAD\_FAILED** - The integration DLL did not get loaded.
- **3035 - EVENT\_ID\_WEBAPP\_CASERVICEDESK\_VERIFICATION\_FAILED** - Ticket verification required as a web app option and verification failed.
- **3036 - EVENT\_ID\_WEBAPP\_CASERVICEDESK\_VERIFICATION\_SUCCESS** - Ticket verification required as a web app option and verification succeeded.
- **3037 - EVENT\_ID\_WEBAPP\_CUSTOM\_USER\_EVENT** - Event generated by integrations using the Web Service interface. This event allows external programs to signal SIEM and trouble-ticket systems that listen to syslog-type events using Web Services for orchestration. Existing configured event sinks forward this event to all listeners configured to receive it. This event is generated by the SendEvent method.
- **3038 - EVENT\_ID\_WEBAPP\_KEY\_RETRIEVAL\_FAILURE** - A web operation using an SSH key failed to retrieve the specified key.
- **3039 - EVENT\_ID\_WEBAPP\_KEY\_RETRIEVAL\_SUCCESS** - A web operation using an SSH key successfully retrieved the specified key.



## File Store operations start at 4000

- **4000 - EVENT\_ID\_FILE\_RETRIEVAL\_REFUSED** - Web user failed to retrieve a file from the file store because the file is already checked out or the requesting user has already checked out the maximum number of files.

## Deferred and zone processor operations start at 5000

- **5000 - EVENT\_ID\_SCHEDULER\_STARTED** - Scheduling service started.
- **5001 - EVENT\_ID\_SCHEDULER\_PROCESSOR\_DISPATCH** - Scheduling service initiated a job run.
- **5002 - EVENT\_ID\_SCHEDULER\_PROCESSOR\_FINISHED** - Scheduling service finished running a job.
- **5003 - EVENT\_ID\_SCHEDULER\_FAILED\_TO\_RUN\_JOB** - Scheduling service encountered an error during a job run.
- **5004 - EVENT\_ID\_SCHEDULER\_FAILED\_LICENSING\_ERROR** - Scheduling service failed to run due to licensing issues.
- **5005 - EVENT\_ID\_SCHEDULER\_JOB\_COMPLETE\_ALERTS** - Job completed with not-critical errors.
- **5006 - -- Not available -- EVENT\_ID\_SCHEDULER\_JOB\_COMPLETE\_ALERTS\_FAILED** - Job completed with critical errors.
- **5007 - EVENT\_ID\_SCHEDULER\_STOPPED** - The scheduling service was stopped.
- **5008 - EVENT\_ID\_SCHEDULER\_JOB\_COMPLETE** - Job completed without errors.

## Audit Events start at 6000

- **6000 - EVENT\_ID\_AUDIT\_DELEGATION\_CHANGE** - web site delegations have been changed.
- **6001 - EVENT\_ID\_AUDIT\_MANAGEMENT\_SET\_CHANGE** - Management set properties have been changed.
- **6002 - EVENT\_ID\_AUDIT\_FULL\_PASSWORD\_STORE\_ACCESS** - Same as event ID 2035 with further information including SystemName and ProcessRunAsUsername (which is the account actually running the console at the time of access).

## Configure Event Sinks in the Management Console

The below instructions describe how to configure event sinks from the management console.

**i** For more information on configuring event sinks from the web application, please see *"Configure Event Sinks in the Web Application" on page 499.*

### Configure Event Sink Logging Options

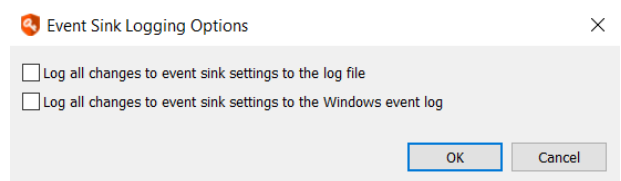
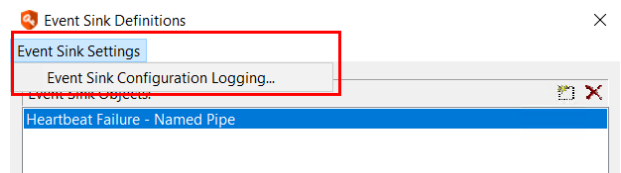
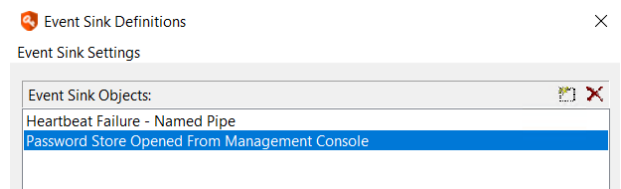
1. In the management console, select **Settings > Extension Components > Configure Event Sinks** from the top menu.
2. All event sinks are shown in the list. Each event sink specifies one or more ranges of events to listen for, an action to take, and an optional transformation that can be applied to the event information before it is handled. Event sinks are stored in the program data store and are available to all components.

**Note:** Currently there is no way to assign an event sink to a specific server. That means all components tied to the same database always attempt to use all configured event sinks.

3. From the **Event Sink Definitions** dialog, click **Event Sink Settings**, and then select **Event Sink Configuration Logging**.

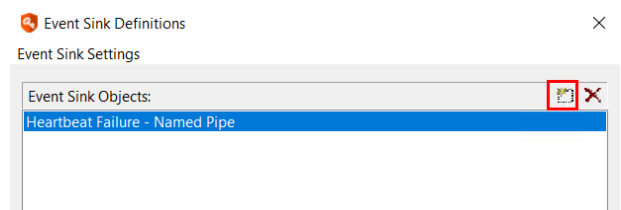
**Note:** Changes to event sinks are logged to the main Privileged Identity log file, as well as to the *PWCLog.txt* log file.

4. Select the desired logging options and click **OK**.



### Create an Event Sink

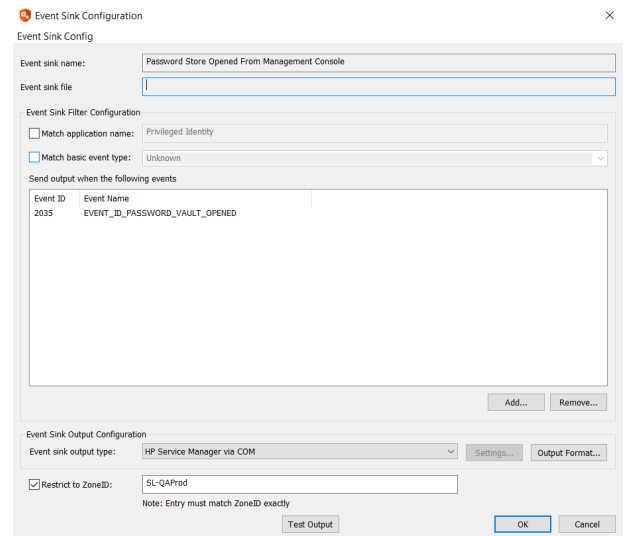
1. In the management console, select **Settings > Extension Components > Configure Event Sinks** from the top menu.
2. Click the **New** button in the top right corner of the **Event Sink Definitions** dialog.



3. Supply an **Event sink name**. This name cannot be changed once created.

4. Specify **Event Sink Filter Configuration**. We generally recommend using only the event sink list and not using the application name of event type filter.

- **Match Application Name:** Any event sink triggered by this application fires any event sink possible for this component. **Privileged Identity** is the only valid application name.
- **Match basic event type:** Triggers any event from a certain scope. Valid scopes are:
  - **Unknown:** There is no other predefined event sink.
  - **Generic Success:** Triggered by any success message.
  - **Generic Failure:** Triggered by any failure message.
  - **Trace:** Triggered by debug specific messages when debugging is enabled for the system.
  - **Job Processing Start:** Triggered when any job begins.
  - **Job Processing Status Update:** Triggered by any updates during a job run.
  - **Job Processing Complete, Success:** Triggers when any job successfully completes.
  - **Job Processing Complete, Failure:** Triggers when any job fails to complete successfully.
  - **Operation Notification, General:** Any operation that sends a notifications and that notification has a status message, including success and failure.
  - **Operation Notification, Success:** Any operation that sends notifications and that notification has a success message.
  - **Operation Notification, Failure:** Any operation that send s notifications and that notification has a failure message.
  - **Operation Status Update:** Any operation that has a status message.
  - **Operation Failure:** The operation failed.
  - **Application Component Status Update:** Any update messages from any component.
  - **Application Component Internal Error:** Any component used during a job encountered an error.
- **Send output when the following events:** Add one or more specific events by clicking **Add**.



The filters are cumulative, meaning the event sink will only process the event if it matches all of the filter settings. For example, if the event must match the Application Name and an event ID range is also provided, then the event must match the Application Name and must also fall into the event range.

If filtering for events that match the application name, only events generated by the application with the same name will be processed by the event sink. By default, the name of the application in the filter field will be the same as the name of the application which is creating the dialog. All the components of an application will use the same application name for the purposes of identifying events (web application, deferred processors, zone processors, etc).

If filtering for a specific event type, only events that match the type will be processed by the event sink. Event types include:

- **Success:** Indicates that an operation completed
- **Failure:** Indicates that an operation failed
- **Error:** Indicates that an error occurred during the course of operation
- **Debug Trace:** Indicates that the event was triggered as a debug diagnostic step

- **Status Update:** Indicates that the event was triggered to provide additional information during the course of a normal operation
- **Unknown:** Indicates an unknown type of event

**i** For more information on specific events, please see *"Event Sink Events List"* on page 536.

## Configure Event Sink Output

Once the items to monitor have been properly identified, you must specify an output type for the event. Use the bottom portion of the **Event Sink Configuration** dialog to configure output types.

The event sink listener configuration controls the action taken when an event is processed by the event sink. Currently, an event sink can do any of the following:

- **Log File:** Write the message to an output file. The only additional argument to specify here is the name of the file that you want to write the messages to. If the file does not exist, it will be automatically created when the first event is processed.
- **Update Registry Value:** Write the message to a value in the registry. Specify the system, base registry key, registry path, and registry value to write the event message. Specify an event notification name.
- **Named Pipe:** Write the message to a named pipe. Specify the name of the named pipe to write the event message data to. The resulting event message is sent as text over the pipe.
- **Com Call:** Call a function on a COM object and pass the message as an argument. Specify a COM Program ID and a COM method name on that interface. The resulting event message is passed as a BSTR argument to the function.
- **Send Email:** Send an email containing the message. Specify an SMTP email configuration profile as well as a semicolon delimited list of message recipients. By default, if the SMTP email settings for the application are configured already, then the SMTP configuration will exist with the configuration name **Default**.

If an alternate email configuration is required for the event sink, then create a different configuration in the registry by creating a new key under the location **HKLM\Software\Lieberman\SmtpSettings** and copy the values from default or change the values as desired. Emails can be sent to single users or to distribution lists using this method.

- **Windows Event Log:** Write the event to an application event log. Specify the target windows server, event type, and event ID.
- **Syslog:** Write the event to a syslog compatible system. Specify the target syslog server, and proper format as syslog format (no options), ArcSight CEF format, or QRadar LEEF format.
- **MSMQ:** Write the event to a Microsoft Message Queue. Specify the target MSMQ server and queue.
- **Run a specified application:** Run any local program on the application or web server. Define the path on the host web or application server and any required parameters.
- There are also a number of default integrations you can select from the list such as ArcSight, QRadar, BMC Remedy, Microsoft System Center Service Manager, ServiceNow, and others.

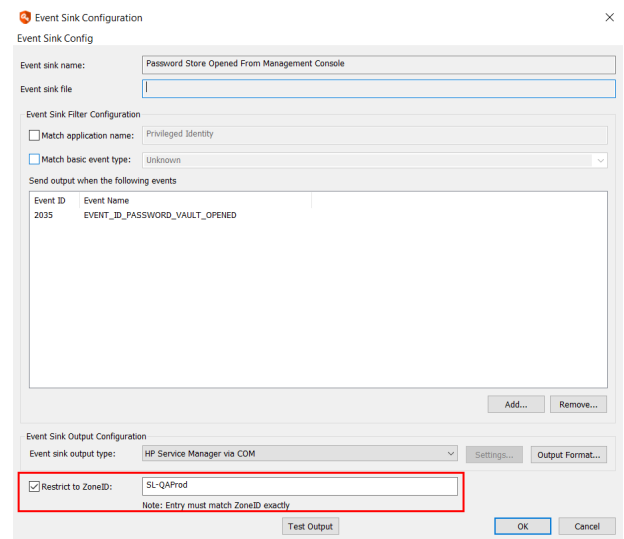
Each output type by default only receives the raw event message when an event is processed. Click **Output Format** to configure the output format settings for the event sink. Use a transformation to modify the format or information that is sent before it is sent to the output type. To use a transform, select **Custom Transform File** from the **Output data format** list, and then specify the path to the **Transform file**. The same transform file can be leveraged for multiple event sinks. A transform file can be created and modified directly from the dialog. The default transform editor used to open the transform files can be configured through the Transform Editor options. The default transform editor is Windows Notepad, located at the default location. The file type and format of the event messages using transform files can also be set. For example, to send HTML file email alerts to an email list, specify an HTML transform file with the correct formatting for an HTML email message, and the recipients will receive HTML encoded email messages in the same format as the transform file when the event corresponding to the event sink occurs.

**i** For more information on Transform file formatting and replaceable arguments, please see *"Event Sink Transform Files"* on page 568.

## Restrict Event Sink to a Specific Zone

In a case where an event sink is processed by a server that resides in an environment where not all Privileged Identity processors have access to that server, you may want to configure the event sink so that it processes by a specific zone processor in a zone where all integration components can access the server processing the event. This ensures the processor can reach the target to process the event and send the event output.

Use the **Restrict to ZoneID** setting to configure the event sink to use a specific zone processor.



Event Sink Configuration

Event Sink name: Password Store Opened From Management Console

Event sink file: [Empty field]

Event Sink Filter Configuration

Match application name: Privileged Identity

Match basic event type: Unknown

Send output when the following events

Event ID	Event Name
2035	EVENT_ID_PASSWORD_VAULT_OPENED

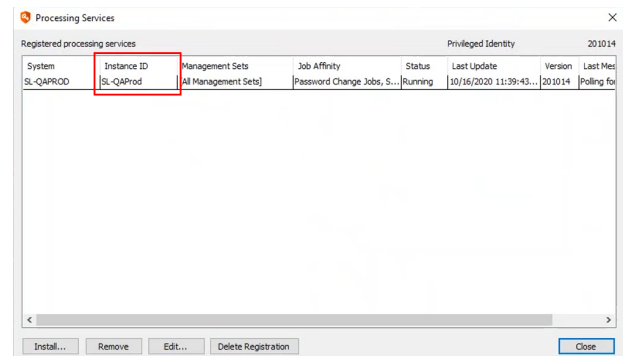
Event Sink Output Configuration

Event sink output type: HP Service Manager via COM

Restrict to ZoneID: SL-QAProd  
Note: Entry must match ZoneID exactly

Buttons: Add..., Remove..., Test Output, OK, Cancel

**Note:** The ZoneID entered must exactly match the Instance ID for the zone processor, as shown on the **Processing Services** window in the management console.



Processing Services

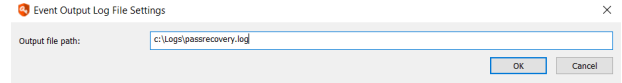
System	Instance ID	Management Sets	Job Affinity	Status	Last Update	Version	Last Met
SL-QAPROD	SL-QAProd	[All Management Sets]	Password Change Jobs, S...	Running	10/16/2020 11:39:43...	201014	Polling fo

Buttons: Install..., Remove, Edit..., Delete Registration, Close

## Configure Log File Event Output Type

The event sink log file outputs the event information to a flat text file at the chosen location. If the file does not exist the first time the event sink is triggered, it is created automatically.

To configure the **Log File** output type, click **Settings**, and then specify the path and the name of the log file to generate, for example, **c:\Logs\passrecovery.log**.



*For more information on the message structure, text encoding, and data format of the output, please see ["Event Data Message Format"](#) on page 566.*

## Configure Update Registry Value Event Output Type

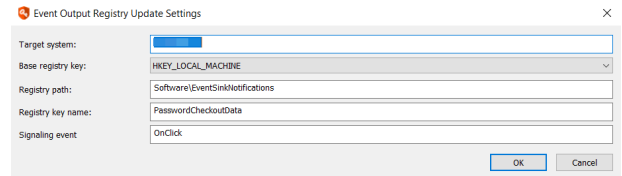
The event sink registry value outputs the event information to a registry path value on the target server. If the file registry value does not exist the first time the event sink is triggered, it is created automatically.

The process works as follows:

1. An event is configured to output to a target Windows host registry.
2. The sink then writes the event sink information to the target system at the registry location specified for the sink.
3. Pending successful write of the registry value, Privileged Identity sends a windows signaling event (e.g clicking a button in a dialog signals OnClick), so that if there is a separate listing application listening for that signaling event, that application can then take additional action. If there is nothing listening for the signaling event on the target system, you may leave the field blank or supply any random name. The OpenEvent function is called on the remote system followed by set event.

Configure the output type as **Update Registry Value** and specify the following values:

- **Target System Name:** Name of the machine which the registry write occurs. This requires access to the remote registry feature (remote registry service must be running). Input is the target server name.
- **Base Registry Key:** The hive key to write to such as, HKEY\_LOCAL\_MACHINE, HKEY\_USERS, or HKEY\_CLASSES\_ROOT.
- **Registry Path:** The path within the base key to write the registry values such as, Software\EventSinks. As Privileged Identity is currently a 32 bit application, The Software registry key is virtual. This means an input of Software\EventSinks is automatically written by the target Windows system as Software\WoW6432Node\EventSinks.
- **Registry Value Name:** Is a string value located within the registry path such as: PasswordCheckoutData.
- **Signaling Event Name:** Is the event name sent to the Windows eventing system (not the event logs).



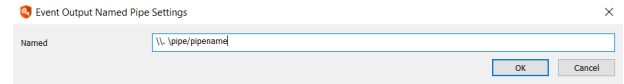
**i** For more information, please see the following:

- [OpenEventA function \(synchapi.h\) at https://msdn.microsoft.com/en-us/library/windows/desktop/ms684305\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/desktop/ms684305(v=vs.85).aspx)
- For information on the message structure, text encoding, and data format of the output, "[Event Data Message Format](#)" on page 566

## Configure Named Pipe Event Output Type

The **Named Pipe** event sink outputs the event information to a specified named pipe call.

Configure the output type as **Named Pipe** and specify its path.



**i** For more information on how to construct and use named pipes, please refer to Microsoft or other applicable vendor documentation.

**i** For more information on the message structure, text encoding, and data format of the output, please see "[Event Data Message Format](#)" on page 566.

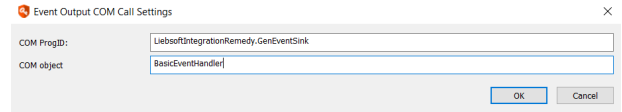


## Configure COM Call Event Output Type

The **COM Call** event sink outputs the event information to a specified COM object by calling a method specific to the COM object. The COM object can take any possible action such as feeding information to a help desk ticketing system or calling other methods or scripts.

Configure the output type as **COM Call** and specify the following values:

- **COM Prog ID:** The registered name of the COM object
- **COM Method Name:** The method or routine as specified by the COM object

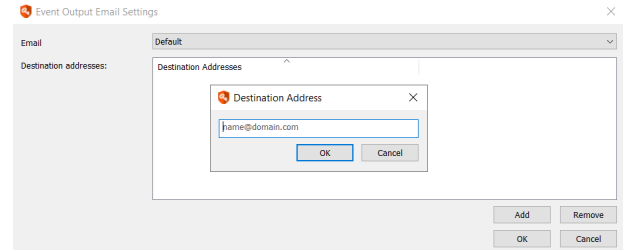


*For more information on the message structure, text encoding, and data format of the output, please see "[Event Data Message Format](#)" on page 566.*

## Configure Send Email Event Output Type

The **Send Email** event sink outputs the event information to the specified email recipients via an email.

An SMTP email configuration profile must be specified as at least one email recipient. By default, if SMTP email settings for the application have already been configured, the SMTP configuration is named **Default**.



Email output types simply write the event message in its entirety to the body of an email or an email attachment.

To format the data using a transform file, specify a transform file to use in the **Use Transform File** field. Transform files will format the data and included information within the log file.

For the email output, if a transform file is not used, the raw event information is attached to a blank email. A sample transformation file is shown here:

```
<html>
  <head>
    <title>Test</title>
  </head>
  <body>
    Email for password recovery from the web application
    %Message%
    Login Name %sLoginName%
    Checked out account: %ContextVariable:sSystemName%\%ContextVariable:sAccountName%
  </body>
</html>
```

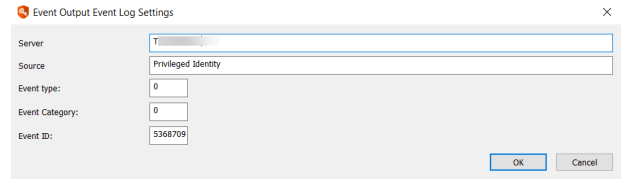


*For more information on the message structure, text encoding, and data format of the output, please see "[Event Data Message Format](#)" on page 566.*

## Configure Windows Event Log Event Output Type

The **Event Log** event sink outputs the event information in its entirety to the application log of the target Windows server.

Configure the output type as **Windows Event Log**, and click **Settings** to specify the following values:



- **Server:** Name of the machine to be used as the Windows event log server
- **Source:** The event source that will be written to the event log
- **Event Type:** Specifies if the type is information, warning, or failure. Event type 0, 1, 2 are for Information, Error, and Warning respectively.
- **Event Category:** Specifies the category of the event. Useful for filters in the event log. Categories are defined by the admin for the sake of information organization.
- **Event ID:** The event ID as it should appear in the event logs for this particular event sink.



*For more information on the message structure, text encoding, and data format of the output, please see "[Event Data Message Format](#)" on page 566.*

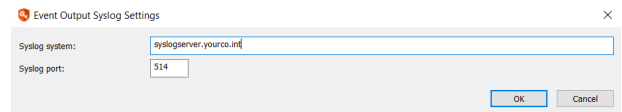
## Configure Syslog Compatible Event Output Types

Both **Syslog** and **IP Address and Port** event sinks output the event information to a target server in a syslog compatible format. The messages are written to the default messages log. **Syslog** uses the historical syslog service and **IP Address and Port** uses the reliable syslog service and functions over TCP or UDP.

### Syslog

Configure the output type as **syslog** and specify the following values:

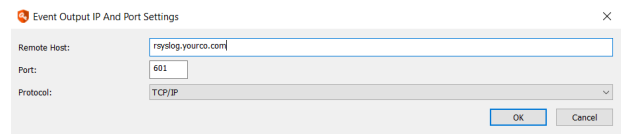
- **Syslog system:** Name of the server to be used as the syslog server.
- **Syslog port:** The default syslog port is **514** and sends via UDP.



### IP Address and Port (Reliable Syslog)

Configure the output type as **IP Address and Port** and specify the following values:

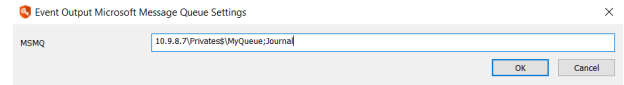
- **Remote host:** Name of the machine to be used as the reliable syslog server.
- **Port:** The default reliable syslog port is **601** for both TCP and UDP.
- **Protocol:** Choose between **TCP/IP** or **UDP**.




To configure the structure of the output and output data format to use a transform file, please see "[Event Data Message Format](#)" on page 566.

## Configure MSMQ Event Output Type

The **MSMQ** (Microsoft Message Queue) event sink outputs the event information to a target server in a Microsoft Message Queue compatible format.



Configure the output type as **MSMQ** and specify the name and queue of the MSMQ. Proper format is: ServerName\QueueName.



*For more information on the message structure, text encoding, and data format of the output, please see "[Event Data Message Format](#)" on page 566.*

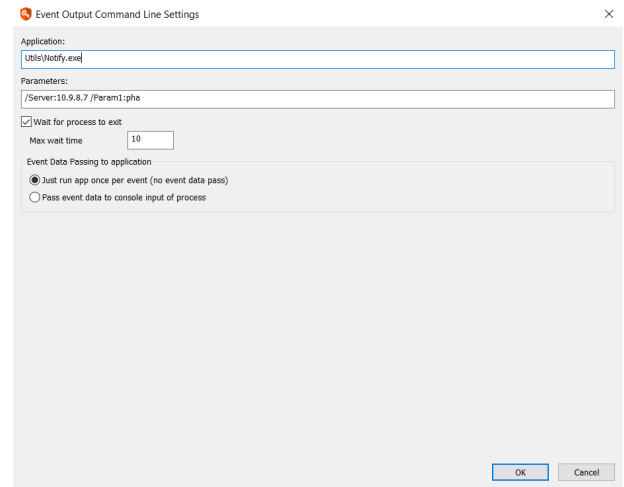
## Configure Run a Specified Application Output Type

The **Run a specified application** event sink runs any arbitrary process and the output can be configured for consumption by Privileged Identity.

Configure the output type as **Run a specified application**, and then click **Settings**.

Provide the following information:

- **Application:** Specify the path to the application to run. The path is a local path relative to the system processing the event sink. For example, if the web server is to process the event sink, provide a local path on the web server.
- **Parameters:** The parameters are any additional parameters that would be provided following the executable name, such as additional paths, passwords, etc.
- **Wait for process to exit:** Causes the Privileged Identity to wait for a return code from the application.
- **Just run app once per event:** Run the specified application once per event.
- **Pass event data to console input of process:** The event sink data is passed to the specified process for further action as **stdin**. The event being run receives the raw event data stream.



As a test of passing the event data to the process, set the following parameters:

- Application to run: **CMD**
- Parameters: **/C more > c:\text.txt**.

This runs the **more** command and pipe the event sink output data stream to the specified output file of **c:\text.txt** (formatted for this manual).

```
<Event CompactMode="1"
sEventType="OpResult"
dwBasicEventType="8"
dwAppSpecificEventID="2035"
sEventID="EVENT_ID_PASSWORD_VAULT_OPENED"
sOriginatingApplicationName
="Privileged Identity Console"
sOriginatingApplicationComponent=""
sOriginatingApplicationVersion="5.5.2.1" sOrigina
tingSystem="LSDSLSCPRD"
sOriginatingAccount="lsds\lscadmin"
dtPostTime="2017-03-05T10:35:43"
sMessage="Privileged Identity Console (running as user lsds\lscadmin) on system LSDSLSCPRD;
opened the password vault."/>
```



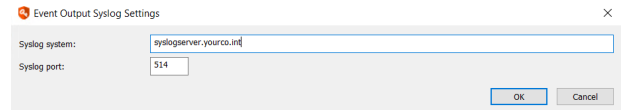
For more information on the message structure, text encoding, and data format of the output, please see "[Event Data Message Format](#)" on page 566.

## Pre-Built Ticketing and Logging Integrations

Privileged Identity ships with pre-built integrations for ticketing or logging systems. The following sub-sections identify these pre-built integrations and how to configure them.

### ArcSight

The event sink syslog will output the event information to a target server in a syslog compatible format. If the target syslog server is ArcSight, set the event sink output type to **Syslog to ArcSight (CEF)**. This will pre-configure the output format to syslog CEF and cannot be changed.



Click **Settings** and specify the following values:

- **Server Name:** Name of the machine to be used as the syslog server
- **Syslog port:** The default syslog port is 514 and sends via UDP.

Syslog output types simply write the event message in its entirety to a syslog server. These messages are written to the default messages log.

**i** For more information on how to configure the structure of the output and output data format to use a transform file, please see "[Event Data Message Format](#)" on page 566.

## BMC Remedy

Privileged Identity can work with BMC Remedy ITSM to generate new incidents when an event sink is configured.

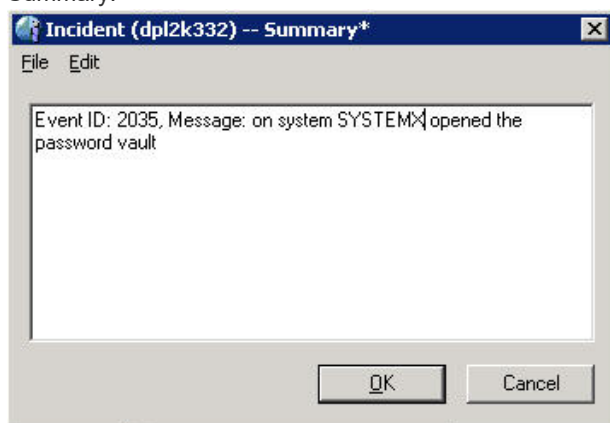
### Configure Event Sinks to Create Incidents

Privileged Identity can automatically create new incidents in the incident system using the event sink system.

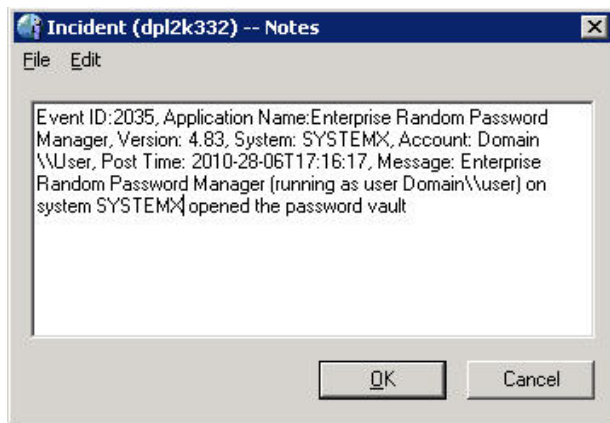
1. Go to **Settings | Extension Components | Configure Event Sinks**.
2. Configure the event sink notification triggers as required. Please see "[Configure Event Sinks in the Management Console](#)" on [page 542](#) for specifics on this topic.
3. Set the Event sink output type to **BMC Remedy via COM**. All required information will be used from the previously configured BMC Remedy integration dialog.
4. Click **OK** to save the event sink.

When the triggers configured in event sink occur, a ticket will be created within Remedy. The relevant information will be found in two locations within the ticket. The SUMMARY section, will have a preview of the information while the NOTES section will contain all information relevant to the event in question.

Summary:



Notes:





## HP Service Manager

Privileged Identity can work with HP Service Manager to generate new incidents when an event sink is configured.

### Configure Event Sinks to Create Incidents

Privileged Identity can automatically create new incidents in the incident system using the event sink system.

1. Go to **Settings | Extension Components | Configure Event Sinks**.
2. Configure the event sink notification triggers as required. Please see "[Configure Event Sinks in the Management Console](#)" on [page 542](#) for specifics on this topic.
3. Set the Event sink output type to **HP Service Manager via COM**. All required information will be used from the previously configured HP Service Manager integration dialog.
4. Click **OK** to save the event sink.

When the triggers configured in event sink occur, a ticket will be created within the incident system.

## Jira

Privileged Identity can work with Jira to generate new incidents when an event sink is configured.

### Configure Event Sinks to Create Incidents

Privileged Identity can automatically create new incidents in the incident system using the event sink system.

1. Go to **Settings | Extension Components | Configure Event Sinks**.
2. Configure the event sink notification triggers as required. Please see "[Configure Event Sinks in the Management Console](#)" on [page 542](#) for specifics on this topic.
3. Set the Event sink output type to **Jira via COM**. All required information will be used from the previously configured HP Service Manager integration dialog.
4. Click **OK** to save the event sink.

When the triggers configured in event sink occur, a ticket will be created within the incident system.

## OTRS

Privileged Identity can work with OTRS to generate new incidents when an event sink is configured.

### Configure Event Sinks to Create Incidents

Privileged Identity can automatically create new incidents in the incident system using the event sink system.

1. Go to **Settings | Extension Components | Configure Event Sinks**.
2. Configure the event sink notification triggers as required. Please see "[Configure Event Sinks in the Management Console](#)" on [page 542](#) for specifics on this topic.
3. Set the Event sink output type to **OTRS via COM**. All required information will be used from the previously configured HP Service Manager integration dialog.
4. Click **OK** to save the event sink.

When the triggers configured in event sink occur, a ticket will be created within the incident system.

## CA Service Desk Manager

Privileged Identity can work with CA Service Desk Manager to generate new incidents when an event sink is configured.

### Configure Event Sinks to Create Incidents

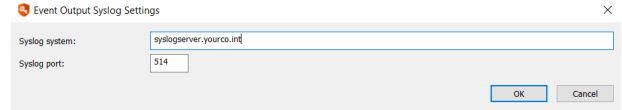
Privileged Identity can automatically create new incidents in the incident system using the event sink system.

1. Go to **Settings | Extension Components | Configure Event Sinks**.
2. Configure the event sink notification triggers as required. Please see "[Configure Event Sinks in the Management Console](#)" on [page 542](#) for specifics on this topic.
3. Set the Event sink output type to **CA Service Desk via COM**. All required information will be used from the previously configured CA Service Manager integration dialog.
4. Click **OK** to save the event sink.

When the triggers configured in event sink occur, a ticket will be created within the incident system.

## QRadar

The event sink syslog outputs the event information to a target server in a syslog compatible format. If the target syslog server is QRadar, set the event sink output type to **Syslog to QRadar (LEEF)**. This will pre-configure the output format to syslog LEEF and cannot be changed.



Click **Settings** and specify the following values:

- **Server Name:** Name of the machine to be used as the syslog server
- **Syslog Port:** The default syslog port is 514 and sends via UDP.

Syslog output types simply write the event message in its entirety to a syslog server. These messages are written to the default messages log.

**i** For more information on how to configure the structure of the output and output data format to use a transform file, please see ["Event Data Message Format" on page 566](#).

## ServiceNow

Privileged Identity can work with ServiceNow to generate new incidents when an event sink is configured.

### Configure Event Sinks to Create Incidents

Privileged Identity can automatically create new incidents in the incident system using the event sink system.

1. Go to **Settings | Extension Components | Configure Event Sinks**.
2. Configure the event sink notification triggers as required. Please see "[Configure Event Sinks in the Management Console](#)" on [page 542](#) for specifics on this topic.
3. Set the Event sink output type to **Service Now via COM**. All required information will be used from the previously configured HP Service Manager integration dialog.
4. Click **OK** to save the event sink.

When the triggers configured in event sink occur, a ticket will be created within the incident system.

## Microsoft System Center Service Manager

Privileged Identity can work with Microsoft System Center Service Manager to generate new incidents when an event sink is configured.

### Configure Event Sinks to Create Incidents

Privileged Identity can automatically create new incidents in the incident system using the event sink system.

1. Go to **Settings | Extension Components | Configure Event Sinks**.
2. Configure the event sink notification triggers as required. Please see "[Configure Event Sinks in the Management Console](#)" on [page 542](#) for specifics on this topic.
3. Set the Event sink output type to **System Center via COM**. All required information will be used from the previously configured HP Service Manager integration dialog.
4. Click **OK** to save the event sink.

When the triggers configured in event sink occur, a ticket will be created within the incident system.

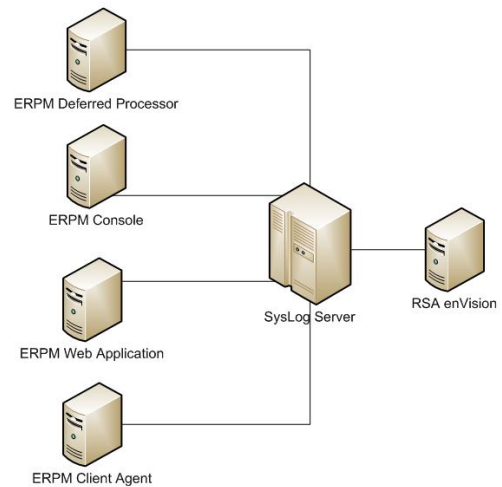
## RSA NetWitness (enVision)

You can configure each component to report its events to a target System Log Server. Configuration for which events are sent as syslog messages is configured through the management console through the use of Events Sinks, and can be configured to send all, or any subset of possible events, to the log system. Events can also be sent to multiple event log servers if desired for redundancy.

This section provides instructions for configuring Privileged Identity with RSA enVision. This document is not intended to suggest optimum installations or configurations.

It is assumed that the reader has both working knowledge of all products involved, and the ability to perform the tasks outlined in this section. Administrators should have access to the product documentation for all products in order to install the required components.

All Privileged Identity components must be installed and working prior to the integration. Perform the necessary tests to confirm that this is true before proceeding.



## Deploy an Event Source Integrator Package

The event source package for Privileged Identity must be deployed on all the appliances in the enVision site so that enVision can support the event source. When deploying the package, a script is run that assigns a unique event source type ID to the event source, which enVision uses when generating reports.

The event source XML package must be deployed on every appliance in the enVision site as described in the following table.

RSA enVision Site	Where to Deploy the Event Source XML Package
Single appliance site	On the appliance
Multiple appliance site	On all components: <ul style="list-style-type: none"> <li>• Application Servers (A-SRVs)</li> <li>• Database Servers (D-SRVs)</li> <li>• Local Collectors (LCs)</li> <li>• Remote Collectors (RCs)</li> </ul>
Multiple appliance site with Enhanced Availability	On all components: <ul style="list-style-type: none"> <li>• Application Servers (A-SRVs)</li> <li>• Database Servers (D-SRVs)</li> <li>• Cluster Appliances (CAs)</li> </ul>



The following steps will refer to an Event Source Package. The event source package will be located in the **SupplementalInstallers** folder of the installation directory. The file is named **LiebsoftERPMPPE.zip**.

1. Extract the EventSource Package into the following folder: **%\_ENVISION\etc\devices**.
2. Run the script file, **UpdateESType.vbs**, to assign an event source type ID to the event source. The time the script file takes to run depends on the number of event source XML files that need to be verified.
3. Restart the **NIC Service Manager Windows Service**. For more information, see the enVision Help topic **Start/Stop Services - Manage Services**.
4. At this point, login to the enVision console and see the new device type under **Overview > System Configuration > Devices > Manage Device Types** listed as **LiebsoftERPMPPE**.
5. Repeat steps 1 to 4 on each appliance in the enVision site.

## Configure Event Sink Output Type for RSA enVision Instance

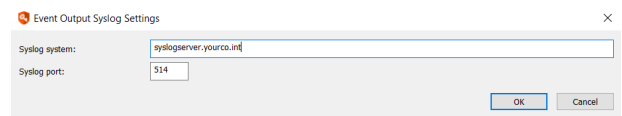
Configure an Event Sink output type for an RSA enVision instance. Open the Event Sink Configuration dialog within the management console and create a new Event Sink entry for RSA enVision.

The event sink syslog will output the event information to a target server in a syslog compatible format. If the target syslog server is RSA Netwitness (enVision), set the event sink output type to **Syslog**.

Click **Settings** and specify the following values:

- **Server Name:** Name of the machine to be used as the syslog server
- **Syslog Port:** The default syslog port is 514 and sends via UDP.

EventLog output types simply write the event message in its entirety to a syslog server. These messages will be written to the default messages log.



**i** For more information on how to configure the structure of the output and output data format to use a transform file, please see ["Event Data Message Format" on page 566](#).

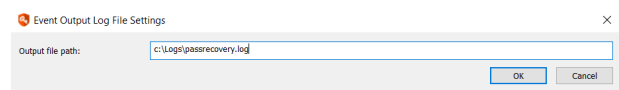
## Event Sink Descriptors and Modifiers

The Event Sink system can be used for a number of alerts and integrations. Some of these may require the data to be formatted for proper use. For example, opening a ticket in a help desk ticketing system may require pertinent pieces of information such as user, date stamp, account, and so on and those items must be listed in a particular order or format. The following sections describe how to format the data output of the event sinks.

## Event Data Message Format

Every event sink output type has data format settings. Click **Output Format** to specify those settings.

- **Output message structure:** Choose default for XML formatted data structure or syslog for syslog formatted output message structure.
- **Output text encoding:** Text type may be output as either UTF-8 or UTF-16.
- **Output data format:** Choose multiple formats:
  - **RAW XML Format:** Data is output in an XML format.
  - **Syslog LEEF:** Data is output in syslog format as modified for the QRadar LEEF specification.
  - **Syslog CEF:** Data is output in syslog format as modified for the ARCSight CEF specification.
  - **Custom Transform file:** Format the data as desired using a transform file.
  - **JSON:** Data is output in JSON format.
  - **JSON for FireEye Tap:** Data is output in JSON format as defined for the FireEye specification. Names of fields and other elements are sent in the FireEye expected format. This format uses name value pairs instead of line attributes.



For more information on using transform files, please see "[Event Sink Transform Files](#)" on page 568.

## Output Attributes

The event data is formatted as an XML string that contains attributes corresponding to the event type. All events contain the following attributes:

- **CompactMode:** XML format style
  - **sEventType:** This is the string value name that corresponds to the dwBasicEventType attribute.
  - **dwBasicEventType:** Event type. This attribute is what the event type filter in the event sink acts on.
  - **dwAppSpecificEventID:** This is the Event ID for this event. This attribute is used to filter events based on Event ID ranges.
- The four items above cannot be replaced in transform files. The remaining attributes listed below can be used in transform files.
- **sOriginatingApplicationName:** The name of the application that generated this event. This attribute is used to filter events by application name.
  - **sOriginatingApplicationVersion:** The major and minor product version for the application that generated this event.
  - **sOriginatingSystem:** The name of the system that is running the application that generated this event.
  - **sOriginatingAccount:** The logged in user account that associated with the running process that generated this event.
  - **dtPostTime:** UTC time that this event was generated.

- **sMessage:** The string message that is associated with this event. The body of the message will be arbitrary based on the process that generated it.

Events generated by the web application will contain the following additional attributes:

- **sIPAddress:** The IP address of the system that is using the web interface.
- **sLoginName:** The login name of the user using the web interface.
- **sManagerName:** The first manager identity in the delegation system that provides the user of the web interface access.

## Examples

Below is a sample base event as it would appear written directly to a text file with no transformation.

```
<Event CompactMode="1" sEventType="OpResult" dwBasicEventType="1" dwAppSpecificEventID="1300"
sOriginatingApplicationName="Random Password Manager" sOriginatingApplicationVersion="4.60"
sOriginatingSystem="SYSTEM" sOriginatingAccount="DOMAIN\pat" dtPostTime="2008-04-12T11:45:33"
sMessage="User DOMAIN\pat started Random Password Manager on system SYSTEM"/>
```

In addition to the base attributes, events can contain additional attributes that are specific to the event type. These additional arguments follow the same format `sAttributeName="AttributeValue"`. Events can also include contextual variables that may or may not be present. If the attribute is not always present based on the event type, it will be in a separate element within the event element named `mapContextVariables`. One or more of these optional attributes can be specified with any event.

Below is a sample event generated when a password is recovered through the web interface and an email alert is sent out notifying the administrator of the recovery.

```
<Event CompactMode="1" sEventType="OpResult" dwBasicEventType="1" dwAppSpecificEventID="109"
sOriginatingApplicationName="Random Password Manager" sOriginatingApplicationVersion="4.60"
sOriginatingSystem="SYSTEM" sOriginatingAccount="DOMAIN\pat" dtPostTime="2008-23-10T12:40:51"
sMessage="Mailed recovery alert to 'pat@example.com' for password checkout"
sIpAddress="192.168.8.1" sLoginName="DOMAIN\User"
sManagerName="DOMAIN\user"><mapContextVariables key="sSystemName"
value="DXPP01"/><mapContextVariables key="sAccountName" value="Administrator"/></Event>
```

Notice that in addition to the standard attributes, this event also includes the following attributes: `sIPAddress`, `sLoginName`, `sManagerName`. These attributes are specific to the web application context and are included with all events generated by the web application. This event also contains two context variables: `sSystemName` and `sAccountName`. These attributes are specific to this operation and are not always included in all events generated by the web application. Attribute values can be accessed and used by transformation files to format the event messages to suit the needs of the output type.

## Event Sink Transform Files

Transformation files can be applied to event sinks to change the format of the event message before it is sent to the output. Transform files use replaceable arguments to specify attributes in the event, and when the event is processed, the variables are replaced with the values of the attributes in the event. For standard attributes, use the **%attributeName%** convention to specify a replacement variable.

Below is an example of an email output transform file. A copy of this sample transform is duplicated in a transform file named **SampleEmailTransform.html** that is included in the Event Sink Code folder under the ExampleCode folder installed to the program installation directory. This example is used to reformat an event message into a specific HTML format before sending the message out to be e-mailed. The result is a properly formatted HTML email message instead of an XML encoded event message sent as an attachment to the email message.

```
<html>
  <head>
  </head>
  <body>
    Message from %OriginatingApplicationName% (version %OriginatingApplicationVersion%)
    <br>
    Running on system %OriginatingSystem% (Running as %OriginatingAccount%)
    <br>
    At %PostTime%
    <br>
    Message %Message%
    <br>
  </body>
</html>
```

Look at the sample event message below to see how the transform is applied.

```
<Event CompactMode="1" sEventType="OpResult" dwBasicEventType="1" dwAppSpecificEventID="1300"
sOriginatingApplicationName="Privileged Identity" sOriginatingApplicationVersion="5.5.2.1"
sOriginatingSystem="SYSTEM" sOriginatingAccount="DOMAIN\\fredo" dtPostTime="2017-04-12T11:45:33"
sMessage="User DOMAIN\\Fredo started Privileged Identity on system SYSTEM"/>
```

Notice in the transform specified, **%OriginatingApplicationName%** is used as one of the replacement arguments. In the event message, this argument corresponds to the **sOriginatingApplicationName** attribute. In this event, the value of that attribute is **Privileged Identity**. When the event goes through the transform, the replacement is done. Similar replacements are done for each of the specified variables.

The following items can be replaced in a transform file:

- **OriginatingApplicationName**: The name of the application that generated this event. This attribute is used to filter events by application name.
- **OriginatingApplicationVersion**: The major and minor product version for the application that generated this event.
- **OriginatingSystem**: The name of the system that is running the application that generated this event.
- **OriginatingAccount**: The logged in user account that associated with the running process that generated this event.
- **PostTime**: UTC time that this event was generated.
- **Message**: The string message that is associated with this event. The body of the message will be arbitrary based on the process that generated it.
- **IPAddress**: The IP address of the system that is using the web interface.
- **LoginName**: The login name of the user using the web interface.
- **ManagerName**: The first manager identity in the delegation system that provides the user of the web interface access.

After the transform is finished, an output message is generated that has this format.

```
<html>
  <head>
  </head>
  <body>
    Message from Enterprise Random Password Manager (version 4.83.9)
    <br>
    Running on system SYSTEM (Running as DOMAIN\Fredo)
    <br>
    At 20015-04-12T11:45:33
    <br>
    Message User DOMAIN\bob started Privileged Identity on system SYSTEM
    <br>
  </body>
</html>
```

Attributes either occur as strings and will be preceded with the letter **s** like **sOriginatingApplication** name, or are numbers and are preceded by **dw** like **dwBasicEvent** type. Attributes can be accessed by using the dropping the prefix and enclosing the rest of the attribute name in percent signs like **%OriginatingApplication%**.

If a specified attribute is not found in the event when the event is processed, the transform will replace the argument with [unknown] in the output message. For example, if specifying **%FakeAttribute%** in the transform file, then the output will also include the [unknown] string at the same location.

Replacing contextual variables is similar, to replace any variable enclosed in the **<mapContextVariable>** tag, use the syntax **%ContextVariable:Key%**. For example, the following web application operation contains two contextual attributes (**sAccountName** and **sSystemName**).

```
<Event CompactMode="1" sEventType="OpResult" dwBasicEventType="1" dwAppSpecificEventID="105"
sOriginatingApplicationName="Privileged Identity" sOriginatingApplicationVersion="5.5.2.1"
sOriginatingSystem="SYSTEM" sOriginatingAccount="DOMAIN\pat" dtPostTime="2017-03-12T11:33:05"
sMessage="Password lookup success for 'SYSTEM\Account'" sIpAddress="192.168.8.17"
sLoginName="pat" sManagerName="pat"><mapContextVariables key="sSystemName"
value="SYSTEM"/><mapContextVariables key="sAccountName" value="Account"/></Event>
```

To access the **sAccountName** variable in the transform file, use the syntax **%ContextVariable:sAccountName%**.



**Note:** As of the current release, transforms are only applied to the email output type.

## Event Sink XML File Format

Event sink files are usually created through the application using the event sink editor. The files themselves are saved to a directory specified by the application. This directory defaults to the program installation directory, but can be configured through the Event Sink Settings menu in the Event Sink Configuration dialog. The setting also is located in the registry under the **HKLM\Software\Lieberman\Liebssoft Generic Events COM Server\EventSinks** key. The **sEventSinksDirectory** string value represents a local file path to the directory to which new event sink files will be saved. This directory is also where all application will look for event sink files.

Event sink files are XML formatted files that specify the type of event to output, the range of events that will be processed, and an optional transformation that will be applied to the event message before it is output. Shown below is a sample event sink for an email alert. A copy of this sample event sink is located in the Event Sink Code directory in the Example code directory.

```
<?xml version="1.0" encoding="UTF-8"?>
<EventSink CompactMode="1" sName="LogFileSink" sDescription="" sEventOutputType="Email">
  <EventFilterSettings bMatchApplicationName="1" sApplicationName="Privileged Identity"
bMatchBasicEventType="1" dwBasicEventType="2" bMatchEventIDRanges="0">
    <listEventIDRanges dwLowValue="105" dwHighValue="110"/>
    <listEventIDRanges dwLowValue="120" dwHighValue="150"/>
  </EventFilterSettings>
  <EventOutput sSmtpSettingsName="Default">
    <EventOutputTransform eTransformType="1"
sTemplateFilePath="C:\\EventSinkExamples\\SampleEmailTransform.html"/>
    <listEmailRecipients eMessageTargetType="1" sName=""
sContactIdentifier="name@example.com"/>
  </EventOutput>
</EventSink>
```

The XML version and encoding information is standard and should not be modified. The **CompactMode** attribute defines the style of XML used, the value of 1 matches this sample format. The following elements, their corresponding attributes and descriptions are listed below. Note that some elements and attributes will change depending on the output type of the event sink as noted.

### EventSink element attributes

- **CompactMode:** XML encoding type. This should be set to 1.
- **sName:** Name of the event sink file. This name will show up in the event sink editor dialog and will also determine the name of the event sink file.
- **sDescription:** A descriptive name of the event sink. This attribute is currently not shown through the editor.
- **sEventOutputType:** The type of output action associated with this event sink. Possible values include: LogFile, ComCall, Email, RegistryValue, and NamedPipe.

### EventFilterSettings element attributes

- **bMatchApplicationName:** Boolean value of 0 or 1 specifying whether the event sink will filter events to only process those events that match the application name.
- **sApplicationName:** String value which defines the application name. If bMatchApplicationName is set to 0, this value is ignored.
- **bMatchBasicEventType:** Boolean value of 0 or 1 specifying whether the event sink will filter events to only process those events that match the basic event type.

- **dwBasicEventType:** Numerical value that specifies the basic event type. This value has the following mappings: 0-unknown, 1-Success, 2-Failure, 3-DebugStatusTrace, 4-LogMessage, 5-Error. If bMatchBasicEventType is set to 0, this value is ignored.
- **bMatchEventIDRanges:** Boolean value of 0 or 1 specifying whether the event sink will filter events to only process those events that fall within the range specified by the listEventIDRanges elements. The listEventIDRanges elements each contain a dwLowValue and dwHighValue attribute that indicate the the start and stop of a contiguous range of event IDs. If an event has an eventID that falls within one of the ranges, the event sink will trigger the associated action when the event is raised. The event ID ranges should not overlap, but overlapping event ID ranges will not cause an error.



*For a list of all possible event IDs and their descriptions, please see "Event Sink Events List" on page 536.*

## EventOutput element attributes

- **EventOutputTransform:** This optional element specifies a transform to apply before processing the event message for output. This element can be present in any event output type, but currently only applies to the email output type. If it is specified with other output types, the values are ignored. The eTransformType should always be set to 1. The sName attribute indicates a friendly name for the transform, this attribute is optional. The sTemplateFilePath attribute indicates a local file path to the XML transform file for this event sink.

The elements and attributes of the EventOutput element are dependent on the type of output. The type of output is determined by the sEventOutputType attribute of the EventSink element.

The Log File output type contains the following additional attributes and elements:

- **sFilename:** Attribute of the eventOutput element that specifies the local file name path to the file where the event messages will be logged.

The Registry value output type contains the following additional attributes and elements:

- **sTargetSystemNetworkName:** Attribute of the eventOutput element that specifies the system name where the registry value will be logged.
- **sRegistryBaseKey:** Attribute of the eventOutput element that specifies the base registry key (Typically HKLM).
- **sRegistryPath:** Attribute of the eventOutput elements that specifies the path to the registry value where the messages will be written.
- **sRegistryValueName:** Attribute of the eventOutput elements that specifies the name of the registry value where the message will be written.
- **sNotificationName:** Attribute of the eventOutput elements that specifies the name of the event notification that will be sent when the message is written to the registry value.

The Named Pipe output type contains the following additional attributes and elements:

- **sPipeName:** Attribute of the eventOutput elements that specifies the name of the named pipe to which event messages will be sent.

The Com Call output type contains the following additional attributes and elements:

- **sComObjectProgID:** Attribute of the eventOutput elements that specifies the program ID of the COM object to call.
- **sMethodName:** Attribute of the eventOutput elements that specifies the name of the method on the COM interface to call. The event message is always passed as the single argument to this method.

The Email output type contains the following additional attributes and elements:

- **sSmtpSettingsName:** Attribute of the eventOutput element that specifies which email settings profile to use when sending email with this event sink. The email profiles are configured using the email settings editor through the application. By default, the profile is named default, but value can point to a different profile by configuring another email profile in the registry and referencing it by the name of the base key. Email profiles are located in the registry under **HKLM\Software\Lieberman\SmtpSettings**.
- **listEmailRecipients:** Element that contains a single email message recipient. One or more of these elements can be specified to indicate multiple email recipients. The sMessageTargetType attribute should be set to 1. The sName attribute is a descriptive name for this recipient that is not used. The sContactIdentifier attribute should be the email address of the target recipient.



## Privileged Sessions

Application launching into a privileged session and session recording are optional features enabled by licensing. See the Application Launcher & Session Recording guide for more information on installing, configuring and using the application launcher.

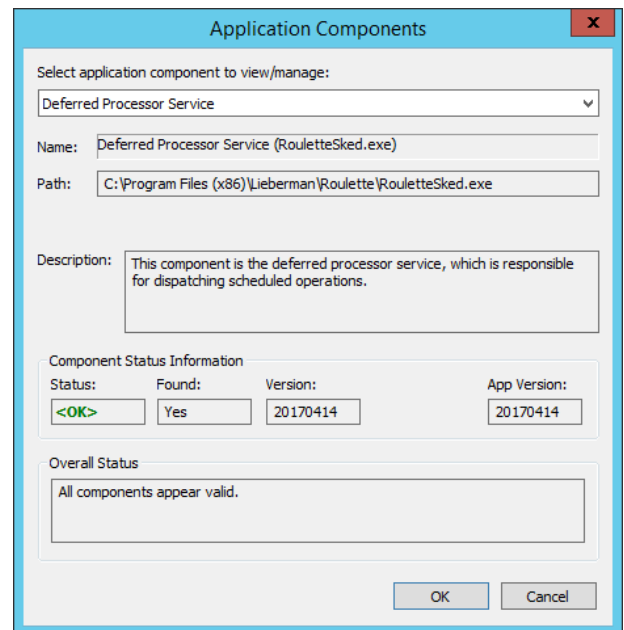
## Program Maintenance & Security

Privileged Identity requires consideration when installing regarding security configuration. Additionally, you must perform ongoing maintenance to maintain database health. This section outlines some of those security considerations and maintenance actions.

### Application Components

Application Components settings are found at **Settings | Application Components**.

Application Components verifies that all of the local components that Privileged Identity requires are installed and configured properly and are the correct versions. The **Overall Status** section near the bottom of the dialog will provide an alert of any problems with application components. For each component in the dropdown list, the name, path to the component, description, status, and version information for that component are displayed. If one or more components have been moved, they must be returned to their original location or re-run the installer to ensure that components can find their required files. The four required components are the Deferred Processor Service, the Deferred Processor, the COM object file for the web interface, and the web interface directory containing the ASP pages.



## Database Maintenance

Privileged Identity leverages a Microsoft SQL database for its primary program data store. Privileged Identity exposes access to SQL Server utilities that can be used to help maintain the database. However, a DBA should be engaged to automate these actions directly on the database host. Moreover, the operations performed are based on the SQL server's recommendations. An experienced DBA may be able to provide greater insight into performance bottlenecks.

Finally, while Privileged Identity can make use of highly available database configurations, a good database backup strategy should always be observed. The more frequent the password management, the more frequent your database should be backed up by your DBA.

This section covers the built-in SQL utilities exposed by the Privileged Identity interface. All operations can be performed directly on the database server if there are any issues running them from the management console. You may experience problems running the utilities from the management console if the queries take too long to execute from the remote management console.

The following database maintenance utilities are exposed by Privileged Identity:

- **SQL Server Auto Index Tuning** - Queries the programs data store server (SQL Server only) for recommended indexes. Indexes are recommended by Microsoft SQL Server and NOT by Privileged Identity. Adding indexes can lead to significant performance gains. There is no equivalent option within the UI for an Oracle database server. Use of this feature requires the SQL Server right of View Server State. If this right is not granted, you will receive an error when opening the index tuning dialog.
- **SQL Server Index Defragmentation** - Indexes can become fragmented over time and impact read/write performance of the program data store. To keep these indexes performing well, use the SQL Server Index Defragmentation option. There is no equivalent option within the UI for an Oracle database server.
- **SQL Server Generate Stats Fullscan** - Is used to help maintain performance over time when using a Microsoft SQL Server database. There is no equivalent option within the UI for an Oracle database server.
- **App Data Store Maintenance** - Is used to help automatically schedule performance and data pruning activities when using a SQL Server database. These options work for all supported data stores.

## SQL Server Auto-Index Tuning

SQL Server Auto-Index Tuning is found under **Settings | Data Store Configuration | SQL Server Auto-Index Tuning**.

SQL Server Auto Index Tuning queries the program's data store server (SQL Server only) for recommended indexes. Indexes are recommended by SQL Server and NOT by Privileged Identity. Adding indexes can lead to significant performance gains. There is no equivalent option within the program UI for an Oracle database server.

Use of this feature requires the SQL Server right of **View Server State**. If this right is not granted, you will receive a non-fatal error when opening this dialog.

To add any recommended index, select the index and click **Add Selected Indexes**. When choosing to add an index, it is best to select one at a time and add the index. This is because there may be multiple optimizations affecting the same table and attempting to add them both will simply result in a non-fatal error. Repeat until all suggestions are gone from the list. If attempting to add the index results in an error during the add index operation (as seen in the main program log), simply delete the existing index directly from within SQL Server. The name of the offending index will be referenced in the main program log.

However, indexes can get out of date and can also lead to significant performance degradation as the usage profile changes and the system, account, and password information changes over time. While this feature makes it easy to control certain aspects of database performance, it is not meant as a replacement for an application DBA specialist who is familiar with performance tuning of the SQL Server database.

As indexes become very outdated (especially through upgrades) they can cause problems if not completely rebuilt from the ground up. Log messages such as this may be returned from the Microsoft OLE DB provider and shown in the program or job log:

```
Transaction (Process ID 1259) was deadlocked on lock resources with another process and has been chosen as the deadlock victim. Rerun the transaction.
```

For more information about Microsoft SQL server deadlock scenarios, please refer to Microsoft's MSDN web site:

[http://msdn.microsoft.com/en-us/library/ms178104\(v=sql.105\).aspx](http://msdn.microsoft.com/en-us/library/ms178104(v=sql.105).aspx)

And to control deadlock priority:

<http://msdn.microsoft.com/en-us/library/ms186736.aspx>

To help reset the index status, Privileged Identity offers a "big-hammer" approach. This approach will delete all auto-created indexes. This may have one or two consequences:

- The database returns to proper working order and no longer has deadlock issues
- The database performance suffers in the short term because no more indexes exist. This is because indexes generally improve performance.

If the big hammer approach is used, we highly recommend running **SQL Server Generate Stats Fullscan**, which is discussed later in this section.

The following query determines the MS SQL index recommendations (replace **database\_name\_here** with the actual data store name):

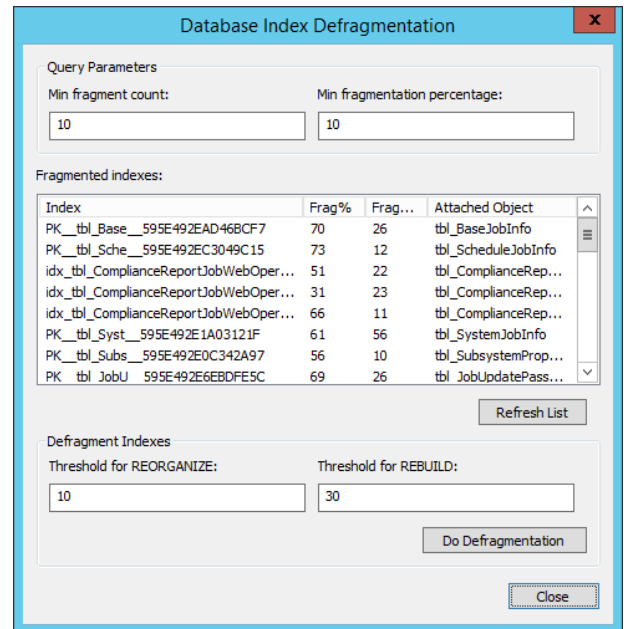
```
SELECT '$PREFIX$_' + LEFT (PARSENAME(mid.statement, 1), 32) + '_$HASH$' AS index_name, migs.avg_total_user_cost * (migs.avg_user_impact / 100.0) * (migs.user_seeks + migs.user_scans) AS improvement_measure, 'CREATE INDEX [$PREFIX$_' + LEFT (PARSENAME(mid.statement, 1), 32) + '_$HASH$]' + ' ON ' + mid.statement + ' (' + ISNULL (mid.equality_columns, '') + CASE WHEN mid.equality_columns IS NOT NULL AND mid.inequality_columns IS NOT NULL THEN ',' ELSE '' END + ISNULL (mid.inequality_columns, '') + ')' + ISNULL (' INCLUDE (' + mid.included_columns + ')', '') AS create_index_statement, migs.*, mid.database_id, mid.[object_id] , PARSENAME
```



## SQL Server Index Defragmentation

SQL Server Index Defragmentation is found under **Settings | Data Store Configuration | SQL Server Index Defragmentation**.

Indexes can become fragmented overtime and impact read/write performance of the program data store. To keep these indexes performing well, use the **SQL Server Index Defragmentation** option. There is no equivalent option within the program UI for an Oracle database server. Many customers create jobs directly in the database that defragment the indexes on a regular basis. This is a concept applicable to both SQL Server and Oracle databases.



Index	Frag%	Frag...	Attached Object
PK__tbl_Base__595E492EAD46BFC7	70	26	tbl_BaseJobInfo
PK__tbl_Sche__595E492EC3049C15	73	12	tbl_ScheduleJobInfo
idx__tbl_ComplianceReportJobWebOper...	51	22	tbl_ComplianceRep...
idx__tbl_ComplianceReportJobWebOper...	31	23	tbl_ComplianceRep...
idx__tbl_ComplianceReportJobWebOper...	66	11	tbl_ComplianceRep...
PK__tbl_Syst__595E492E1A03121F	61	56	tbl_SystemJobInfo
PK__tbl_Sub__595E492E0C342A97	56	10	tbl_SubsystemProp...
PK__tbl_JobU__595E492E6EBDFESC	69	26	tbl_JobUpdatePass...

The query used to determine the MS SQL Server index fragmentation is (replace **database\_name\_here** with the actual data store name):

```
SELECT tblIndexes.name AS IndexName, tblIndexStats.avg_fragmentation_in_percent AS
FragmentationPercent, tblObjects.name AS ObjectName, tblIndexStats.fragment_count AS
FragmentCount FROM sys.dm_db_index_physical_stats ( DB_ID(), OBJECT_ID(N'database_name_here'),
NULL, NULL, NULL ) AS tblIndexStats JOIN sys.indexes AS tblIndexes ON tblIndexStats.object_id =
tblIndexes.object_id AND tblIndexStats.index_id = tblIndexes.index_id JOIN sys.objects AS
tblObjects ON tblIndexStats.object_id = tblObjects.object_id WHERE tblIndexStats.avg_
fragmentation_in_percent >= 10 AND tblIndexStats.fragment_count >= 10
```

## SQL Server Generate Stats FullScan

SQL Server Generate Stats Fullscan is found under **Settings | Data Store Configuration | SQL Server Generate Stats Fullscan**.

As the program data store is used, over time the usage profile changes. Starting with MS SQL Server 2005, usage statistics are tracked by the SQL Server. This allows it to predict most used tables and queries and "optimize" itself. However, the usage of Privileged Identity is not a constant over time. For example, during deployment there may be heavy usage of discovery and management features and next to no password retrieval and auditing. Once the deployment is largely contained, the management features become relatively static in overall usage but the password retrieval and auditing now take a primary role. To help maintain consistent performance overtime, it is recommended to periodically run the **SQL Server Generate Stats Fullscan** option. There is no equivalent option within the program UI for an Oracle database server. Many customers create jobs which re-generate the usage statistics on the the entire database on a regular basis; this is a concept applicable to both SQL Server and Oracle databases.

The query used to generate the SQL Server Stats Fullscan is:

```
update statistics table_name with fullscan;
```

This is done for every table within the primary database.

If there are any problems running the fullscan, such as a database timeout, it will be noted in the main program log. The query and table will be noted in the log. For such long running queries, it is possible to either change the database timeout on the database basic configuration dialog, or the query may be run directly from SQL management studio.

## App Data Store Maintenance

To open the App Data Store Maintenance configuration dialog in the management console, go to **Settings | Data Store Configuration | App Data Store Maintenance**.

Privileged Identity maintains the data it collects indefinitely unless explicit steps are taken to remove the data. To help automate this data pruning, use the App Data Store Maintenance utility. When enabled, it prunes old information from the database, such as obsolete jobs, audit messages, operation messages, and more.

You can run this utility on demand using the **Run Now** button, or you can schedule this utility to run on a recurring basis.

The following data can be pruned from the data store:

- **Truncate Operation Messages** - Removes operation messages created during job runs. Define the number of days to keep old job log data for.
- **Truncate Web App Audit Log Messages** - Removes operations logged by the web application, visible in the audit logs. Define the number of days to keep old job log data for.
- **Truncate Obsolete Jobs** - Removes jobs that have previously ran and will never run again. Typically this means jobs that are set to run immediately, run once, and interactive jobs. Define the number of days to keep old jobs for.
- **Cleanup Item References** - There are many items that are cross referenced by other objects, e.g. job logs, job settings, and jobs.
  - **Remove credential references without username** - Data of this type is not typically found in the program data store and is most often the result of direct database manipulation. This option removes "account in use" references when the account to associate with the in use item is not cataloged with the item.
  - **Remove account store references to management sets that no longer exist** - If a management set is removed and account stores are left behind, this option will clean up the orphaned account stores.
  - **Remove password change job data for jobs that no longer exist** - When a password change job is deleted, the password change settings that are stored separately are left behind. This option will remove that data.
  - **Remove job log data for jobs that no longer exist** - When a job is deleted, the job logs are left in the database for future reference. This option will remove the job log data.
  - **Remove custom account store configuration references** - If an account store is removed it leaves behind associated account stores (systems). Enable this option to remove orphaned account stores.
  - **Remove propagation target data for password change jobs that no longer exist** - Jobs track propagation information separately from the job in the database. When the job is removed, the information is left behind. Enable this option to cleanup the data left behind when the job is removed.
  - **Remove permissions on shared credential lists that no longer exist** - This option will remove shared credential list permissions left behind when a shared credential is removed before the permissions are removed from the list.
  - **Remove permissions on groups for identities that no longer exist** - When an identity is granted permissions on a management set, and the identity is removed, the permissions will be left behind. This option will remove any permissions left behind after the identity is removed.
  - **Remove permissions on management sets for management sets that no longer exist** - When a management set is removed that has been defined with per-management set permissions, the permissions will be left behind. This option will remove any permissions left behind after the management set is removed.
  - **Remove items in use data for orphaned systems** - When a system is refreshed for account usage, that information is stored in the program data store. When a system is removed, that usage information is left behind. Enabling this option will remove the account usage data left behind after a system is removed from the solution.
  - **Remove permissions on files for identities that no longer exist** - Identities granted permissions on files in the file store, will leave those permissions behind when the identity is removed. This option will remove those orphaned permissions.



- **Remove permissions for roles for which the authentication server or role no longer exist** - An authentication server entry is used to direct an identity to the correct authentication server. If the authentication server entry is removed, the identities associated with the authentication server entry will be left behind. This option will remove the orphaned identities.
- **Remove file data from the store for file entries that have been removed** - Enable this option to remove the binary data left behind after a file has been removed from the file store.
- **Remove job schedule data from the store jobs that have been removed** - When a job is removed, the scheduling data is left behind. Enabling this option removes the scheduling data for jobs that were removed.
- **Remove system in group data for systems that no longer exist** - Data of this type is not typically found in the program data store and is most often the result of bad data merges or direct database manipulation. This option will remove the data associating a non-existing system with a management set.
- **Optimization Remove Unused Auto-Created Indexes** - If indexes are created using "[SQL Server Auto-Index Tuning](#)" on page 576, they may be later replaced with newer indexes. Enabling this option will cleanup previously created indexes that are no longer used.
- **Truncate Management Set Update Data** - When a management set is updated, information about that management set update job is stored in the database. This option will remove old update information. Define the number of days to keep old jobs for.
- **Truncate Network Scan Data** - When the network scanner is used to help populate a management set, it captures data about the devices scanned, networks, scanned etc. This option will cleanup old data, for example, scan metadata that could be used to help create a user defined system type mapping.

App data store maintenance does not run automatically until configured to do so. This means Privileged Identity will keep all data indefinitely until a cleanup action is taken.

If you desire to run app data store maintenance only on demand, configure your options, leave the job configuration to **Do not run app data store maintenance periodically**, and click **Run Now** when you wish to run it. If you wish to run the maintenance automatically set the periodic maintenance option to **Run app data store maintenance periodically** and configure a job schedule using the **Configure** button.

## Security Considerations

The security options mentioned in this section are descriptions of Windows, IIS and other settings that can be made. The suggestions made are not meant to be definitive and are not supported by BeyondTrust. Some of these settings can represent a fundamental change in your networks' overall security posture, may impact other network operations, and most will require additional configuration of your infrastructure. Please evaluate and research all settings fully for implementation in your own network.

### Program Data Protection

- **Encryption of passwords in the database** - Use an HSM such as those offered by Safenet or Thales/nCipher. Although the software encryption is the FIPS certified algorithm, it is encryption done in software which means the clear text information and encryption keys are in main system memory and CPU. Using an HSM will remove the key management from main system memory and provide further encryption protection as addressed by FIPS 140-2 level 2 and level 3 mechanisms. Whether this is provided as a network device or a local PCI device is up to you. A local device will be faster by comparison of a network device, but key management can be much harder to manage when dealing with VMs or distributed components.
- Use SSL/TLS protection for the SQL database connections. Passwords are encrypted before being sent/written to the SQL Server. However, some data, like system information is sent in the clear. SQL Server supports SSL connections for its network traffic. This has the benefit of ensuring absolutely no data is transmitted in the clear over the wire.
- Enable TDE ([Transparent Data Encryption](#)) in the program data store to further protect data at rest.
- Use SSL/TLS protection from the web server to the client browser. The use of SSL certificates to implement an HTTPS connection is a function of IIS. For better protection consider disabling SSLv3 at the server level and force the use of the latest TLS scheme.
- Export the encryption key post installation and store in a secured location, preferably as an encrypted file.
- Change the program encryption key periodically. This is a manual process but helps to ensure no data required to access the stored passwords remains static indefinitely.
- Change the default password on the password store access from within the console.

### Hardware Security Module (HSM) Distribution

The encryption process takes place before information is ever written to the data store (MS SQL/Oracle). This means, if the intention is to use a local HSM device, an HSM would be installed wherever there is a management console/deferred processor/zone processor or a web site. Following a fully redundant deployment scenario where there are 6 machines (2 x web, 2 x console, 2 x DB) there would be 4 HSM devices. In this scenario, it could be more cost efficient to go with a network capable HSM, but speed and availability become dependent on network infrastructure for the HSM.

### Network Traffic Protection

- Implement IPSec with ESP and AH. Preferred authentication methods would be PKI or Kerberos. IPSec can be selectively implemented to be required whenever traffic is sent from the management console and deferred processor and/or zone processor systems. IPSec policy can be configured such that it is required for all systems or only for traffic emanating from Privileged Identity

This is most easily implemented as a group policy and will help in a few ways:

- ESP will protect the entire packet payload
- AH will verify the source and destination systems
- Traffic will no longer appear as SMB but rather ISAKMP with no relevant information being gleaned from the data

## Password Management and Hashing

- For Microsoft Windows systems, set passwords that are 15 characters or longer and/or use group policy to disable LAN Manager hashes. If the group policy to disable LAN Manager hashes is not enabled, setting password that are 15 characters in length will have the same effect. This helps prevent against rainbow table attacks.
- Never configure a static value password when changing passwords. These passwords are known by the Privileged Identity administrator, and anyone else that password is given to which limits the company's ability to assign accountability should something bad happen involving the account. Further these passwords will never change unless a new password change job is run against them. Rather use the random password option and ensure the option in the program is set to use a unique password for each account. This ensures that no two accounts will ever have the same password and that it will likely be randomized following password checkout which further helps against pass-the-hash attacks.

## Web Application Configuration

These settings are part of the Security tab configuration for the web application.

- Do not allow default authenticated user access to the web site. This allows any user not explicitly granted a permission to login access to the web site of some kind just as the “Authenticated Users” group does in Windows.
- Enable and configure the Hide Recovered password option to automatically hide the displayed password after a certain amount of time.
- Force inactive web session timeouts to a short value that is relevant to how your users work. A shorter value will ensure users who recover passwords and then continue to leave the web site open will not be able to leave themselves logged in indefinitely.
- Configure the web site options to require secure sessions. If a session is established and SSL is not enabled, the web site will not authenticate the user and will not pass the credentials over the wire.
- Use integrated authentication and do not use browsers other than internet explorer 9 or later. This helps to ensure passwords are not sent in the clear and helps to guarantee your patch management cycle.
- Disable the copy button will stop the copy button feature from working in our product thus making it harder for the user to place password data onto their clipboard.
- Enable the option to Disable concurrent logins from a single user.
- Enable the option to embed a unique identifier with each page and with each request. This purpose of these options is to help prevent replay attacks and cache poisoning attacks. The downside is that users will only be able to click one link per page before having to re-navigate to the same page to perform a second action. This will also effectively disable the back button just as an online banking site would.
- Enable the Store only authentication information in the token. This may slow down some page operations for low powered users, it will ensure no information about the user login can be gleaned by examining the cookie.
- Configure the web site options to force logout on page errors to help protect against brute force or injection or other types of attacks.
- Configure two-factor authentication using either RSA or using OATH (included). Specifically, configure the users with 8 digit HOTP type tokens. This requires the user's to have either a soft token on the computer or smart phone or a physical device. TOTP tokens are available but provide weaker security as they will either email or SMS the pass code to the user which means other users with access to the user's email account can still potentially leverage the user's two-factor token.
- Enable Prevent the requesting user from granting a password request option to prevent users engaging in a workflow (password request) from being able to grant their own request when permissions would otherwise allow them to grant their own requests.
- Use certificate-based authentication rather than delegating to users and groups or explicit accounts where possible. This implies there is some form of certificate management but is again designed to prevent the passing of user credentials over the wire. This requires users have certificates and IIS is configured to at least accept user certificates and also requires the web site to require SSL.

- Configure Frequent request redirection to a value of 10 (value may need to be adjusted based on actual usage) to prevent DDOS style attacks against the web site.

## Management Console Access

- Ensure the host system does not permit interactive logon except by admins of the product.
- Ensure the host system does not permit network access except by admins of the product and the service account(s).
- Configure the Management console's console delegation to ensure only those specific users can launch the console. These users will still need to be administrators of the host system and be granted specific rights as defined in the Database Access by Service Accounts section for the interactive user in this document.

## Database Access by Service Accounts

- The COM object access the database to read and write data and will require the ability to run stored procedures and query views. However, it has no hand in creating tables or stored procedures or views. As such it should not be the same user account as is used to run any of the scheduling services as it does not require any target system access and does not require the same database privilege. It will only need:
  - Server connect/logon
  - Db\_datareader
  - DB\_datawriter
  - Execute on the database
- When the console is launched, it is responsible for ensuring the integrity of the database and its tables, views, stored procedures, etc. As such the interactive login account here should have the following permissions:
  - Server connect/logon
  - Db\_datareader
  - DB\_datawriter
  - Execute on the database
  - Db\_ddladmin
  - **Optional - View server state server level permission** - used for generation of indexes and index defrag and stats full scan operations
- The service accounts which access the program data store have limited rights compared to the interactive users but have more overall rights than the COM object identity as it is used to manage target systems. As such, it should be a different account than that used for the COM identity. However, the database rights are consistent with those of the COM identity:
  - Server connect/logon
  - Db\_datareader
  - DB\_datawriter
  - Execute on the database

For Oracle, the rights are simplified to:

- CONNECT
- CREATE TRIGGER
- CREATE SEQUENCE

- CREATE TABLE
- CREATE VIEW

## Service Accounts

- The COM object is not typically used to connect to or manage target systems. Though there are two functions which may be performed by the web site which do connect to systems, these functions are not typically allowed or performed. As such it should not be the same user account as is used to run any of the scheduling services
- The service accounts which run the deferred processor or zone processor services do perform actual system management such as password changes, propagation, and account elevations. As such, it should be a different account than that used for the COM identity.\*

## Service Account Rights to Active Directory

- On the topic of least privilege, what is required in AD depends on the accounts being managed and the password propagation being included.
- Administrator rights of any member server or workstation will be required in order to reset the password of the accounts such as Administrator or anyone in the administrators group. Whether that happens based on local group membership or via AD group membership doesn't matter.
- For Active Directory, domain admin or an administrator membership in the domain is not necessarily a requirement. These requirements depend on what type of account is being managed and if it will also be propagated to a domain controller service/task/etc.. Specifically...
- If managing the password of a regular user, the Privileged Identity service account needs only be delegated "reset password". Regular user is defined as user who is not in a protected group such as Account Operators, Server Operators, Administrators, Domain Admins, Enterprise Admins, etc.. This is a Microsoft restriction which can be changed by modifying specific security policies on the enterprise to permit low powered users to reset administrative accounts.
- If managing the password of a protected user (see above), then you must be at least in the administrators (domain local) group of the domain.
- For either of the above cases, if the service account being managed will also have its password propagated to a domain controller, then the service account must be in at least the administrators (domain local) group or higher in the domain.
- In another scenario, regarding service accounts, the deferred processor/zone processor may be configured to run as LocalSystem rather than a user account. This would then require the computer account be granted appropriate permissions in lieu of the service account credentials. The COM identity must however run as a real account.

## Database Services

- Database should be hosted on its own servers and database files should be on separate spindles than the logs or the OS.
- Database server should be hosted in its own unique instance that is not shared with other applications.
- Database host should not host other database for other programs at all.
- Database instance should not be on a default port (1433 for MS SQL or 1521 for Oracle).
- Database should not host other unnecessary services.
- Use integrated authentication where possible.
- For MS SQL, configure Privileged Identity to use a non-default schema that is set to DBO (DB basic configuration). This ensures a consistent schema usage across all operations.

- Services should run as an AD managed service account if possible.
- Disable unnecessary database services such as SQL Agent and SQL browsing services.

## IIS

- Configure web site to run in an application pool that is separate from any other web sites hosted on the web server host.
- Configure the web site to run under non-standard ports (other than 80 or 443).
- Web server could be a core installation of 2008 R2 server or later, with the most current version of Windows Server recommended.
- Enforce the use of the latest TLS scheme rather than allowing simple SSL or require IPsec.

## Supplemental

Overall, apply high-security policies but be aware, certain provisions will have to be made to those policies to allow our service accounts to run and connect - see installation guide for more information. There are various other recommendations for HA and DR. The basic premise is put each role (web, app, DB) on their own set of servers and add redundancy as is appropriate to each tier. I am happy to talk in depth about these things as well.

DOD STIGs also provide much guidance for hardening a Windows Server, it will be worthwhile to review these items as well:

- 2008 R2: <https://nvd.nist.gov/ncp/checklist/377>
- 2012 R2: <https://nvd.nist.gov/ncp/checklist/560>
- 2016: <https://nvd.nist.gov/ncp/checklist/753>

## Addenda

The addenda section contains supplementary information about this solution or related components. While some topics are covered in this guide, for more help, be sure to check out our forums and support center.

## Help Desk Integrations on Remote Systems

When a help desk integration is configured via the management console, the configuration is only stored with the management console. That means any remote web application hosts, web service hosts, zone processors or secondary consoles will not have the integration available to them without further administrative action.

To configure the help desk integrations for remote systems requires installation of the IntegrationComponents and manual copying of registry entries and files and folders from the local file system.

### Configuring Help Desk Integrations on a Remote Host

1. On the management console, configure the required help desk integration system.
2. Open regedit.
3. Navigate to **HKEY\_LOCAL\_MACHINE\SOFTWAREWow6432Node\Lieberman\ThirdPartyRemedyApp**.
4. Locate the specific integration sub-key, e.g. ServiceNow for the ServiceNow integration, and select it.
5. Make note of the **ConfigDataDirectory** in that sub-key.
6. Right-click on the sub-key and select **Export**. Save the registry file export as it must be imported on the remote system.
7. Copy the registry export file to the target host and import the registry export file on the target host.
8. Open file explorer, go to the **ConfigDataDirectory** directory noted in step 5.
9. Copy the entire directory to the exact same location on the target server.
10. Copy IntegrationComponents.msi from the SupplementalInstaller folder where Privileged Identity is installed to the target host.
11. Run the IntegrationComponents.msi installer on the target hosts to install the integration components. Accept all defaults.

If the target is a zone processor or management console host, restart the zone processor (via the services snapin) or restart the console.

If the target server is a web application or web service host server, open **Component Services** (dcomcnfg), expand **COM applications**, find **PWCWebComApp** (web application) or **Privilege Identity ERPM Web Service** (web service). Right-click on the application and select **Shut down**. The application will auto-start the next time it is accessed.

# Host Server Patching, Anti-Virus & IDS/IPS

## Host Server Patching

Microsoft security patches should be applied to all servers hosting Privileged Identity components according to your organization's patching guidelines. You do not need to stop any services or make any other changes prior to installing security patches.

Installing security patches should not have any impact on systems or accounts managed by Privileged Identity.

## Anti-Virus

While most customers report little to no impact, it has been noted that certain anti-virus configurations, especially if you run both the Windows Firewall and the A/V Firewall at the same time, can severely impact network performance. Network communication is used for nearly every component of Privileged Identity for normal operations.

Typically most anti-virus programs will not cause functional problems with Privileged Identity. However, part of troubleshooting and deployment may involve examining A/V logs or temporarily suspending anti-virus services.

## IDS and IPS

During normal management or refresh operations, Privileged Identity can spawn hundreds of threads or more. The protocols will vary based on the target systems or the operations being performed. This can cause IDS and IPS systems to trigger alerts or block network traffic during management operations, even though a ping or basic RPC connection may work during normal troubleshooting.



# Namespace Values

Namespaces are used to define the type of system being acted upon during a password import or retrieval. The namespaces, with the exception of windows systems, are all wrapped in square brackets.

System Type	Namespace
<b>Operating Systems</b>	
AS400	[AS400]
Linux/Unix	[Linux]
OS390	[OS390]
TN3270	[3270]
Unix	[Linux]
Windows	NetBIOS system name OR NetBIOS domain name without brackets
<b>Databases</b>	
Microsoft SQL Server	[SQL Server]
My SQL	[MySQL]
Oracle	[Oracle]
PostgreSQL	[PostgreSQL]
Sybase	[Sybase]
Teradata	[Teradata]
<b>LDAP Directories</b>	
IBM Tivoli Directories	[LDAP]
Novell eDirectory Databases	[LDAP]
Oracle Internet Directories	[LDAP]
ViewDS Directories	[LDAP]
<b>Middleware, Application Servers, and Enterprise Software</b>	
Oracle PeopleSoft	[PeopleSoft]
Oracle WebLogic	[Oracle WebLogic]
<b>Network Devices</b>	
Cisco	[Cisco]
Dell Remote Access Control (DRAC) Devices	[DRAC]
IPMI	[IPMI]
<b>Cloud Service Providers</b>	
Amazon Web Services (AWS)	[Amazon Web Services]
Microsoft Azure	[Azure Active Directory]
Rackspace	[RackSpace Public Cloud]
Salesforce	[Salesforce]
SoftLayer	[SoftLayer]

System Type	Namespace
<b>Other</b>	
VMware ESX	[VMWare (ESX)]
Personal Password	[PersonalPassword]
External Accounts	[External]
All Other Custom Account Stores	[XXX] Where XXX is the name of the custom store

## Privileged Identity Limited Warranty

The media (optional) and manual that make up this software are warranted by BeyondTrust to be free of defects in materials and workmanship for a period of 30-days from the date of your purchase. If you notify us within the warranty period of such defects in material and workmanship, we will replace the defective manual or media (if either were supplied).

The sole remedy for breach of this warranty is limited to replacement of defective materials and/or refund of purchase price and does not include any other kinds of damages.

Apart from the foregoing limited warranty, the software programs are provided "AS-IS," without warranty of any kind, either expressed or implied. The entire risk as to the performance of the programs is with the purchaser. BeyondTrust does not warrant that the operation will be uninterrupted or error-free. BeyondTrust assumes no responsibility or liability of any kind for errors in the programs or documentation of/for consequences of any such errors.

This agreement is governed by the laws of the State of California.

Should you have any questions concerning this Agreement, or if you wish to contact BeyondTrust, please email [info@beyondtrust.com](mailto:info@beyondtrust.com).

# Privileged Identity License Agreement

This is a legal and binding contract between you, the end user, and BeyondTrust. By using this software, you agree to be bound by the terms of this agreement. If you do not agree to the terms of this agreement, you should return the software and documentation as well as all accompanying items promptly for a refund.

1. **Your Rights:** BeyondTrust hereby grants you the right to use RED Systems Management to manage the licensed number of systems purchased. This software is licensed for use by a single client and its designated employees, contractors and authorized 3rd parties to manage the systems owned/used by a single client. The software license may not be shared with unrelated 3rd parties.

The serial number provided by BeyondTrust is designed for installation on a specific machine. You may install an unlimited number of copies of RED Systems Management for your administrators that connect to the single licensed machine. All administrators can share the pool of purchased managed node licenses.

There are no limits to the number of web servers or clients that may access the data stored by your licensed copy of RED Systems Management. You may install and use the “RED Systems Management: Web Interface to Random Password Generator Password Recovery Console” with your duly licensed copy of RED Systems Management + Random Password Generator without any additional payment to BeyondTrust.

The cost of Microsoft web servers, SSL certificates, and other supporting equipment and technology are the sole responsibility of the user of this software-not BeyondTrust.

2. **Copyright.** The SOFTWARE is owned by BeyondTrust and is protected by United States copyright law and international treaty provisions. Therefore, you must treat the software like any other copyrighted material (e.g. a book or musical recording) except that you may either (a) make one copy of the SOFTWARE solely for backup and archival purposes, or (b) transfer the SOFTWARE to a single hard disk provided you keep the original solely for backup and archival purposes. The manual is a copyrighted work also--you may not make copies of the manual for any purpose other than the use of the software.
3. **Other Restrictions:** You may not rent, lease, or transfer the SOFTWARE to any other entity. You may not reverse engineer, de-compile, or disassemble the SOFTWARE that is provided solely as executable programs (EXE files). If the SOFTWARE is an update, any transfer must include the update and all prior versions.
4. **Notice:** This software contains functionality designed to periodically notify BeyondTrust of demo usage and of the detection of suspected pirated license keys. By using this software, you consent to allow the software to send information to BeyondTrust under these circumstances, and you agree to not hold BeyondTrust responsible for the use of any or all of the information by BeyondTrust or any third party.

When used lawfully, this software periodically transmits to us the serial number and network identification information of the machine running the software. No personally identifiable information or usage details are transmitted to us in this case. The program does not contain any spyware or remote control functionality that may be activated remotely by us or any other 3rd party.

BeyondTrust Corporation  
578 Highland Colony Parkway  
Ridgeland, MS 39157  
866.205.3650

Support: [www.beyondtrust.com/docs/index.htm](http://www.beyondtrust.com/docs/index.htm)

Web Site: [www.beyondtrust.com](http://www.beyondtrust.com)