



Click Studios

Passwordstate

Password Discovery, Reset and Validation

Requirements

Table of Contents

1	OVERVIEW	3
2	PASSWORDSTATE WEB SERVER SYSTEM REQUIREMENTS	5
3	HOSTS IN NON-TRUSTED DOMAINS	6
4	INSTALLING ORACLE DATA ACCESS COMPONENTS (ODAC).....	7
5	OFFICE 365 AND AZURE ACTIVE DIRECTORY ACCOUNTS	8
6	REMOTE SITE LOCATIONS AGENT	9
7	PASSWORD RESET SCRIPT REQUIREMENTS	10
8	PASSWORD VALIDATION SCRIPT REQUIREMENTS	12
9	PASSWORD DISCOVERY SCRIPT REQUIREMENTS	13
10	ENABLING POWERSHELL REMOTING PER HOST	14
11	ENABLING POWERSHELL REMOTING VIA GROUP POLICY	15
12	ACCOUNT DISCOVERY AND PASSWORD RESETS BETWEEN NON-TRUSTED ACTIVE DIRECTORY DOMAINS, OR AGAINST WORKGROUP COMPUTERS	19
13	LOCAL ADMINISTRATOR ACCOUNT PASSWORD RESETS WITHOUT THE USE OF A PRIVILEGED ACCOUNT CREDENTIAL.....	20
14	PASSWORD RESETS AND ACCOUNT VALIDATION FOR LINUX ROOT ACCOUNTS.....	21

1 Overview



In Passwordstate, through the use of PowerShell scripts, you're able to reset passwords for the following:

- Active Directory
- Windows Accounts
- Windows Scheduled Tasks
- Windows Services
- IIS Application Pools
- Cisco Routers and Switches
- Linux Accounts - including root (CentOS, Debian, Fedora, Mac OS X, Mint, Open SUSE, Oracle Linux, Oracle Solaris, RedHat Linux, Scientific Linux, Solaris, SUSE Enterprise Desktop, SUSE Enterprise Server, Ubuntu)
- Microsoft SQL Accounts
- MySQL Accounts
- Oracle Accounts
- MariaDB Accounts
- Palo Alto Firewalls
- PostgreSQL Accounts
- COM+ Components
- HP iLO out of band management cards
- HP H3C switches and routers
- HP Procurve switches and routers
- F5 BIG-IP Load Balancers
- IBM's IMM out of band management cards
- Dell's iDRAC out of band management cards
- VMWare ESX Accounts (Use Linux Scripts)
- Juniper Junos devices
- Juniper ScreenOS firewalls Accounts
- Fortigate Firewall Accounts
- SonicWALL Firewall Accounts
- Office 365 and Azure Active Directory Accounts
- You can also create your own scripts to perform any sort of processing when a Password is updated within Passwordstate

You are also able to perform certain 'validation' tasks to ensure the passwords in Passwordstate are accurate compared to what is being used on remote hosts, and you are also able to 'discover' Local Administrator Accounts, and various other 'Windows Dependencies – such as Windows Services, IIS Application Pools and Scheduled Tasks.

Click Studios

Click Studios designed the Password Reset feature to make use of Microsoft's PowerShell scripting capabilities, to eliminate the need to install custom agents on remote Hosts. These Reset & Validation features can also be used on Hosts in non-trusted domains.

-  Note: Passwordstate can also reset Active Directory accounts, but uses native .NET code for this instead of PowerShell scripts.
-  Note: If you do have strict firewalling between various networks, or manage client's infrastructure over the Internet, there is also a Remote Site Agent which can be deployed which can communicate securely over HTTPs. See Remote Site Locations documentation below for more information

2 Passwordstate Web Server System Requirements

To make use of the PowerShell Password Reset Scripts, the following is required on your Passwordstate Web Server:

- Microsoft Windows Server 2008 R2 & IIS 7.5
- Microsoft Windows Server 2012 & IIS 8.0
- Microsoft Windows Server 2012 R2 & IIS 8.5
- Microsoft Windows Server 2016 & IIS 10.0
- Microsoft Windows Server 2019 & IIS 10.0
- Windows 7 & IIS 7.5
- Windows 8 & IIS 8.0
- Windows 10 & IIS 10.0
- Microsoft .Net Framework 4.5
- AzureRM PowerShell Module if you want to reset Office 365 or Azure AD Accounts
- PowerShell 4.0 or Higher
- Oracle Data Access Components (ODAC) if you want to reset Oracle Passwords
- Microsoft Visual C++ 2013 Runtime - <https://www.microsoft.com/en-au/download/details.aspx?id=40784> (this will automatically be installed for you)

3 Hosts in Non-Trusted Domains

It is also possible to perform Password Reset and Validations for hosts which are in non-trusted domains. For this to occur, the following is required:

- Functioning DNS so domain controllers and Hosts can be contacted
- Firewall ports must be open to allow traffic through. Please refer to the following Open Ports documents which describes all features/modules of Passwordstate - https://www.clickstudios.com.au/downloads/version8/Passwordstate_Open_Port_Requirements.pdf
- A Privileged Account Credential must be supplied on the screen Administration -> Passwordstate Administration -> Privileged Account Credentials, in FQDN format i.e. user@mydomain.com
- The Active Directory Domain information needs to be added on the screen Administration -> Passwordstate Administration -> Active Directory Domains, and then linked to the relevant Privileged Account Credentials
- And when added host records on the Hosts screen, it is recommended the Host names are specified using FQDN i.e. serverabc@mydomain.com

4 Installing Oracle Data Access Components (ODAC)

If you wish to perform password resets for Oracle user accounts, you need to install the Oracle Data Access Components on the Passwordstate web server, and modify the path to these components in the two Passwordstate PowerShell scripts. To do this, please follow these instructions:

- Download **ODP.NET_Managed121012.zip** from <http://www.oracle.com/technetwork/database/windows/downloads/index-090165.html>
- Unzip the contents to a directory of your choice on the Passwordstate Web Server (not within the Passwordstate folder though)
- Open a command prompt as an Administrator and change to the directory "c:\oracleodp\odp.net\managed\x64" – the path will be different for you depending on where you unzipped the file
- Now type "configure.bat" and press the enter key. The screen will output a series of commands and then read "The operation completed successfully."
- If the path you've installed the data access components to is different to 'c:\oracleodp', then you will need to go to the screen Administration -> System Settings -> Password Reset Options tab, and update the path here
- Now restart the Passwordstate Windows Service

5 Office 365 and Azure Active Directory Accounts

In order to perform Password Resets and Account Heartbeat validations, you must first install the AzureRM PowerShell module on your Passwordstate Web Server. To do this, you can follow these steps:

- Open a PowerShell console as an 'Administrator'
- Type Install-Module -Name AzureRM
- Accept the two prompts to install the module, and wait for it to complete – it can take several minutes to complete

Your Passwordstate web server must also be able to make calls to the Internet to use this PowerShell module.

Azure Active Directory Permissions:

A standard user in Azure AD cannot reset their own account password, using the Powershell module Passwordstate uses. If you grant the user one of the following roles in Azure, then they will be able to reset their own password:

1. Helpdesk (Password) administrator
2. User Administrator
3. Global Administrator


Helpdesk administrator is the role with the least privileges, however this will also give the user the ability to reset other Azure user passwords. If you feel these permissions are too high, then you should use a privileged account that has this Helpdesk Administrator role, and assign it on your Password record. This privileged account will perform the reset of the password on behalf of the user.

To assign the Helpdesk Administrator role in Azure AD, log into the Azure AD portal as an Administrator, select **Azure Active Directory** -> **Roles and administrators**, and open the **Helpdesk (password) Administrator** role. Then click Add Assignment and search for the appropriate user, and save your changes.

6 Remote Site Locations Agent

If you have environments located behind firewalled environments, or look after client's networks with only Internet access to them, then you are able to deploy a Remote Site Agent to each network – please note additional license subscription is required for this.

With this Remote Site Agent, it has the same system requirements for account discovery, password reset, and account heartbeat as your internal network does, but the agent does communicate securely over HTTPS back to your Passwordstate API on a single port. Not only is the traffic passed in encrypted format within the HTTPS tunnel, but each Site Location also has its own In-Transit Encryption Key with further encrypts all traffic within the HTTP Body using 256bit AES Encryption.

 Note 2: Where you deploy the agent also requires PowerShell 4.0 or above, and the Agent is installed as a Windows Service. A Microsoft SQL Server is not required, as it uses a local SQLite database to store various data.

7 Password Reset Script Requirements

There are different System Requirements, and host configurations, depending upon which Password Reset scripts you would like to use. The following table describes the possible scenarios.

Windows Server 2008, Server 2008 R2, Windows 7

Script	Requirements
Reset Local Windows Accounts	<ul style="list-style-type: none"> PowerShell 2.0 or above PowerShell Remoting enabled
Reset Window Services Accounts	<ul style="list-style-type: none"> PowerShell 2.0 or above PowerShell Remoting enabled
Reset Windows Scheduled Task Accounts	<ul style="list-style-type: none"> PowerShell 2.0 or above PowerShell Remoting enabled
Reset IIS Application Pool Accounts	<ul style="list-style-type: none"> PowerShell 2.0 or above PowerShell Remoting enabled Server 2008 (not R2) requires the IIS PowerShell Snap-In to be installed on the target host - http://www.iis.net/downloads/microsoft/powershell Also requires the following PowerShell Cmdlet to be run in order for scripts to be run (default is Restricted on these operating systems): Set-ExecutionPolicy RemoteSigned
Reset COM+ Component Passwords	<ul style="list-style-type: none"> PowerShell 2.0 or above PowerShell Remoting enabled

Windows Server 2012, Server 2012 R2, Server 2016, Windows 8, Windows 10

Script	Requirements
Reset Local Windows Accounts	<ul style="list-style-type: none"> PowerShell 3.0 or above, and PowerShell Remoting enabled (these are default settings)
Reset IIS Application Pool Accounts	<ul style="list-style-type: none"> PowerShell 3.0 or above, and PowerShell Remoting enabled (these are default settings) Also requires 'Set-ExecutionPolicy RemoteSigned' to be set
Reset Windows Scheduled Task Accounts	<ul style="list-style-type: none"> PowerShell 3.0 or above, and PowerShell Remoting enabled (these are default settings)
Reset Window Services Accounts	<ul style="list-style-type: none"> PowerShell 3.0 or above, and PowerShell Remoting enabled (these are default settings)
Reset COM+ Component Passwords	<ul style="list-style-type: none"> PowerShell 3.0 or above PowerShell Remoting enabled

SQL Server, MySQL, Cisco Switches/Routers, Linux/Unix Hosts, HP iLO Cards and VMWare

Script	Requirements
Reset Microsoft SQL Server Accounts	<ul style="list-style-type: none">• Firewall allows access on SQL Server port – default port is 1433 for SQL Standard and above, and SQL Express can use a Dynamic Port – generally 49260• You must also have the TCP/IP Protocol enabled for SQL Server, and this can be done using the SQL Server Configuration Manager Utility, under the section SQL Server Network Configuration -> Protocols for <InstanceName>. Generally, this is not enabled for SQL Server Express• The Privileged Account Credential you are using to perform resets must have the 'ALTER ANY LOGIN' permission as minimum on order to perform resets. The Privileged Account Credential can be either an SQL Account, or an Active Directory Account - if an AD Account, the Username field must be in the format of domain\Username. If no Privileged Account Credential is being used, an SQL Account can change its own password without any special privileges required in SQL Server.


Open Ports Requirements

For a full list of open port requirements for Password Resets, you can refer to section '10. Password Resets' in the following document -

https://www.clickstudios.com.au/downloads/version8/Passwordstate_Open_Port_Requirements.pdf

8 Password Validation Script Requirements

Password Validation (Account Heartbeats) is also achieved using a variety of different PowerShell scripts, and each of the Validations Scripts has the same System Requirements as the equivalent Password Reset Script.

- Validations can also be performed in the User Interface of Passwordstate, either from the 'Actions' dropdown menu for a password record, or when you open the record you will also see the following Heartbeat icon 

9 Password Discovery Script Requirements


The following PowerShell Scripts are provided to help discover Local Admin Accounts on your network, and various 'Windows Resources' – such as Windows Services, IIS Application Pools and Scheduled Tasks, database accounts, network accounts, etc:


- Get-ADAccounts.ps1
- Get-CiscoAccounts.ps1
- Get-Dependencies.ps1
- Get-FortigateAccounts.ps1
- Get-H3CAccounts.ps1
- Get-JunosAccounts.ps1
- Get-LinuxAccounts.ps1
- Get-LocalAdminAccounts.ps1
- Get-MariaDBAccounts.ps1
- Get-MSSQLAccounts.ps1
- Get-MySQLAccounts.ps1
- Get-OracleAccounts.ps1
- Get-PostgreSQLAccounts.ps1
- Get-SonicWallAccounts.ps1


These scripts are located in the folder /setup/scripts, and are imported and encrypted in the Passwordstate database.

The following two Discovery Scripts also require Windows Management Instrumentation (WMI) - TCP/135, to be enabled on the Host – this is WMI executing locally on the Host once PowerShell Remoted in, and not a remote WMI connection from the Passwordstate web server:

- Get-LocalAdminAccounts.ps1 (Discover Windows Local Admin Accounts)
- Get-Dependencies.ps1 (Discover Windows Account Dependencies)

 Note 1: Each of the Discovery Scripts above have the same System Requirements as their respective Password Reset Scripts

 Note 2: For SQL Server account discoveries, the Privileged Account Credential you are using to perform resets must have the 'ALTER ANY LOGIN' permission as minimum. The Privileged Account Credential can be either an SQL Account, or an Active Directory Account - if an AD Account, the Username field must be in the format of domain\Username. Your SQL Server must be configured in mixed-mode authentication in order to discover SQL Accounts.

 Note 3: The Active Directory Accounts Discovery Script (Get-ADAccounts.ps1), the 'Remote Server Administration Tools (RSAT)' for PowerShell must be installed on your Passwordstate web server, or where you have deployed the 'Remote Site Locations Agent'. On Windows Server Operating Systems, you can install this by running the following PowerShell command (run PowerShell as Admin):

```
Add-WindowsFeature RSAT-AD-PowerShell
```

On Windows Desktop Operating Systems, you must manually download and install the RSAT tools from Microsoft's web site – following is a link for Windows 10 <https://www.microsoft.com/en-us/download/details.aspx?id=45520>

10 Enabling PowerShell Remoting per Host

All versions of Windows Desktop Operating Systems, and Windows Server 2008, do not have PowerShell Remoting enabled by default. It can be enabled on each Host individually by following these steps:

- On the destination Host, run PowerShell as an Administrator
- Now type `Enable-PSRemoting -Force`

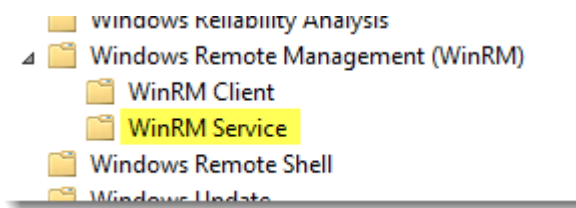
Running this command performs the following:

- Sets the 'Windows Remote Management' service to Automatic (delayed), and starts it
- Enables a HTTP listener
- Adds a firewall exception

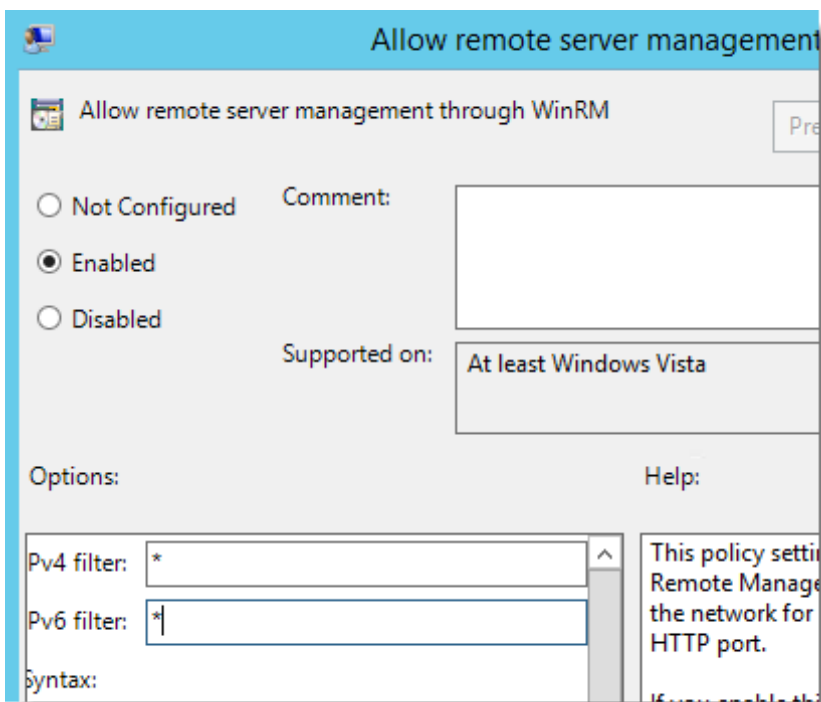
11 Enabling PowerShell Remoting via Group Policy

To enable PowerShell Remoting for multiple hosts at a time in your environment, you can use Group Policy to make the required changes. The following instructions provide detail of how to do this (screenshots here are from a Windows Server 2012 R2 domain controller):

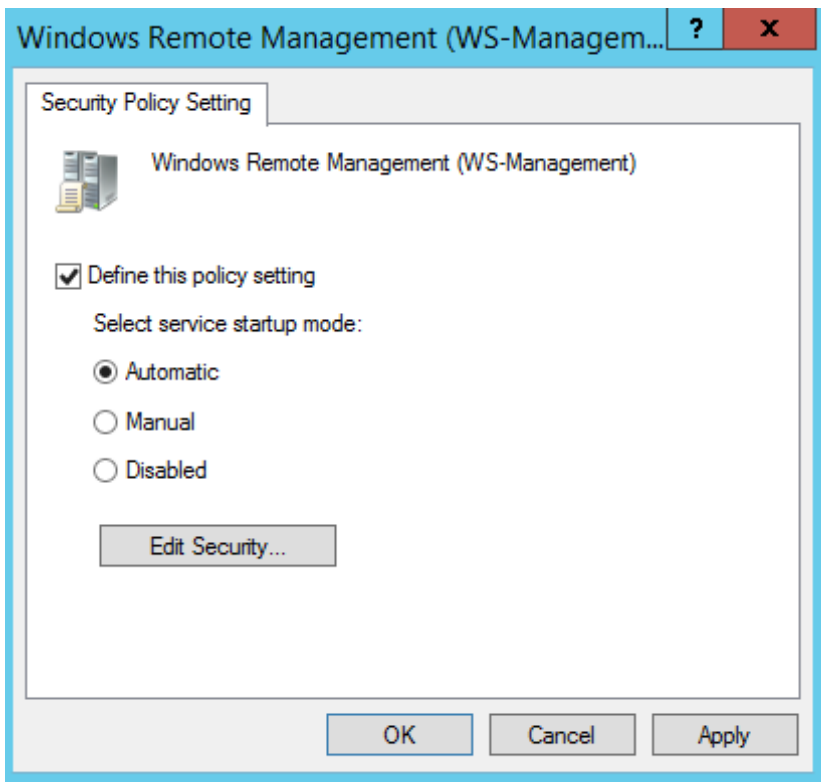
- Open the Group Policy Management Console
- Create or use an existing Group Policy Object, open it, and navigate to Computer Configuration -> Policies -> Administrative templates -> Windows Components
- Here you will find the available Group Policy settings for Windows PowerShell, WinRM and Windows Remote Shell:



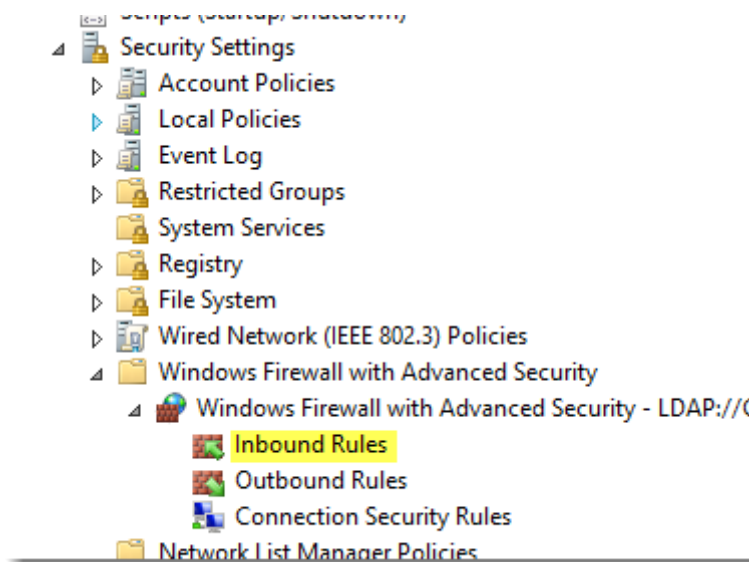
- Open “Allow remote server management through WinRM” setting
- Enable the Policy and set the IPv4 and IPv6 filter values to *



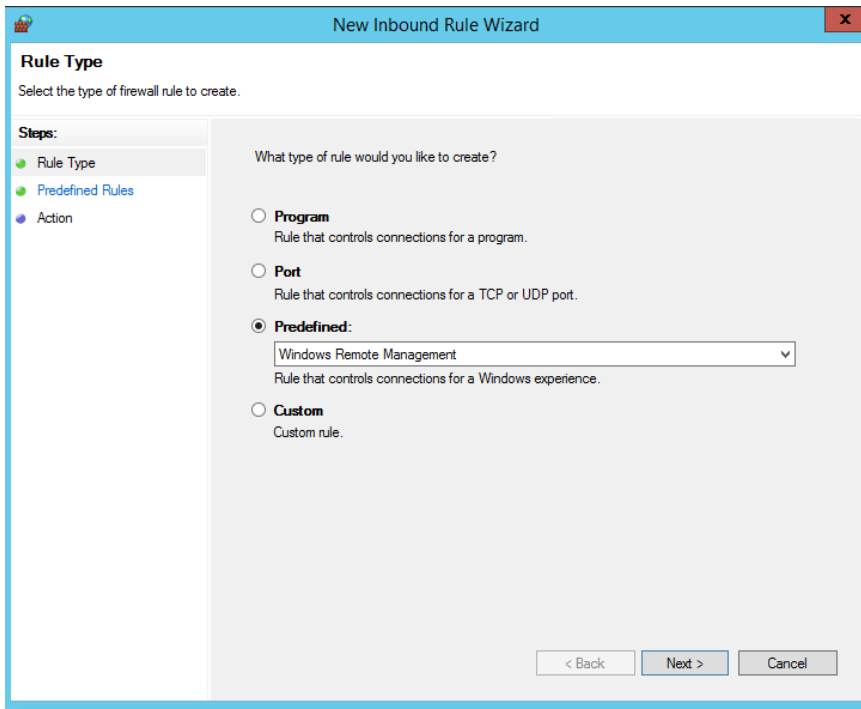
- Click OK
- Navigate to Windows Settings -> Security Settings -> System Services
- Select Windows Remote Management (WS-Management) Service and set the startup mode to Automatic



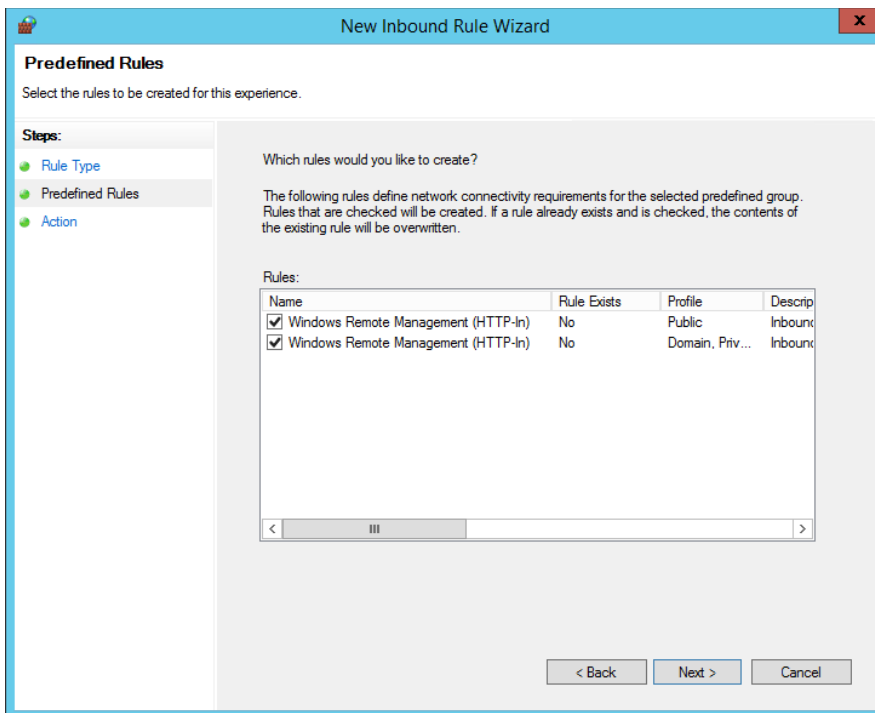
- Click OK
- You need to create a new Inbound Rule under Computer Configuration->Policies->Windows Settings->Windows Firewall with Advanced Security->Windows Firewall with Advanced Security->Inbound Rules:

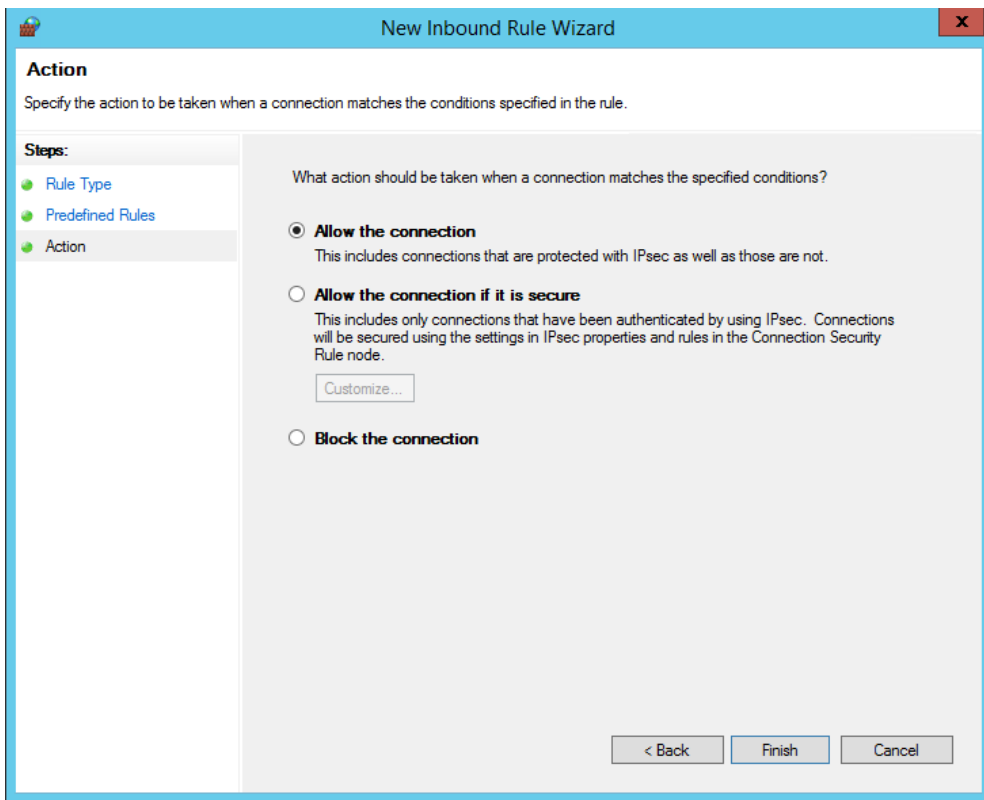


- The WinRM port numbers are predefined as “Windows Remote Management”:



With WinRM 2.0, the default http listener port is TCP 5985.





- Close the Group Policy Editor
- Link the PowerShell Settings GPO to correct OU for testing
- Run gpupdate on your test computers, or reboot them

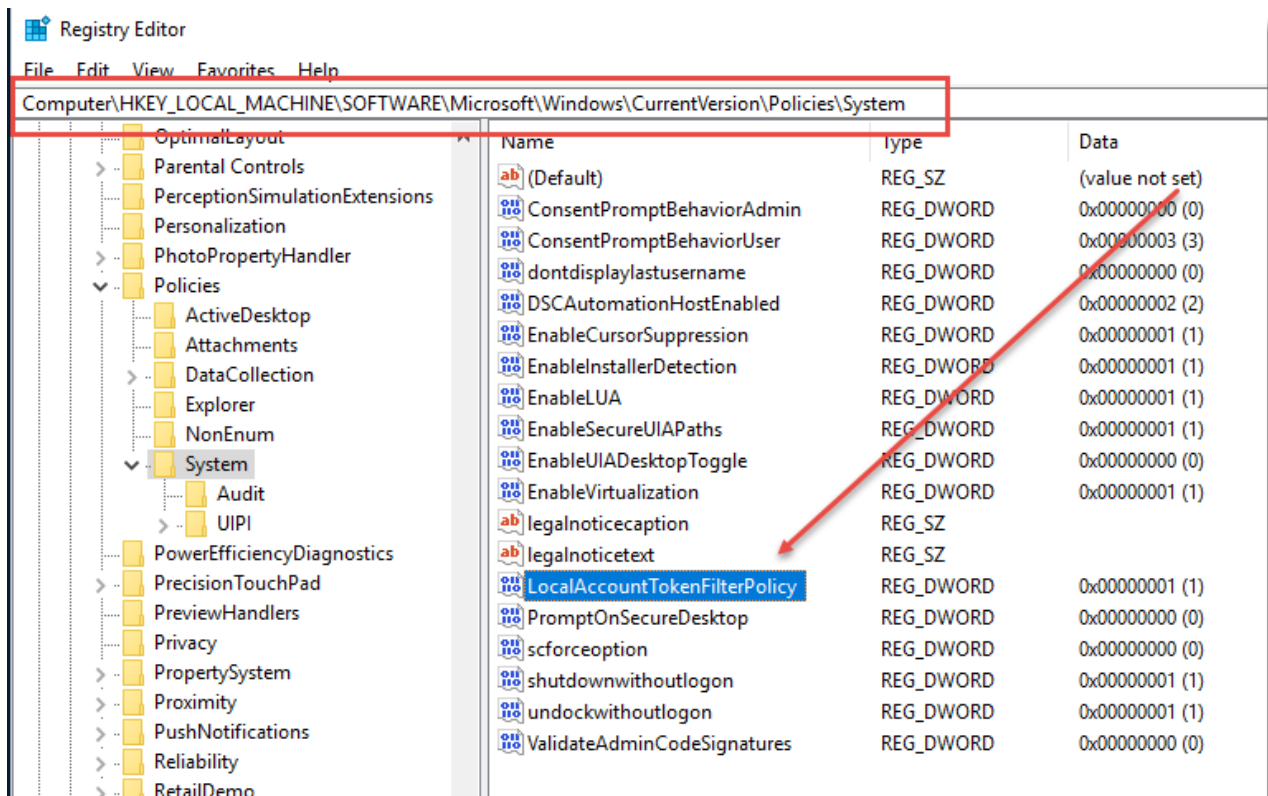
12 Account Discovery and Password Resets between Non-Trusted Active Directory Domains, or against Workgroup Computers

If you are wanting Passwordstate to perform Account Discovery and Password Resets between non-trusted domains, or on computers which are not joined to the domain, you will need to configure PowerShell on your Passwordstate Web Server to “trust” all remote hosts. You can do this by running the following PowerShell command:

```
Set-Item WSMAN:\localhost\Client\TrustedHosts -value *
```

Account Discoveries on Work Group machines will also need to enable the following registry key on the remote host to avoid ‘WinRM’ errors, which are related to UAC blocking Powershell Remoting sessions when used with the Invoke-Command Powershell commandlet, which is what we use to do discoveries.

- Path = HKLM:\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System
- Registry key Name = LocalAccountTokenFilterPolicy
- Type = REG_DWORD
- Data = 1

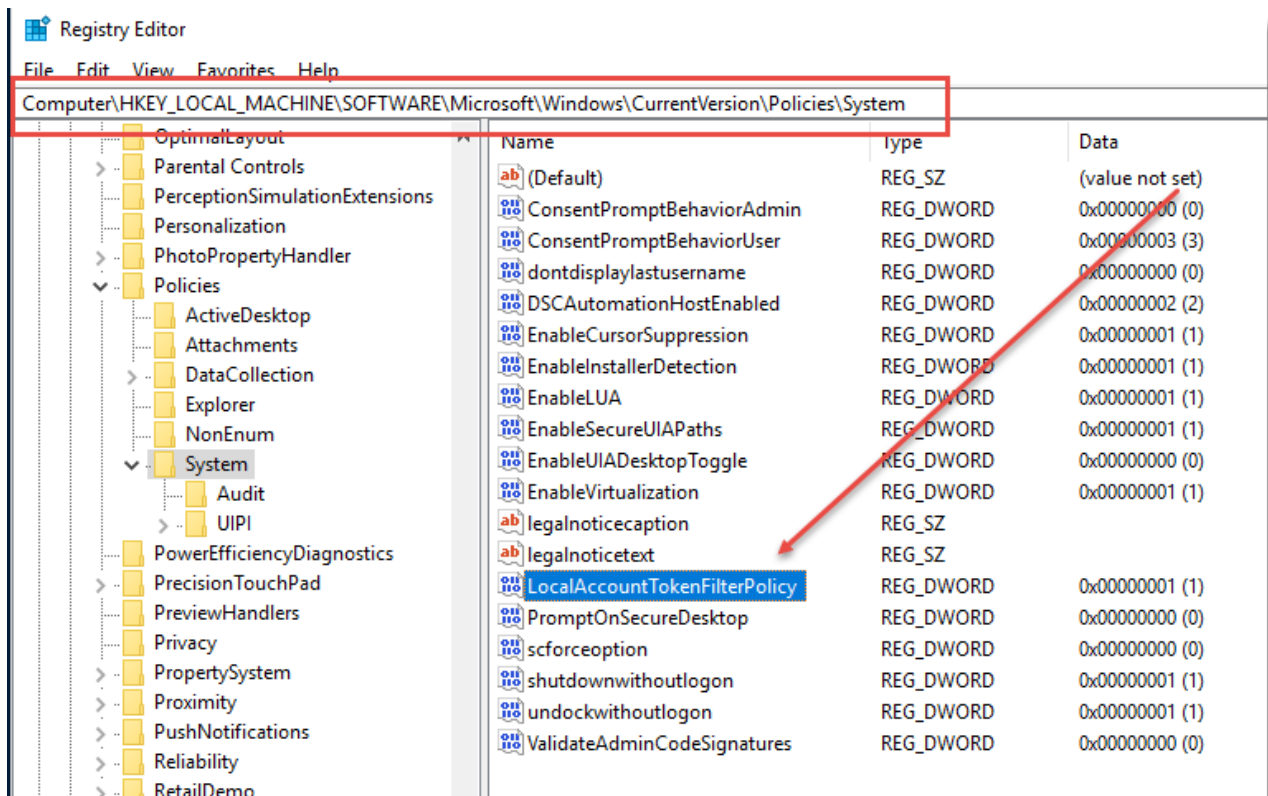


Please restart the Passwordstate Windows Service after making these changes.

13 Local Administrator Account Password Resets Without the Use of a Privileged Account Credential

If you are wanting to perform Password Resets on Windows Local Administrator Accounts, but not associated a Privileged Account Credential with the password record in Passwordstate i.e. reset the password using its own account, then you may need to add/enable the following registry key on the remote host to avoid 'Access Denied' PowerShell Remoting issues.

- Path = HKLM:\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System
- Registry key Name = LocalAccountTokenFilterPolicy
- Type = REG_DWORD
- Data = 1



14 Password Resets and Account Validation for Linux Root Accounts

By default, most Linux Operating Systems do not allow you to SSH in using the root account – for security reasons.

Because of this restriction, it is recommended on the root password record in Passwordstate, that you select a 'Privileged Account Credential' which can SSH into the Linux Host, and perform Password Resets and Account Heartbeats.

In order for this functionality to work, changes are required to each of the Sudoers file on your Linux desktops/servers. Below are the changes required:

- Open the Sudoers file with visudo using the following command:

Sudo visudo -f /etc/sudoers

- When editing the Sudoers file, scroll to the bottom and add the following two lines, entering in the appropriate username you use in Passwordstate as your Privileged Account:

**## Enable sudo rootpw for Passwordstate Privileged Account
Defaults:<username> rootpw**

- 🚩 Please note: If you make this change for the Privileged Account Credential, then only this account can only be used to reset the 'root' account, and no others on that Linux host. If you have other accounts on the Linux host which require password resets, you will need to use a separate Privileged Account Credential which is not configured as per the instructions above.

Edit Password

Please edit the password below, stored within the 'Linux Accounts' Password List (Tree Path = \Infrastructure).

password details notes security **reset options** heartbeat options

Password Reset Script and Privileged Account Credentials

Please select the appropriate Password Reset Script, and Privileged Account Credential, in order to perform the password reset.

Password Reset Script: Reset Linux Password

Privileged Account: msand on Redhat01

🚩 - Active Directory Accounts do not require you to select a Reset Script.
- Not all Reset Scripts require a Privileged Account. See KB Article in menu Help -> User Manual.

Password Reset Schedule

When this Password expires, Auto-Generate a new one and perform any reset tasks at the time of:
00 Hour 00 Minute, and add 90 Days to the Expiry Date

Save Cancel

Edit Password

Please edit the password below, stored within the '**Linux Accounts**' Password List (Tree Path = \Infrastructure).

password details notes security reset options heartbeat options

Heartbeat Validation Options

Select the **Password Validation Script** to use for the Heartbeat verification, and what schedule you would like to use to validate the password is correct:

Validate Password for Linux Account

Use the Privileged Account Credential selected on the 'Reset Options' tab to perform the authentication for this validation (only used for Linux root accounts if required):

Validate Password every day at:

09 Hour 36 Minute

Password Reset tasks will be queued if Password updated. Save Cancel

Please note that for password resets to occur for 'root' accounts, the password value for the root account in Passwordstate must be correct before any resets can occur. This means that if you are using a Linux Account Discovery Job, and a root account is discovered and added into a Password List, then you must edit the password record and make the following changes:

- Untick the option 'Password Enabled for Resets'
- Reset the password to the correct value save the record
- Edit the record again, tick the 'Password Enabled for Resets', and save the record again

Once this is done, schedule and manual password resets can occur for your root accounts.