Monitoring Windows Systems with PowerShell

SL1 version 10.1.4, rev. 2

Table of Contents

Chapter

1

Introduction

Overview

This manual describes how to monitor Windows systems in SL1 using SNMP and PowerShell credentials and Dynamic Applications.

Use the following menu options to navigate the SL1 user interface:

- To view a pop-out list of menu options, click the menu icon (▤).

- To view a page containing all the menu options, click the Advanced menu icon (•••).

The following sections provide an overview of SNMP and PowerShell, as well as the PowerPacks you can use to monitor Windows systems in SL1:

> **NOTE:** ScienceLogic provides this documentation for the convenience of ScienceLogic customers. Some of the configuration information contained herein pertains to third-party vendor software, which is subject to change without notice to ScienceLogic. ScienceLogic makes every attempt to maintain accurate technical information and cannot be held responsible for defects or changes in third-party vendor software. There is no written or implied guarantee that information contained herein will work for all third-party variants. See the End User License Agreement (EULA) for more information.

Monitoring Windows Devices in the ScienceLogic Platform

SL1 can monitor a Windows device using the following methods:

- Requesting information from the Windows SNMP agent
- Requesting information by executing a remote PowerShell command
- Requesting information from the Windows Management Instrumentation (WMI) agent
- Requesting information using the SL1 agent

> **NOTE:** This manual describes how to monitor Windows with SNMP and PowerShell. For more information about using WMI to monitor Windows devices, see the **Monitoring Windows with WMI** manual.

What is SNMP?

SNMP (*Simple Network Management Protocol*) is a set of standard protocols for managing diverse computer hardware and software within a TCP/IP network. SNMP is the most common network protocol used by network monitoring and management applications to exchange management information between devices. SL1 uses this protocol and other protocols to collect availability and performance information.

SNMP uses a server-client structure. Clients are called **agents**. Devices and software that run SNMP are agents. The server is called the **management system**. SL1 is the management system.

Most network hardware is configured for SNMP and can be SNMP-enabled. Many enterprise software applications are also SNMP-compliant. When SNMP is running on a device, it uses a standard format to collect and store data about the device and/or software. For example, SNMP might collect information on each network interface and the traffic for each interface. SL1 can then query the device to retrieve the stored data.

What is PowerShell?

Windows PowerShell is a command-line shell and scripting language for administration of Windows systems. SL1 can execute PowerShell requests on target Windows devices via WinRM (Windows Remote Management). For an overview of Windows PowerShell, see https://docs.microsoft.com/en-us/powershell/scripting/powershell-scripting?view=powershell-6.

SL1 supports the following PowerShell versions for monitoring Windows devices:

- PowerShell 3.0
- PowerShell 4.0
- PowerShell 5.1

PowerPacks

This manual describes content from the following PowerPack versions:

- Microsoft: Active Directory Server, version 101
- Microsoft: DHCP Server, version 1.0
- Microsoft: DNS Server, version 100
- Microsoft: Exchange Server, version 100
- Microsoft: Exchange Server 2010, version 1.2
- Microsoft: Hyper-V Server, version 101
- Microsoft: IIS Server, version 102
- Microsoft: Lync Server 2010, version 1.0
- Microsoft: SharePoint Server, version 1.0
- Microsoft: SQL Server, version 100
- Microsoft: Windows Event Logs, version 100
- Microsoft: Windows Server, version 109
- Microsoft: Windows Server Cluster, version 101
- Microsoft: Windows Server Services, version 101
- ScienceLogic: Concurrent PowerShell Monitoring, version 100

Chapter

2

Configuring Windows Systems for Monitoring with SNMP

Overview

Use the following menu options to navigate the SL1 user interface:

- To view a pop-out list of menu options, click the menu icon ( ).

- To view a page containing all the menu options, click the Advanced menu icon (   ).

The following sections describe how to configure Windows Server 2016, Windows Server 2012, and Windows Server 2008 for monitoring by SL1 using SNMP:

Configuring SNMP for Windows Server 2016 and Windows Server 2012

To configure SNMP on a Windows 2016 Server or a Windows 2012 Server, you must:

1. *Configure "ping" responses*.
2. *Install the SNMP service*.
3. *Configure the SNMP service*.
4. *Configure the firewall to allow SNMP requests*.
5. *Configure Device Classes*. (*Windows Server 2016 only*)

Configuring Ping Responses

For SL1 to discover a device, including SNMP-enabled devices, the device must meet one of the following requirements:

- The device must respond to an ICMP "Ping" request.
- One of the ports selected in the **Detection Method & Port** field for the discovery session must be open on the device. If the *Default Method* option for the **Detection Method & Port** field is selected, SL1 scans TCP ports 21, 22, 23, 25, and 80.

The default configuration for a Windows Server does not allow ICMP "Ping" requests and does not allow connections to TCP ports 21, 22, 23, 25, or 80. Therefore, to discover a Windows Server in SL1, you must perform one of the following tasks:

- Reconfigure the firewall on the Windows Server to allow ICMP "Ping" requests. This section describes how to perform this task.
- Reconfigure the firewall on the Windows Server to allow connections to port 21, 22, 23, 25, or 80. If you have already configured your Windows Server to accept SSH, FTP, Telnet, SMTP, or HTTP connections, this task might have been completed already. You should perform this task only if you were already planning to allow SSH, FTP, Telnet, SMTP, or HTTP connections to your Windows Server.
- When you create the discovery session that will discover the Windows Server, select at least one port in the **Detection Method & Port** field that is open on the Windows Server. For example, if your Windows Server is configured as an MSSQL Server, you could select port 1433 (the default port for MSSQL Server) in the **Detection Method & Port** field.

To reconfigure the firewall on a Windows Server to allow ICMP "Ping" requests, perform the following steps:

1. In the Start menu search bar, enter "firewall" to open a **Windows Firewall with Advanced Security** window.
2. In the left pane, select *Inbound Rules*.
3. If you want SL1 to discover your Windows Server using an IPv4 address, locate the *File and Printer Sharing (Echo Request - ICMPv4-In)* rule.

4. If you want SL1 to discover your Windows Server using an IPv6 address, locate the *File and Printer Sharing (Echo Request - ICMPv6-In)* rule.

5. Right click on the rule that you located, then select *Enable Rule*:

Installing the SNMP Service

To install the SNMP service on a Windows 2012 Server or Windows 2016 Server, perform the following steps:

1. Open the **Server Manager** utility.

2. In the upper-right of the window, select **[Manage]** > *Add Roles and Features*. The **Add Roles and Features** window is displayed.

3. If the server does not skip the **Before you begin** page, click the **[Next >]** button to manually skip it. The **Select installation type** page is displayed:

Configuring SNMP for Windows Server 2016 and Windows Server 2012
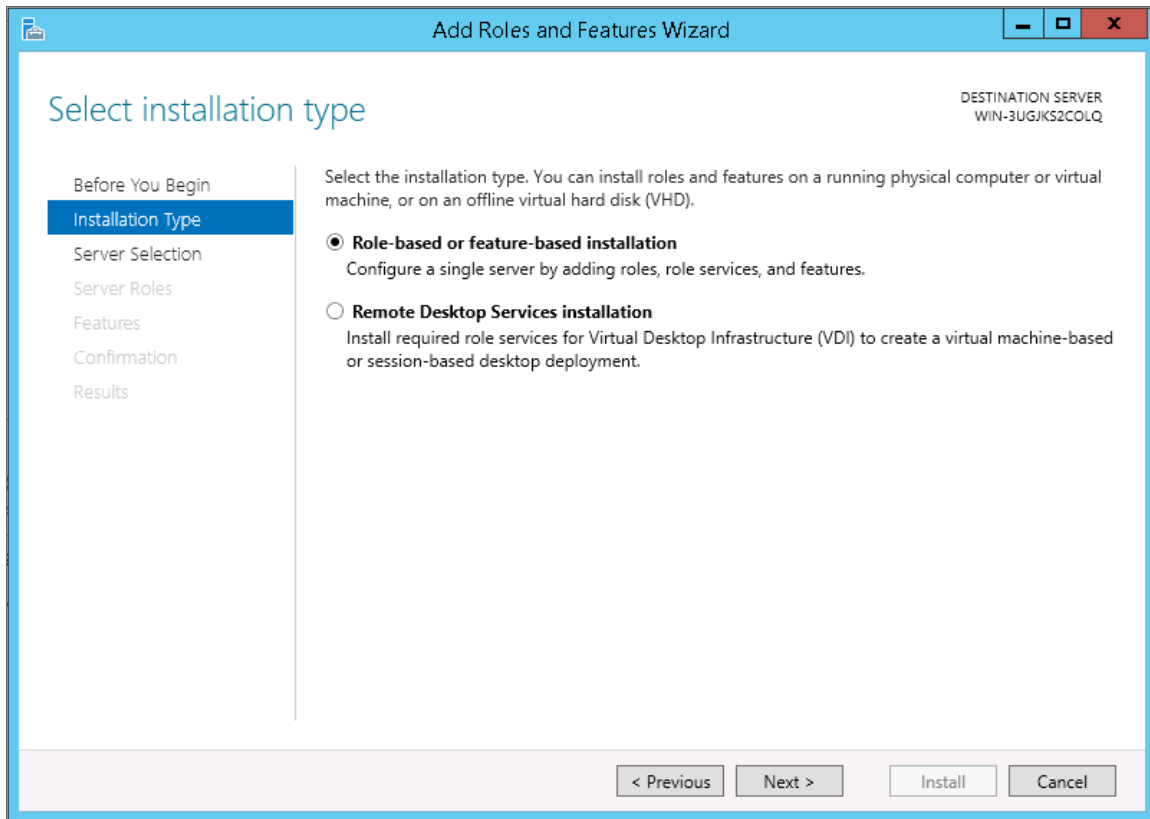
4. Click the [Next >] button to continue with *Role-based or feature-based installation*. The **Select destination server** page is displayed:
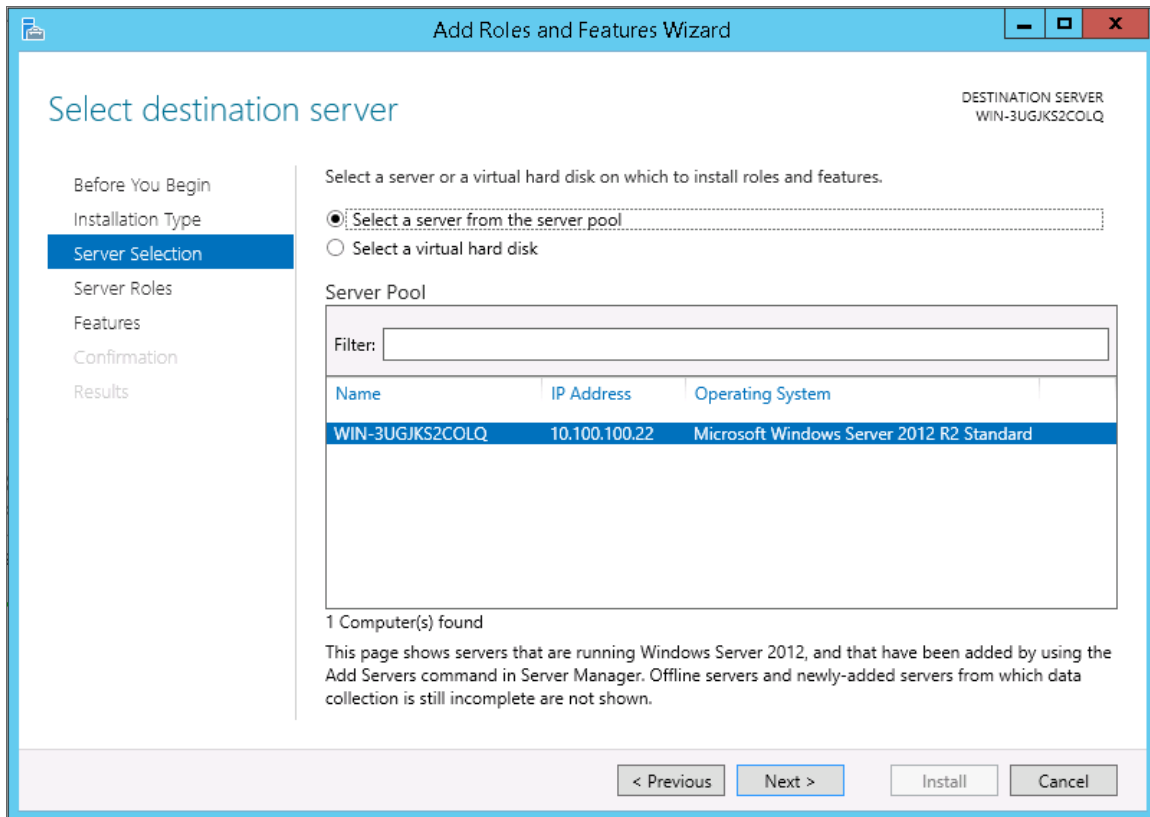
5. Ensure the Windows 2012 server or Windows 2016 Server is selected and then click the **[Next >]** button. The **Select server roles page** is displayed.

6. Click the **[Next >]** button without selecting any additional roles. The **Select features** page is displayed:

Configuring SNMP for Windows Server 2016 and Windows Server 2012
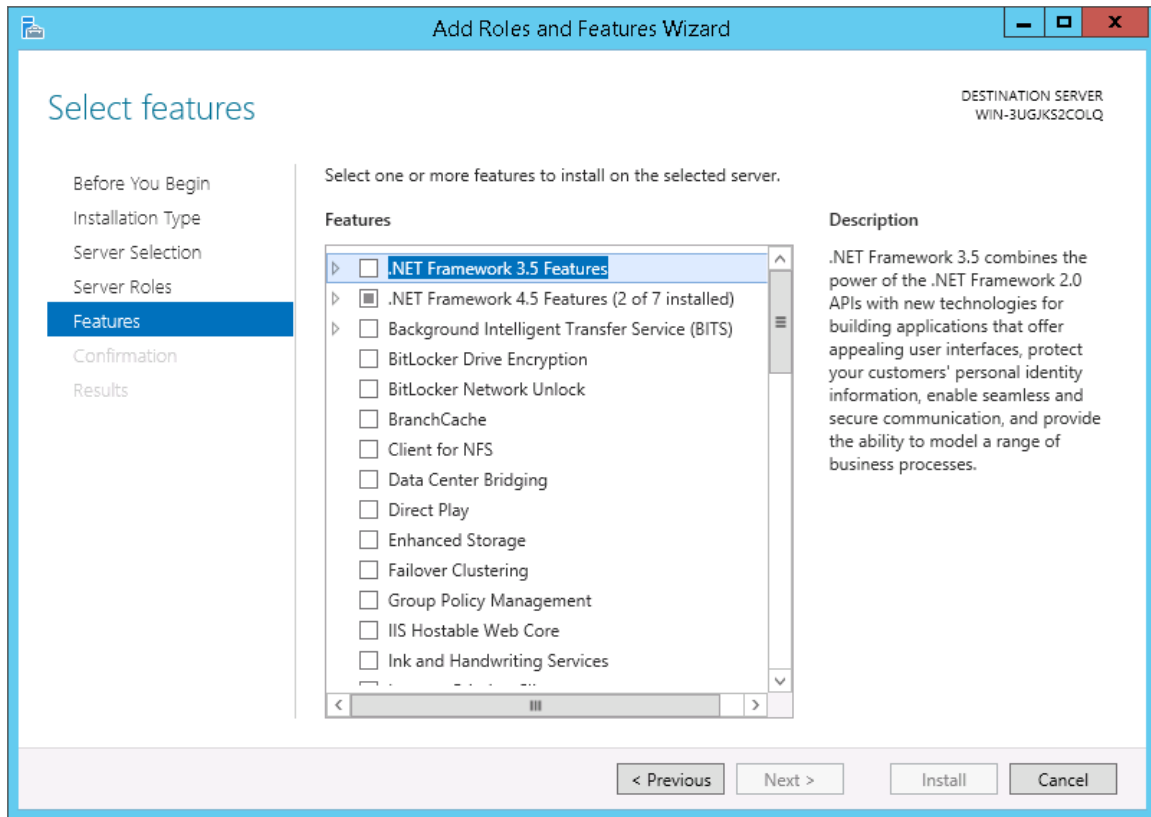
7. Select the *SNMP Service* checkbox. The following confirmation window is displayed:
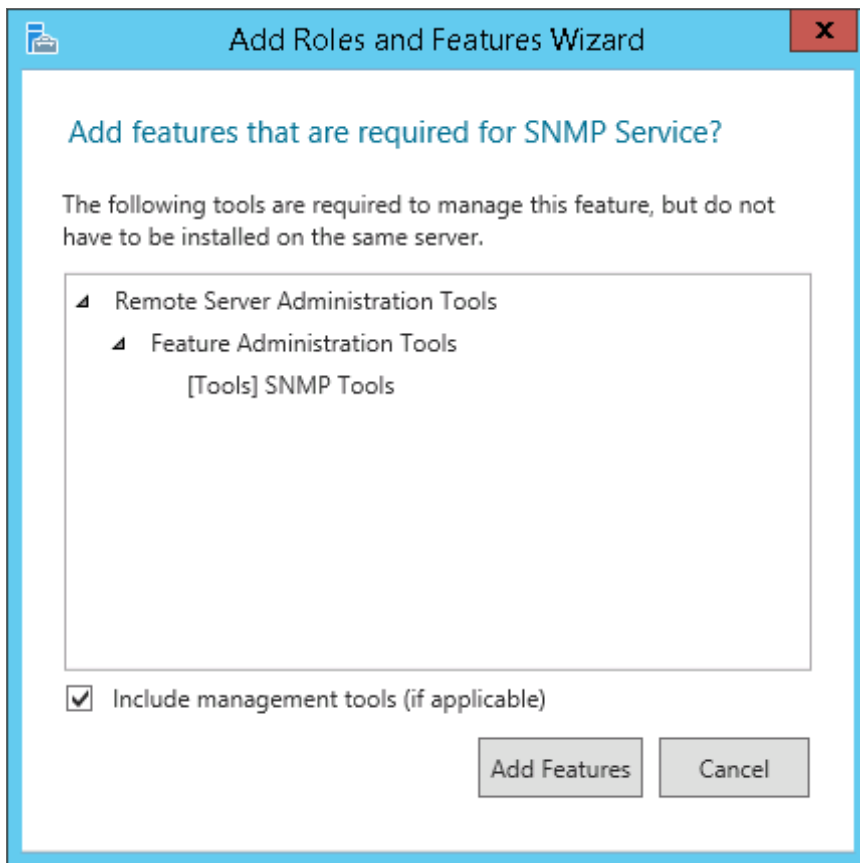
8. Click the **[Add Features]** button.
9. In the Select features page, expand SNMP Service and select the SNMP WMI Provider checkbox.

10. Click the **[Next >]** button. The **Confirm installation selections page** is displayed:
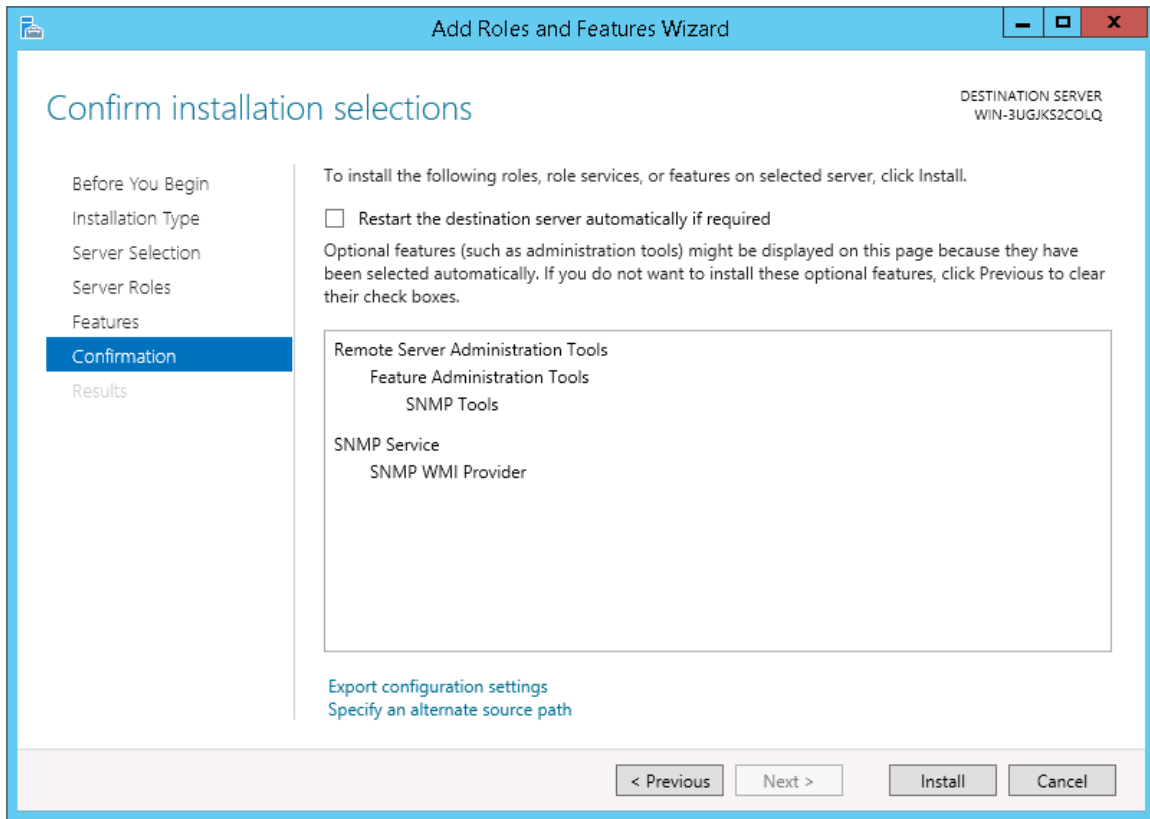
11. Click the **[Install]** button.

12. After the installation is complete, click the **[Close]** button.

Configuring the SNMP Service

To configure the SNMP service on a Windows 2012 Server or Windows 2016 Server, perform the following steps:

> **NOTE:** If you recently installed the SNMP service, you must wait for the **Server Manager** window to refresh to allow the SNMP service snap-in to be added. You can manually refresh the **Server Manager** window by closing the **Server Manager** and then re-opening the **Server Manager**.

1. In the upper-right of the **Server Manager** window, select **[Tools]** > *Services*. The **Services** window is displayed.

Configuring SNMP for Windows Server 2016 and Windows Server 2012

2. In the **Services** window, right-click on *SNMP Service*, and then select *Properties*. The **SNMP Service Properties** window appears:
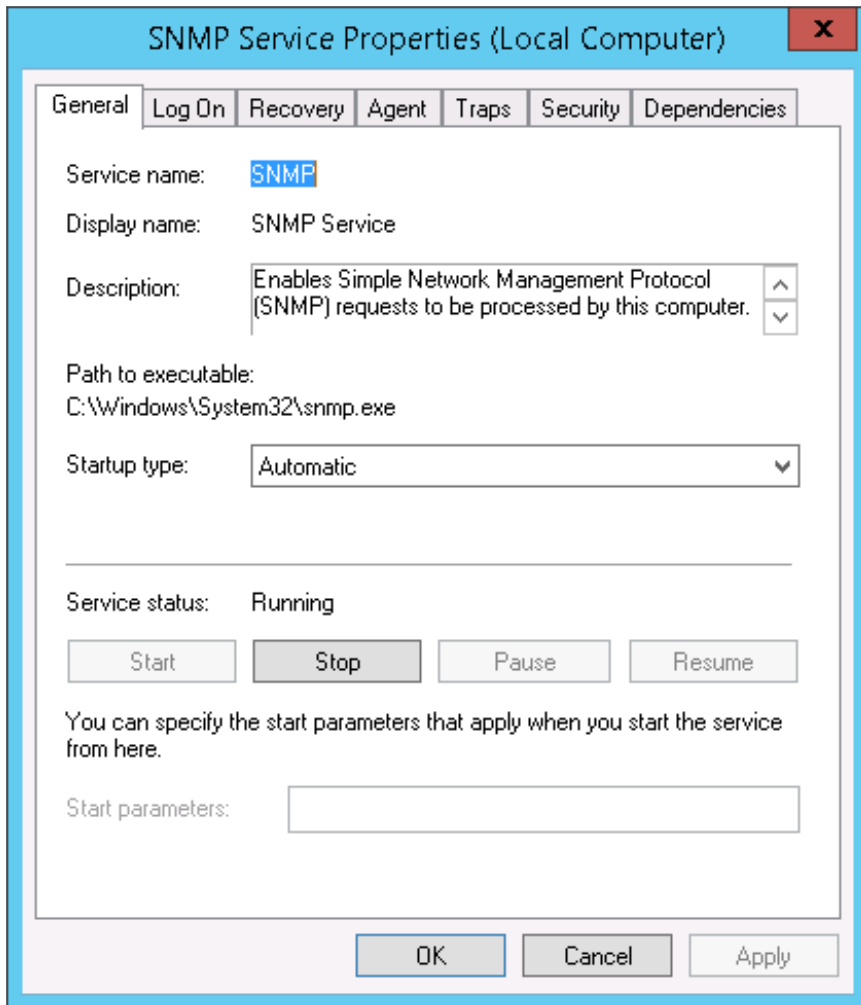
3. In the **Startup type:** field, select *Automatic*.

4. Select the **[Security]** tab. The security settings are displayed:

Configuring SNMP for Windows Server 2016 and Windows Server 2012

5. In the **Accepted community names** panel, click the **[Add…]** button. The **SNMP Service Configuration** pop-up window is displayed:
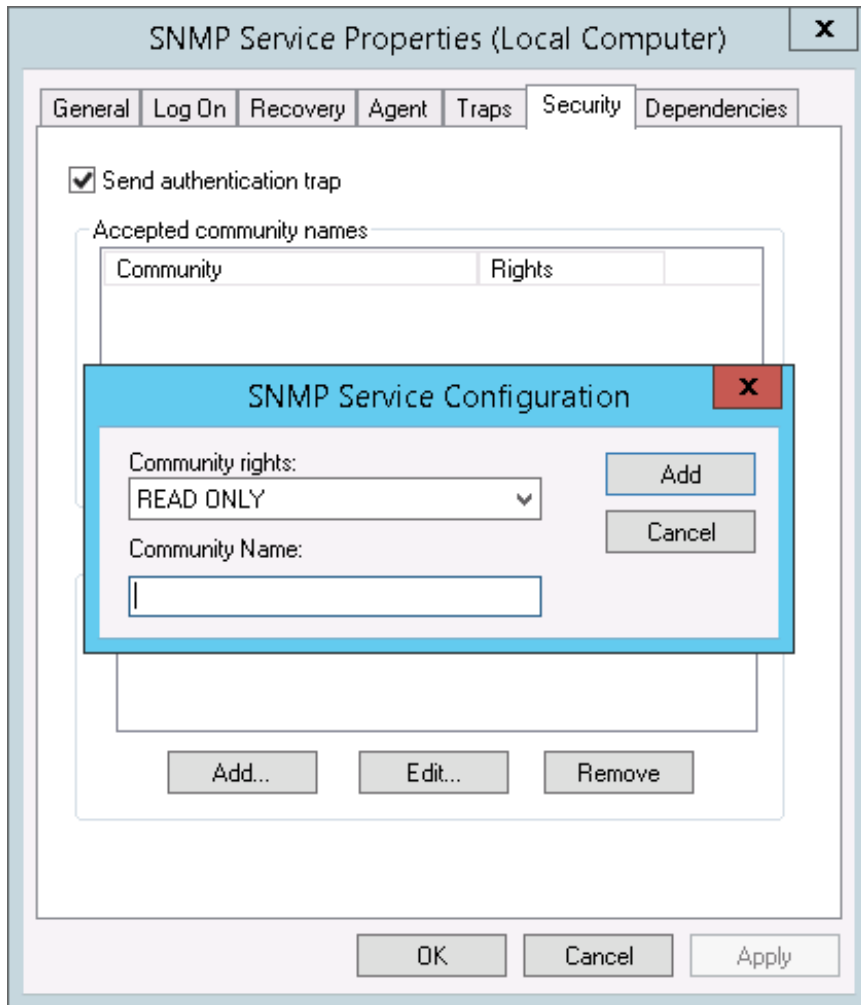
6. Enter a value in the following fields:

- *Community rights*. Select one of the following options from the drop-down list:

  ○ *READ ONLY*. Select this option to allow SL1 to request information from this Windows 2012 Server or Windows 2016 Server using this SNMP community string. This option does not allow SL1 to perform write operations on this Windows 2012 Server or Windows 2016 Server using this SNMP community string.

  ○ *READ WRITE*. Select this option to allow SL1 to request information from this Windows 2008 server and to perform write operations on this Windows 2012 Server or a Windows 2016 Serve using this SNMP community string.

- *Community name*. Enter the SNMP community string that SL1 will use when making SNMP requests to this Windows 2012 Server or Windows 2016 Server. When you create a credential for this Windows 2012 Server or Windows 2016 Server in SL1, you will enter this community string in one the following fields in the **Credential Editor** modal page:

  ○ *SNMP Community (Read-Only)*. Enter the SNMP community string in this field if you selected *READ ONLY* in the **Community rights** drop-down list.

  ○ *SNMP Community (Read/Write)*. Enter the SNMP community string in this field if you selected *READ WRITE* in the **Community rights** drop-down list.

7. Click the **[Add]** button to add the community string to the list of community strings this Windows 2012 Server or Windows 2016 Server accepts.

8. In the **Accept SNMP packets from these hosts** panel, click the **Add...** button. The **SNMP Service Configuration** pop-up window is displayed:

9. In the **Host name, IP or IPX address** field, enter the IP address of the All-In-One Appliance or Data Collector that will monitor this server.

10. Click the **[Add]** button to add the appliance to the list of authorized devices.

11. If you are using SL1 with a distributed architecture, repeat steps 8–10 for each Data Collector in the collector group that will monitor this server.

12. Click the **[Apply]** button to apply all changes.

Configuring the Firewall to Allow SNMP Requests

To configure the Windows Firewall to allow SNMP requests on a Windows 2012 server or Windows 2016 Server, perform the following steps:

1. In the Start menu search bar, enter "firewall" to open a **Windows Firewall with Advanced Security** window.

2. In the left pane, click *Inbound Rules*.

3. Locate the two *SNMP Service (UDP In)* rules.

4. If one or both of the rules is not enabled, right-click on the rule and then select *Enable Rule*:

Configuring Device Classes for Windows Server 2016 and Windows 10

There is a known problem with the Microsoft OID that contains the version number for the operation system. This problem prevents SL1 from using SNMP to automatically align device classes to Windows 10 devices and Microsoft Server 2016 devices.

Because Microsoft has deprecated support of SNMP on Microsoft Server 2016 and Windows 10, users who want to use SNMP to monitor Windows 10 and Microsoft Server 2016 should use one of these workarounds:

- After discovering a Microsoft Server 2016 or Windows 10 device, manually align the device class and disable nightly auto-discovery
- Edit the registry key

Both workarounds are described in the following sections.

Manually Align the Device Class

After discovering Microsoft Server 2016 devices and Windows 10 devices, you can manually align a device class with the discovered devices. To preserve your manual changes, you must disable nightly auto-discovery for those devices. You can manually align the discovered devices with one of these device classes:

- Windows Server 2016
- Windows Server 2016 Domain Controller
- Windows 10 Workstation

For details on manually assigning a device class to a device, follow the steps in the section on *Manually Changing the Device Class for a Device* in the **Device Management** manual chapter on *Managing Device Classes and Device Categories*. For details on disabling nightly auto-discovery for a device, see the section on *Maintaining the New Device Class During Auto-Discovery* in the **Device Management** manual chapter on *Managing Device Classes and Device Categories*.

Edit the Registry Key

You can log in to the device that you want to monitor and manually edit the Windows Registry Key "HKEY_LOCAL_ MACHINE\Software\Microsoft\Windows NT\CurrentVersion". You can define the value CurrentVersion as either "2016" or "10.0". To do this:

1. Click the Start menu and choose Run.
2. In the Run dialog box, type regedit and then click OK.
3. Navigate to HKEY_LOCAL_MACHINE\Software\Microsoft\Windows NT\CurrentVersion
4. In the right pane, double click on the Default key.
5. Enter the appropriate value:

    - For Microsoft Server 2016, change the **Value** to *2016*
    - For Windows 10, change the **Value** to *10.0*

Configuring SNMP for Windows Server 2008

To configure SNMP on a Windows 2008 Server, you must:

1. *Configure "ping" responses*.
2. *Install the SNMP service*.
3. *Configure the SNMP service*.
4. *Configure the firewall to allow SNMP requests*.

Configuring Ping Responses

For SL1 to discover a device, including SNMP-enabled devices, the device must meet one of the following requirements:

- The device must respond to an ICMP "Ping" request.
- One of the ports selected in the **Detection Method & Port** field for the discovery session must be open on the device. If the *Default Method* option for the **Detection Method & Port** field is selected, SL1 scans TCP ports 21, 22, 23, 25, and 80.

The default configuration for a Windows Server does not allow ICMP "Ping" requests and does not allow connections to TCP ports 21, 22, 23, 25, or 80. Therefore, to discover a Windows Server in SL1, you must perform one of the following tasks:

- Reconfigure the firewall on the Windows Server to allow ICMP "Ping" requests. This section describes how to perform this task.
- Reconfigure the firewall on the Windows Server to allow connections to port 21, 22, 23, 25, or 80. If you have already configured your Windows Server to accept SSH, FTP, Telnet, SMTP, or HTTP connections, this task might have been completed already. You should perform this task only if you were already planning to allow SSH, FTP, Telnet, SMTP, or HTTP connections to your Windows Server.
- When you create the discovery session that will discover the Windows Server, select at least one port in the **Detection Method & Port** field that is open on the Windows Server. For example, if your Windows Server is configured as an MSSQL Server, you could select port 1433 (the default port for MSSQL Server) in the **Detection Method & Port** field.

To reconfigure the firewall on a Windows Server to allow ICMP "Ping" requests, perform the following steps:

1. In the Start menu search bar, enter "firewall" to open a **Windows Firewall with Advanced Security** window.
2. In the left pane, select *Inbound Rules*.
3. If you want SL1 to discover your Windows Server using an IPv4 address, locate the *File and Printer Sharing (Echo Request - ICMPv4-In)* rule.
4. If you want SL1 to discover your Windows Server using an IPv6 address, locate the *File and Printer Sharing (Echo Request - ICMPv6-In)* rule.

5. Right click on the rule that you located, then select *Enable Rule*:

Installing the SNMP Service

To install the SNMP service on a Windows 2008 Server, perform the following steps:

1. Open the **Server Manager** utility.

2. In the left pane of the **Server Manager** window, select *Features*. The **Features Summary** is displayed:
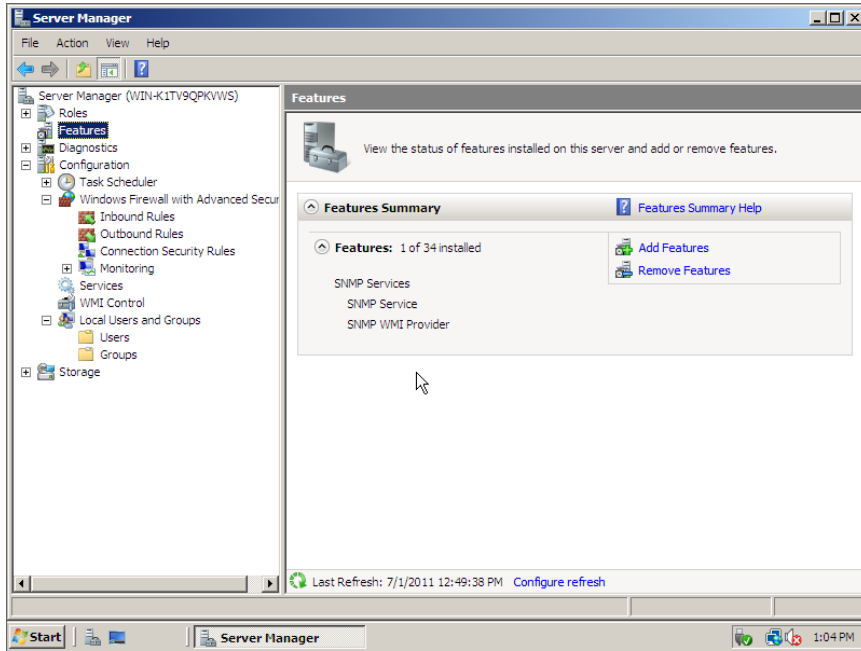
3. If the **Features Summary** displays "SNMP Service" and "SNMP WMI Provider" in the list of installed services (as shown above), you can skip to the section on configuring the SNMP service. If "SNMP Service" and "SNMP WMI Provider" are not included in the list of installed services, select *Add Features*:
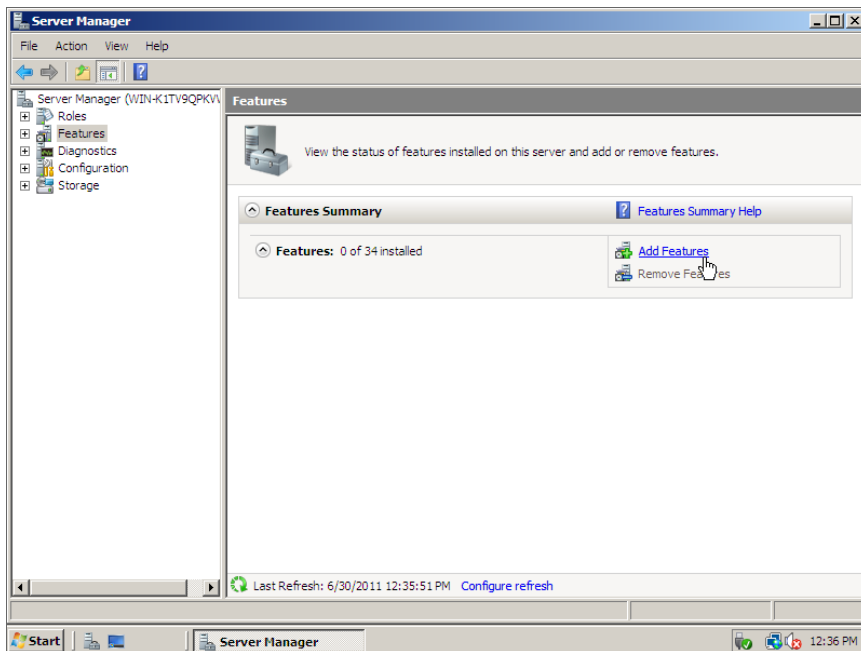
4. In the **Select Features** window, select *SNMP Services*:
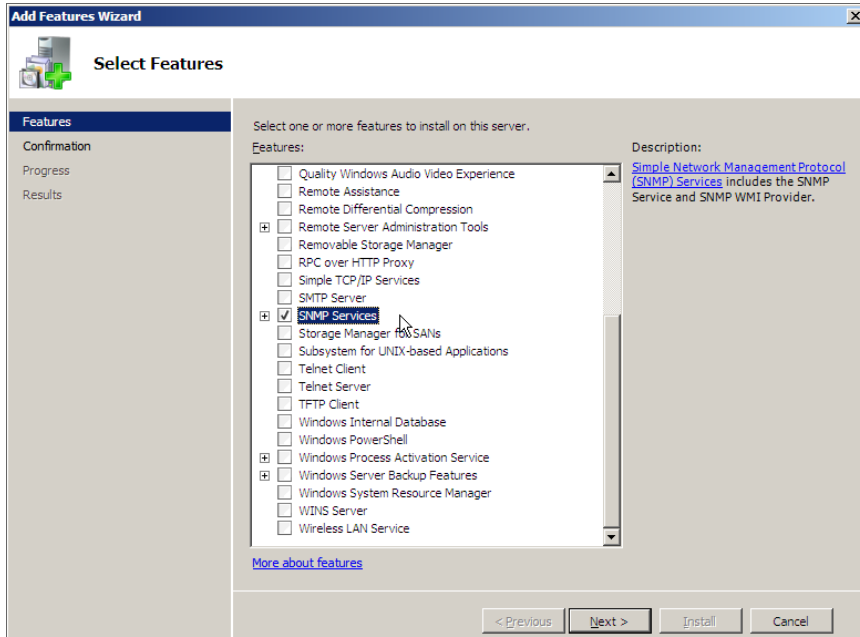
5. Click the **[Next >]** button. The **Confirm Installed Selections** window is displayed with "SNMP Service" and "SNMP WMI Provider" in the list of features that will be installed:

Configuring SNMP for Windows Server 2008
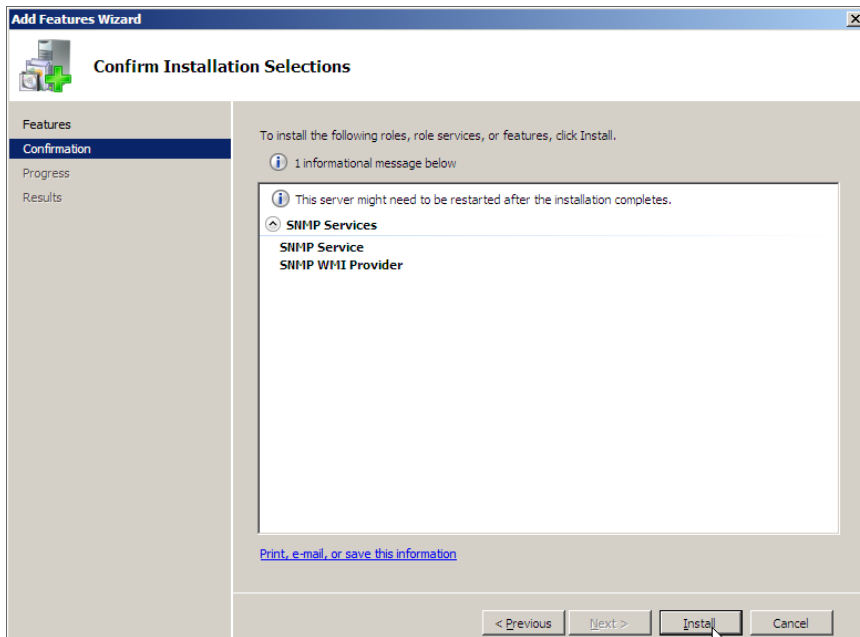
6. Click the **[Install]** button. After the installation is completed, the **Installation Results** window will be displayed:
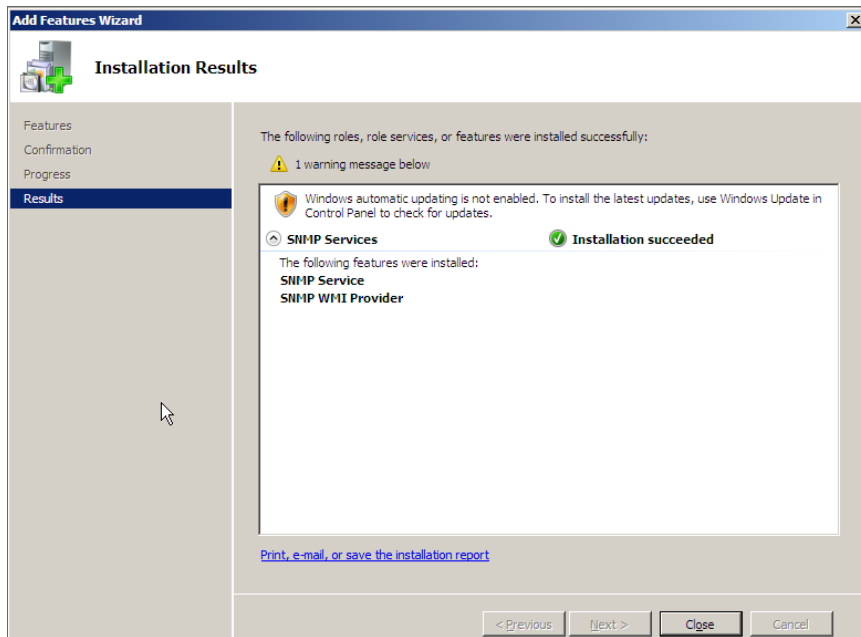
7. Click the **[Close]** button.

Configuring the SNMP Service

To configure the SNMP service on a Windows 2008 Server, perform the following steps:

NOTE: If you recently installed the SNMP service, you must wait for the **Server Manager** window to refresh before it will display the SNMP service snap-in. You can manually refresh the **Server Manager** window by closing the **Server Manager** and then re-opening the **Server Manager**.

1. In the left pane of the **Server Manager** window, expand the *Configuration* section, and then select *Services*.

2. In the list of services, right-click on *SNMP Service*, and then select *Properties*. The **SNMP Service Properties** window appears:

3. In the ***Startup type:*** field, select *Automatic*.

Configuring SNMP for Windows Server 2008

4. Select the **[Security]** tab. The security settings are displayed:

5. In the **Accepted community names** panel, click the **[Add...]** button. The **SNMP Service Configuration** pop-up window is displayed:

6. Enter a value in the following fields:

    - *Community rights*. Select one of the following options from the drop-down list:

        ○ *READ ONLY*. Select this option to allow SL1 to request information from this Windows 2008 Server using this SNMP community string. This option does not allow SL1 to perform write operations on this Windows 2008 Server using this SNMP community string.

        ○ *READ WRITE*. Select this option to allow SL1 to request information from this Windows 2008 server and to perform write operations on this Windows 2008 Server using this SNMP community string.

- *Community name*. Enter the SNMP community string that SL1 will use to make SNMP requests to this Windows 2008 Server. When you create a credential for this Windows 2008 Server in SL1, you will enter this community string in one the following fields in the **Credential Editor** modal page:

  ○ SNMP Community (Read-Only). Enter the SNMP community string in this field if you selected *READ ONLY* in the **Community rights** drop-down list.

  ○ SNMP Community (Read/Write). Enter the SNMP community string in this field if you selected *READ WRITE* in the **Community rights** drop-down list.

7. Click the **[Add]** button to add the community string to list of community strings this Windows 2008 Server accepts.

8. In the **Accept SNMP packets from these hosts** panel, click the **Add...** button. The **SNMP Service Configuration** pop-up window is displayed:

9. In the **Host name, IP or IPX address** field, enter the IP address of the All-In-One Appliance or Data Collector that will monitor this server.

10. Click the **[Add]** button to add the appliance to the list of authorized devices.

11. If you are using SL1 with a distributed architecture, repeat steps 8–10 for each Data Collector in the collector group that will monitor this server.

12. Click the **[Apply]** button to apply all changes.

Configuring the Firewall to Allow SNMP Requests

To configure the Windows Firewall to allow SNMP requests on a Windows 2008 server, perform the following steps:

1. In the Start menu search bar, enter "firewall" to open a **Windows Firewall with Advanced Security** window.

2. In the left pane, click *Inbound Rules*.

3. Locate the two *SNMP Service (UDP In)* rules.

4. If one or both of the rules is not enabled, right-click on the rule and then select *Enable Rule*:

Chapter

3

Configuring Windows Servers for Monitoring with PowerShell

Overview

Use the following menu options to navigate the SL1 user interface:

- To view a pop-out list of menu options, click the menu icon (☰).

- To view a page containing all the menu options, click the Advanced menu icon (⋯).

The following sections describe how to configure Windows Server 2016, 2012, 2012 R2, or 2008 R2 for monitoring by SL1 using PowerShell:

Prerequisites

Before configuring PowerShell, ensure the following:

- Forward and Reverse DNS should be available for the target Windows server from the SL1 Data Collector. Port 53 to the domain's DNS server should thus be available.

- When using an Active Directory user account as the SL1 credential, port 88 on the Windows Domain Controller, for the Active Directory domain, should be open for Kerberos authentication.

- If encrypted communication between the SL1 Data Collector and monitored Windows servers is desired, port 5986 on the Windows server should be open for HTTPS traffic. If unencrypted communications is being used, then port 5985 on the Windows server should be opened for HTTP traffic

- If multiple domains are in use, ensure that they are mapped in the [domain_realm] section of the Kerberos krb5.conf file on the Linux operating system of the SL1 collector appliance.

Configuring PowerShell

To monitor a Windows Server using PowerShell Dynamic Applications, you must configure the Windows Server to allow remote access from SL1. To do so, you must perform the following general steps:

1. *Configure a user account* that SL1 will use to connect to the Windows Server. The user account can either be a local account or an Active Directory account.

> **TIP:** For ease of configuration, ScienceLogic recommends using an Active Directory account that is a member of the local Administrators group on the Windows Server.

2. *Configure a Server Authentication Certificate* to encrypt communication between SL1 and the Windows Server.

3. *Configure Windows Remote Management*.

4. Optionally, *configure a Windows server as a Windows Management Proxy*.

> **NOTE:** If you are configuring multiple Windows servers for monitoring by SL1, you can apply these settings using a Group Policy.

5. Optionally, you can *increase the number of PowerShell Dynamic Applications that can run simultaneously* against a single Windows server.

Step 1: Configuring the User Account for the ScienceLogic Platform

To enable SL1 to monitor Windows servers, you must first configure a user account on a Windows Server that SL1 can use to make PowerShell requests. You will include this user account information when creating the PowerShell credential that SL1 uses to collect data from the Windows Server.

To configure the Windows Server user account that SL1 can use to make PowerShell requests, complete one of the following options:

- **Option 1**: *Create an Active Directory Account with Administrator access*
- **Option 2**: *Create a local user account with Administrator access*
- **Option 3**: *Create a non-administrator user account*

> **TIP:** For ease-of-configuration, ScienceLogic recommends creating an Active Directory user account.

After creating your Windows Server user account, depending on your setup and the servers you want to monitor, you might also need to configure the user account for remote PowerShell access to the following server types:

- *Microsoft Exchange Server*
- *Hyper-V Servers*

Option 1: Creating an Active Directory Account with Administrator Access

For each Windows server that you want to monitor with PowerShell or WinRM, you can create an Active Directory account that is a member of the local Administrators group on each server. For instructions, consult Microsoft's documentation. On Windows Domain Controller servers, you can use a domain account that is not in the Domain Administrators group by following the configuration instructions for *Option 3: Creating a Non-Administrator User Account*.

After creating your Active Directory account:

- If you use SL1 to monitor Microsoft Exchange Servers, you must *configure the user account for remote PowerShell access to Microsoft Exchange Server*.

- If you use SL1 to monitor Hyper-V Servers, you must *configure the user account for remote PowerShell access to the Hyper-V Servers*.

- Otherwise, **you can skip the remainder of this section and** *proceed to Step 3*.

Option 2: Creating a Local User Account with Administrator Access

If you have local Administrator access to the servers you want to monitor and are monitoring Windows Server 2016 or Windows Server 2012, you can alternatively create a local user account with membership in the Administrators group instead of an Active Directory account. For instructions, consult Microsoft's documentation.

> **WARNING:** This method does not work for Windows Server 2008.

After creating your local user account with Local Administrator access:

- If you use SL1 to monitor Microsoft Exchange Servers, you must *configure the user account for remote PowerShell access to Microsoft Exchange Server*.

- If you use SL1 to monitor Hyper-V Servers, you must *configure the user account for remote PowerShell access to the Hyper-V Servers*.

- Otherwise, **you can skip the remainder of this section and** *proceed to Step 2*.

Option 3: Creating a Non-Administrator User Account

If you do not have Local Administrator access to the servers that you want to monitor with PowerShell or WinRM, or if the monitored Windows server is a Domain Controller that will not be in the local Administrators group, then you must first create a domain user account or create a local user account on the Windows Server. For instructions, consult Microsoft's documentation.

After creating your domain user account or local user account:

- You must configure the Windows servers to allow that non-administrator user access. To do so, **follow the steps in this section**.

- If you use SL1 to monitor Microsoft Exchange Servers, you must also *configure the user account for remote PowerShell access to Microsoft Exchange Server*.

- If you use SL1 to monitor Hyper-V Servers, you must also *configure the user account for remote PowerShell access to the Hyper-V Servers*.

To configure Windows Servers to allow access by your non-administrator user account:

1. Start a Windows PowerShell shell with **Run As Administrator** and execute the following command:

```
winrm configsddl default
```

2. On the **Permissions for Default** window, click the **[Add]** button, and then add the non-administrator user account.

3. Select the *Allow* checkbox for the **Read (Get, Enumerate, Subscribe)** and **Execute (Invoke)** permissions for the user, and then click **[OK]**.

4. Access the Management console. To do this:

   - In Windows Server 2008, click **[Start]**, right-click **[Computer]**, click **[Manager]**, and then expand **[Configuration]**.

   - In Windows Server 2016 and 2012, right-click the Windows icon, click **[Computer Management]**, and then expand **[Services and Applications]**.

5. Right-click on **[WMI Control]** and then select *Properties*.

6. On the **WMI Control Properties** window, click the **[Security]** tab, and then click the **[Security]** button.

7. Click the **[Add]** button, and then add the non-administrator user or group in the **Select Users, Service Accounts, or Groups** dialog, then click **[OK]**.

8. On the **Security for Root** window, select the user o group just added, then in the **Permissions** section at the bottom of the window, select the **Allow** checkbox for the *Execute Methods*, *Enable Account*, and *Remote Enable* permissions.

9. Under the **Permissions** section of the **Security for Root** window, click the **[Advanced]** button.

10. In the **Advanced Security Settings** window, double-click on the user account or group you are modifying.

11. On the **Permission Entry** window, in the *Type* field, select *Allow*.

12. In the *Applies to* field, select *This namespace and subnamespaces*.

13. Select the **Execute Methods**, **Enable Account**, and **Remote Enable** permission checkboxes, and then click **[OK]** several times to exit the windows opened for setting WMI permissions.

14. Restart the WMI Service from services.msc.

NOTE: To open services.msc, press the Windows + R keys, type "services.msc", and then press Enter.

---

15. In the Management console, go to System Tools > Local Users and Groups > Groups.

16. Right-click *Performance Monitor Users*, and then select *Properties*.

17. On the **Performance Monitor Users Properties** window, click the **[Add]** button.

18. In the *Enter the object names to select* field, type the non-administrator domain user or group name, and then click **[Check Names]**.

19. Select the user or group name from the list and then click **[OK]**.

20. In the **Performance Monitor Users Properties** window, click **[OK]**.

21. Perform steps 16-20 for the **Event Log Readers** user group and again for the **Distributed COM Users** user group, the R**emote Management Users** user group, and if it exists on the server, the **WinRMRemoteWMIUsers__** user group.

22. If you intend to use encrypted communications between the SL1 collector host and your monitored Windows servers, each Windows server must have a digital certificate installed that has "Server Authentication" as an Extended Key Usage property. You can create a self-signed certificate for WinRM by executing the following command:

```
$Cert = New-SelfSignedCertificate -CertstoreLocation Cert:\LocalMachine\My -DnsName
"myHost"
```

23. Add an HTTPS listener by executing the following command:

```
New-Item -Path WSMan:\LocalHost\Listener -Transport HTTPS -Address * -
CertificateThumbPrint $Cert.Thumbprint –Force
```

NOTE: This command should be entered on a single line.

24. Ensure that your local firewall allows inbound TCP connections on port 5986 if you are going to use encrypted communications between the SL1 collector(s) and the Windows server, or port 5985 if you will be using unencrypted communications between the two. You may have to create a new rule on Windows Firewall if one does not already exist.

Optional: Configuring the User Account for Remote PowerShell Access to Microsoft Exchange Server

If you use SL1 to monitor Microsoft Exchange Servers:

1. Follow the steps in the section *Configuring the User Account for SL1*.

2. Add the new user account to the "Server Management" Exchange security group in Active Directory.

3. The user account will then be able to connect to the relevant WinRM endpoint to use cmdlets installed with the Exchange Management Shell. For example, this will give the user account access to the cmdlet "Get-ExchangeServer".

Optional: Configuring the User Account for Remote PowerShell Access to Hyper-V Servers

To use PowerShell Dynamic Applications to monitor a Hyper-V server, you must:

- Create a user group in Active Directory
- Add the user account you will use to monitor the Hyper-V server to the group
- Set the session configuration parameters on the Hyper-V Server
- Set the group permissions on the Hyper-V Server
- Create a PowerShell credential using the new user account

Creating a User Group and Adding a User in Active Directory

To create a group in Active Directory and add a user:

1. In Active Directory, in the same DC as the Hyper-V host you want to monitor, in the OU called **Users**, create a group. For example, we called our group **PSSession Creators**.

2. Add a user that meets the requirements for monitoring a Windows server via PowerShell to the group. This is the user that you will specify in the PowerShell credential.

> **NOTE**: For details on using Active Directory to perform these tasks, consult Microsoft's documentation.

Setting the Session Configuration Parameters and Group Permissions

To set the Session Configuration and the Group Permissions on the Hyper-V Server:

1. Login to the Hyper-V server.
2. Open a PowerShell session. Enter the following command:

```
Set-PSSessionConfiguration -ShowSecurityDescriptorUI -Name Microsoft.PowerShell
```

3. When prompted, select **A**.
4. The **Permissions** dialog appears.
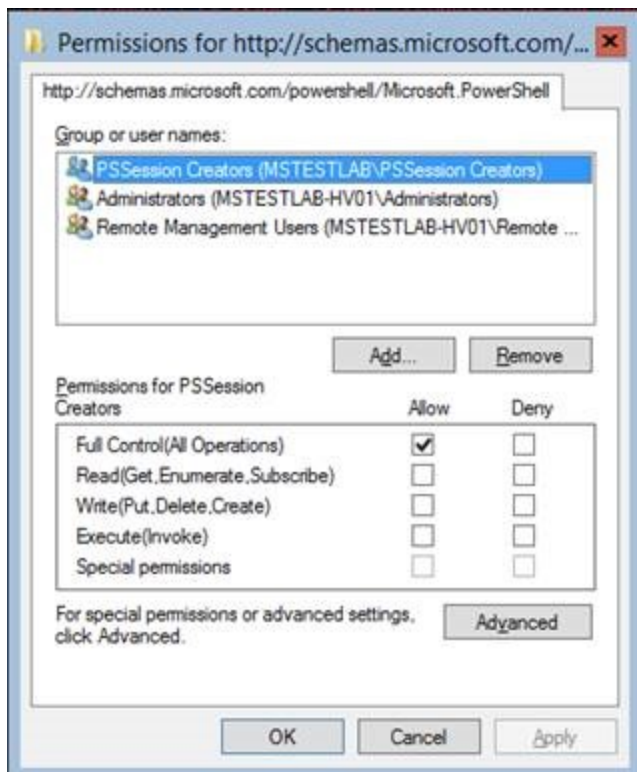
5. In the **Permissions** dialog, supply values in the following fields:

- **Group or user names**. Select the name of the group you created in Active Directory.

- **Permissions for group**. For **Full Control (All Operations)**, select the *Allow* checkbox.

6. Click the **[OK]** button.

Creating a PowerShell Credential

To create a PowerShell credential using the new user account, follow the instructions in the *Creating a PowerShell Credential* section.

Optional: Configuring the User Account for Access to Windows Failover Cluster

To configure Windows Servers to allow access to your Windows Failover Cluster:

1. Start a Windows PowerShell shell with **Run As Administrator** and execute the following command:

    `'Grant-ClusterAccess -User <domain>\<user> -ReadOnly'`

Step 2: Configuring a Server Authentication Certificate

ScienceLogic highly recommends that you encrypt communications between SL1 and the Windows Servers you want it to monitor.

If you have created a **local account on the Windows Server that uses Basic Auth** and that account will allow communication between SL1 and the Windows server, the best practice for security is to enable HTTPS to support encrypted data transfer and authentication. To do this, you must configure WinRM to listen for HTTPS requests. This is called configuring an HTTPS listener.

> NOTE: For details on configuring WinRM on your Windows servers to use HTTPS, see https://support.microsoft.com/en-us/help/2019527/how-to-configure-winrm-for-https.

The sections below describe how to configure a Server Authentication Certificate on the Windows Server. This is only one task included in configuring an HTTPS listener. However, not all users need to configure a Server Authentication Certificate. You can find out if your Windows computer has a digital certificate installed for Server Authentication by running `'Get-ChildItem -Path Cert:\LocalMachine\My -EKU "*Server Authentication*"'` from a PowerShell command shell.

To support encrypted data transfer and authentication between SL1 and the servers, one of the following must be true:

- You have created an **Active Directory** user account on the Windows Server to allow communication between SL1 and the server. In this scenario, Active Directory will use Kerberos and AES-256 encryption to ensure secure data transfer and authentication, which means you do not need to configure a self-signed Server Authentication Certificate. *You can skip this section and* proceed to Step 3.

- You have created a **local account** on the Windows Server that uses Basic Auth to allow communication between SL1 and the server, and your network **includes a Microsoft Certificate server**. In this scenario, you should work with your Microsoft administrator to get a certificate for your Windows Server instead of configuring a self-signed Server Authentication Certificate. *You can skip this section and* proceed to Step 3.

- You have created a **local account** on the Windows Server that uses Basic Auth to allow communication between SL1 and the server, and your network **does not include a Microsoft Certificate server**. In this scenario, you must configure a self-signed Server Authentication Certificate on the Windows Server that you want to monitor with SL1 using one of the following methods:

    - **Option 1**: *Use the Microsoft Management Console*.
    - **Option 2**: If your Windows Server includes Windows Software Development Kit (SDK), you can *use the makecert tool*.
    - **Option 3**: If you are running PowerShell 4.0 or later, you can *use the New-SelfSignedCertificate and Export-PfxCertificate commands*.

NOTE: Self-signed certificates are appropriate for use on a trusted network, such as a LAN that includes both a ScienceLogic Data Collector and the Windows Server to be monitored.

Option 1: Using the Microsoft Management Console to Create a Self-Signed Authentication Certificate

To use the **Microsoft Management Console** to create a self-signed certificate:

1. Log in to the Windows Server that you want to monitor with SL1.
2. In the Start menu search bar, enter "mmc" to open a **Microsoft Management Console** window.

3. Select **[File]**, then *Add/Remove Snap-Ins*. The **Add or Remove Snap-ins** window is displayed:
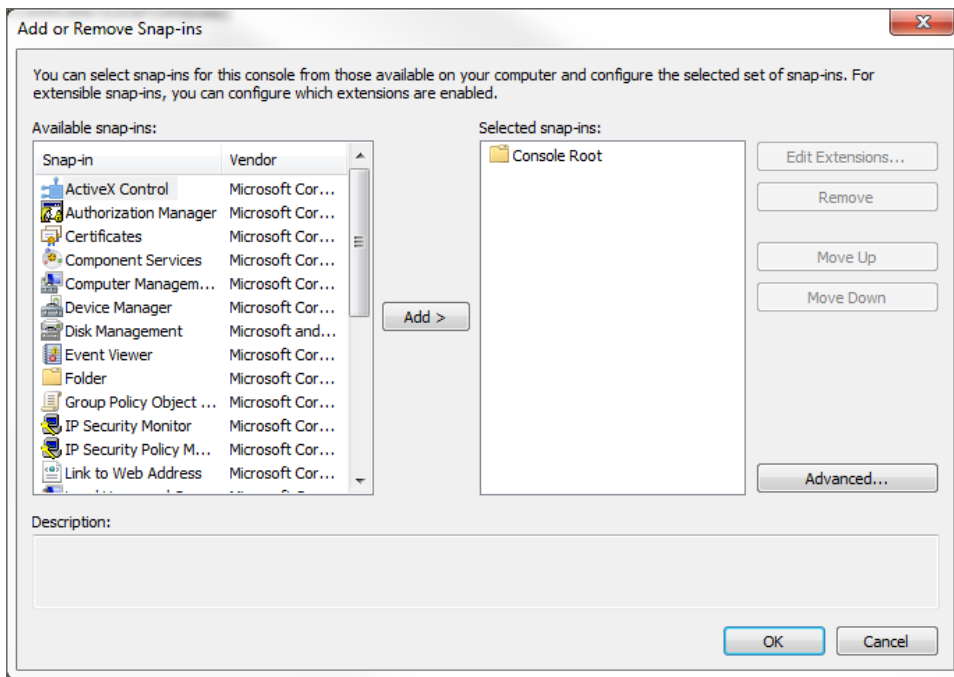
4. In the ***Available snap-ins*** list, select *Certificates*.
5. Click the **[Add >]** button. The **Certificates snap-in** window is displayed:
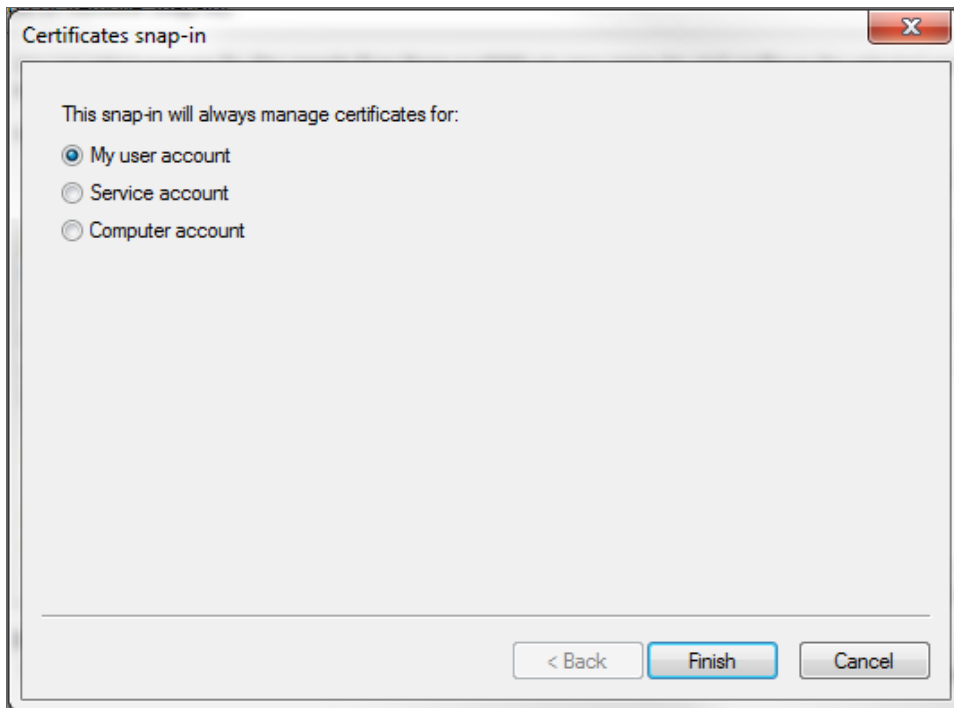
6. Select *Computer account*.

7. Click the **[Next >]** button.

8. Click the **[Finish]** button.

9. In the **Add or Remove Snap-ins** window, click the **[OK]** button.

10. In the left pane of the **Microsoft Management Console** window, navigate to Console Root > Certificates (Local Computer) > Personal.

11. Right-click in the middle pane and select *All Tasks > Request New Certificate….* The **Certificate Enrollment** window is displayed.

12. Click the **[Next]** button. The **Select Certificate Enrollment Policy** page is displayed.

13. Select *Active Directory Enrollment Policy*.

14. Click the **[Next]** button. The **Request Certificates** page is displayed.

15. Select the *Computer* checkbox.

16. Click the **[Enroll]** button.

17. After the certificate is installed, click the **[Finish]** button.

Option 2: Using the MakeCert Tool to Create a Self-Signed Authentication Certificate

If your Windows system includes Windows Software Development Kit (SDK), you can use the MakeCert tool that is included in the kit to create a self-signed certificate.

- For information on the MakeCert tool, see:

    https://msdn.microsoft.com/library/windows/desktop/aa386968.aspx

- For details on creating a self-signed certificate with MakeCert and installing the certificate in the Trusted Root Certification Authorities store, see:

    https://msdn.microsoft.com/en-us/library/ms733813%28v=vs.110%29.aspx

Option 3: Using PowerShell Commands to Create a Self-Signed Authentication Certificate

If your Windows system includes PowerShell 4.0 or later, you can use the following PowerShell commands to create a self-signed certificate:

- You can use the **New-SelfSignCertificate** command to create a self-signed certificate. For information on **New-SelfSignCertificate**, see:

    https://docs.microsoft.com/en-us/powershell/module/pkiclient/new-selfsignedcertificate?view=win10-ps

- You can use the **Export-PfxCertificate** command to export the private certificate. For information on the **Export-PfxCertificate**, see:

Step 3: Configuring Windows Remote Management

To provide SL1 remote access to the Windows Servers you want to monitor, you must configure Windows Remote Management.

> **NOTE:** This step is required regardless of the user account type that SL1 will use to connect to the Windows Server.

There are three ways to configure Windows Remote Management:

- **Option 1**: *Use the script provided by ScienceLogic.*
- **Option 2**: *Manually perform the configuration*.
- **Option 3**: *Use a group policy*.

Option 1: Using a Script to Configure Windows Remote Management

ScienceLogic provides a PowerShell script in a .zip file in the PowerPack download folder that automates configuration of Windows Remote Management and permissions required for the user account that will be used in the SL1 credential. The script configures all of the base Windows permissions required, except for opening up Windows Firewall ports for HTTP and/or HTTPS traffic. The configuration performed by the script is useful primarily for running collection with the **Microsoft: Windows Server**, **Microsoft: Windows Server Services**, **Microsoft: Windows Server Event Logs**, and **Microsoft: SQL Server Enhanced** PowerPacks. (Microsoft: SQL Server Enhanced requires further instance-specific permissions. See the **Monitoring SQL Servers** manual for more information.)

To use the PowerShell script, perform the following steps:

1. When you download the *Microsoft: Windows Server* PowerPack from the ScienceLogic Support site, a .zip file for the **WinRM Configuration Wizard Script (winrm_configuration_wizardv3.0.ps1)** will be in the folder with the PowerPack's EM7PP file.

2. Unzip the downloaded file.

3. Using the credentials for an account that is a member of the Administrator's group, log in to the Windows server you want to monitor. You can log in directly or use Remote Desktop to log in.

4. Copy the PowerShell script named **winrm_configuration_wizardv3.0** to the Windows server that you want to monitor with SL1.

5. Right-click on the PowerShell icon and select **Run As Administrator**.

6. At the PowerShell prompt, navigate to the directory where you copied the PowerShell script named **winrm_configuration_wizardv3.0**.

7. At the PowerShell prompt, enter the following to enable execution of the script:

```
Set-ExecutionPolicy -ExecutionPolicy Unrestricted -Scope Process -Force
```

NOTE: The execution policy setting persists only during the current PowerShell session.

---

8. After the warning text, select Y.

NOTE: If your Windows configuration requires further steps to allow execution of the script, PowerShell will
display prompts. Follow the prompts.

---

9. To run the script with interactive dialogs, enter the following at the PowerShell prompt:

```
.\winrm_configuration_wizardv3.0.ps1 -user <domain>\<username>
```

The user account you wish to use for SL1 collection must be specified with the -user command-line
argument regardless of other arguments used. You can obtain the full help for the PowerShell configuration
script by entering the following:

```
help .\winrm_configuration_wizardv3.0.ps1 -full
```

The most common way to run the script is silently:

```
.\winrm_configuration_wizardv3.0.ps1 -user <domain>\<username> -silent
```

10. If you start the script without using the -silent command-line argument, the **WinRM Installation Wizard**
modal page appears. Click **[OK]**.

WinRM Installation Wizard

This program will configure Windows Remote Management on your Windows
Server, permissions to use WMI, query performance counters and event logs. It
will ask a series of questions to determine the preferred security settings and will
display your choices at the end. Configuration changes will not be made until the
end of the wizard. Before beginning, here are your current settings:

Basic Authentication = True
Kerberos Authentication = True
Allow Unencrypted WinRM Traffic = True
Maximum WinRM Requests = 300
HTTP Port = 5985
HTTPS Port = 5986

NOTE: This wizard cannot override settings applied by Group Policy (GPO). To
overwrite those settings, please contact a system administrator.

Click OK to Continue.

OK   Cancel

11. The **Windows Account Type** modal page appears. Select the appropriate choice for your environment.
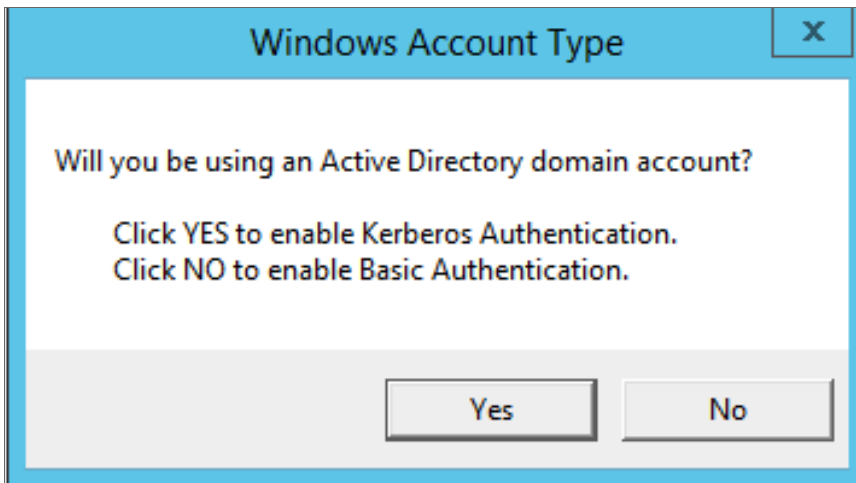
12. The **Set Encryption Policy** modal page appears. Select the appropriate choice for your environment.
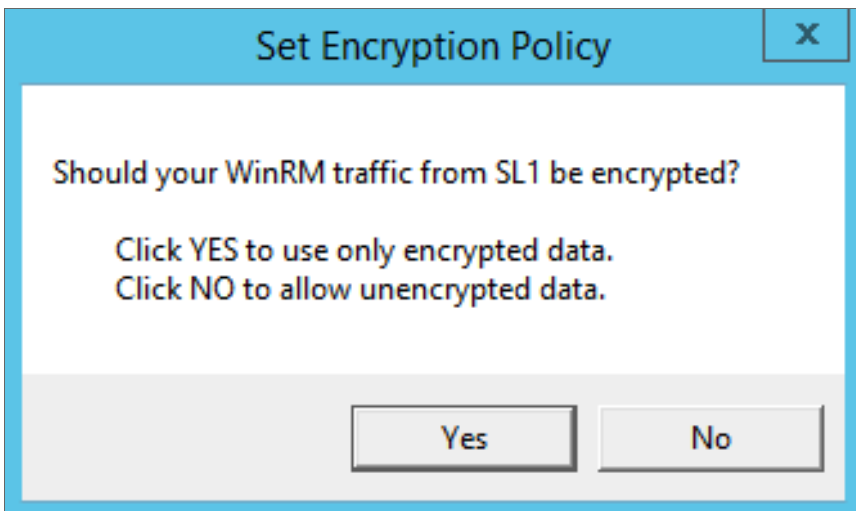
- *Click YES to us only encrypted data*. Click Yes to configure an HTTPS listener for using encrypted communications between the SL1 collectors and the Windows server. Setting up an HTTPS listener requires a digital certificate with Server Authentication EKU to be available on the server. For information on creating a self-signed certificate, see *Configuring a Server Authentication Certificate*.

- *Click NO to allow unencrypted data*. For communication between SL1 collectors and the Windows server, if unencrypted traffic is allowed, an HTTP listener will be configured for communication.
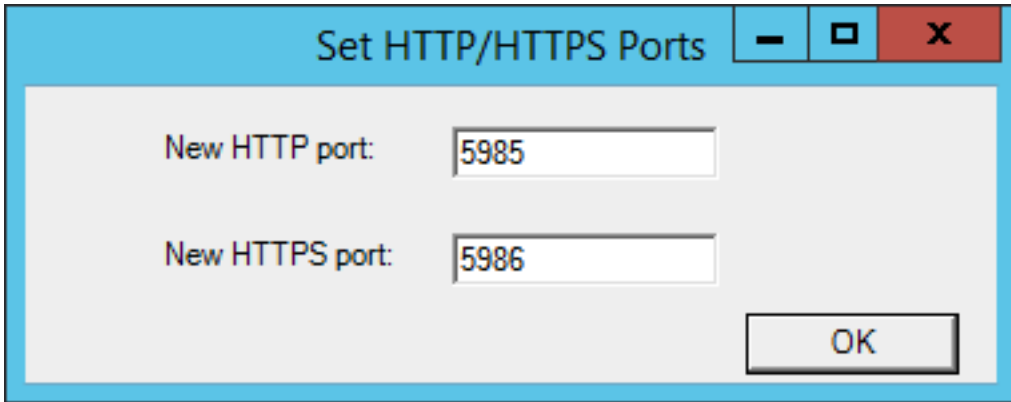
13. The **Change Max Requests** modal page appears. Click **[Yes]**.
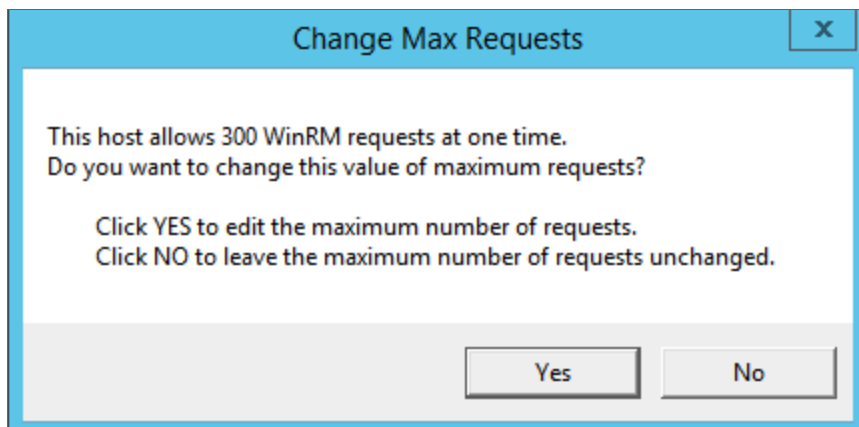
14. The **Set Ports for WinRM Traffic** modal page appears, and it shows the current settings for the HTTP and HTTPS ports. If you want to make a change to these, click **[YES]**; otherwise, click **[NO]** to continue.
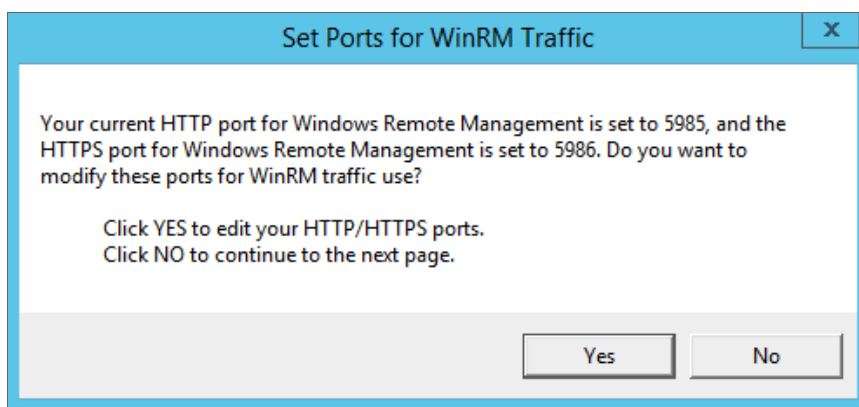
15. Choose which port values you would like SL1 to use when communicating with the Windows server.
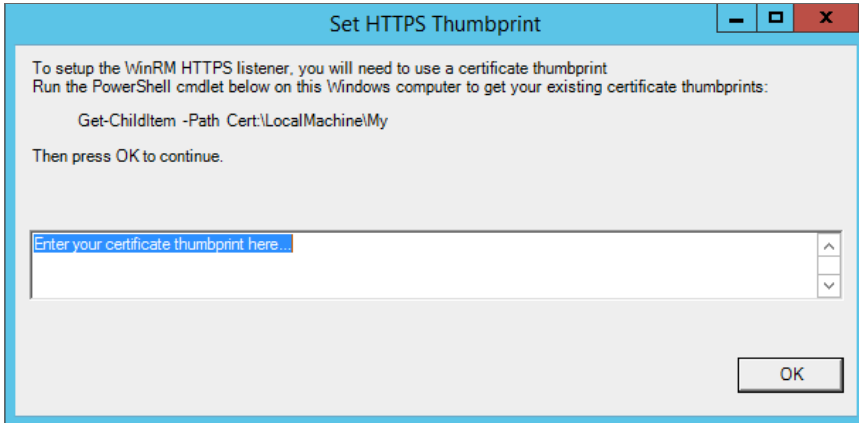
16. The **Set HTTPS Thumbprint** modal page appears. Enter the information for your certificate thumbprint, which is used to create an HTTPS listener, then click **[OK]**.

NOTE: If the certificate structure for your certificate thumbprint is incomplete or incorrect, an error message appears indicating that the WinRM client cannot process the request. If you think you made an error, click **[OK]** and try to correct it. Otherwise, contact a system administrator for help.

---

17. The **Confirm Settings** modal page appears. If the settings are as you specified, click **[OK]**.
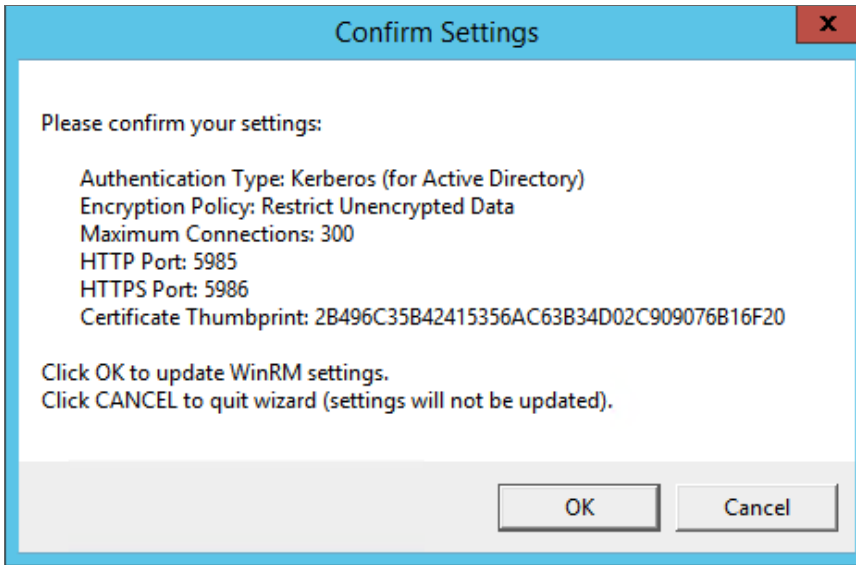
18. The **Complete** modal page appears. If the settings are correct, click **[OK]**.
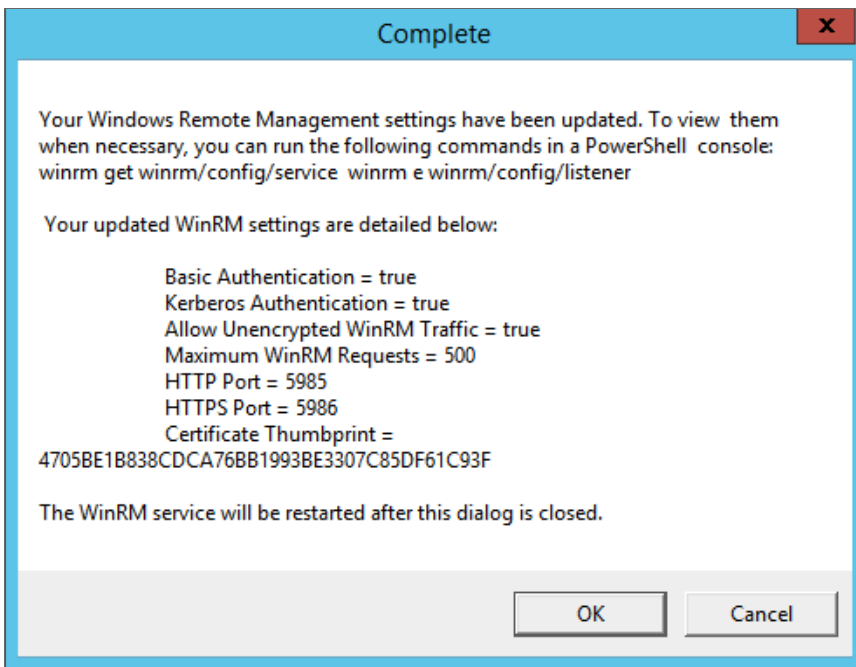
19. Exit the PowerShell session.

Option 2: Manually Configuring Windows Remote Management

To configure a Windows server for monitoring via PowerShell directly, perform the following steps:

1. Log in to the server with an account that is a member of the local Administrators group, or a Domain Administrator's account if on a Windows server with the Domain Controller role installed.

2. Right-click on the PowerShell icon in the taskbar or the **Start** menu, and select *Run as Administrator*.

3. Execute the following command:

   ```
   Get-ExecutionPolicy
   ```

4. If the output is "Restricted", execute the following command:

   ```
   Set-ExecutionPolicy RemoteSigned
   ```

5. Enter "Y" to accept.

6. Execute the following command:

   ```
   winrm quickconfig
   ```

7. Enter "Y" to accept.

8. If you are configuring this Windows server for encrypted communication, execute the following command:

   ```
   winrm quickconfig -transport:https
   ```

9. Enter "Y" to accept.

10. Execute the following command:

    ```
    winrm get winrm/config
    ```

    The output should look like this (additional lines indicated by ellipsis):

    ```
    Config
       ...
       Client
          ...
          Auth
             Basic = true
             ...
             Kerberos = true
             ...
          ...
       Service
          ...
          AllowUnencrypted = false
          ...
          DefaultPorts
             HTTP = 5985
             HTTPS = 5986
          ...
          AllowRemoteAccess = true
       Winrs
          AllowRemoteShellAccess = true
          ...
    ```

11. In the Service section, if the parameter ***AllowRemoteAccess*** is set to *false*, execute the following command:

> **NOTE:** This setting does not appear for all versions of Windows. If this setting does not appear, no action is required.

```
Set-Item WSMan:\Localhost\Service\AllowRemoteAccess -value true
```

12. In the Winrs section, if the parameter **AllowRemoteShellAccess** is set to *false*, execute the following command:

```
Set-Item WSMan:\Localhost\Winrs\AllowRemoteShellAccess -value true
```

13. If you are configuring this Windows server for unencrypted communication and the parameter **AllowUnencrypted** (in the Service section) is set to *false*, execute the following command:

```
Set-Item WSMan:\Localhost\Service\AllowUnencrypted -value true
```

14. If you are configuring this Windows server for unencrypted communication, verify that "HTTP = 5985" appears in the DefaultPorts section.

> **NOTE:** ScienceLogic recommends using encrypted communication, particularly if you are also using an Active Directory account. Using an Active Directory account for encrypted authentication enables you to use Kerberos ticketing for authentication.

15. If you are configuring this Windows server for encrypted communication, verify that "HTTPS = 5986" appears in the DefaultPorts section.

16. If you are using an Active Directory account to communicate with this Windows server and in the Auth section, the parameter **Kerberos** is set to *false*, execute the following command:

```
Set-Item WSMan:\Localhost\Service\Auth\Kerberos -value true
```

> **NOTE:** ScienceLogic recommends using an Active Directory account.

17. If you are using a local account to communicate with this Windows server and in the Auth section, the parameter **Basic** is set to *false*, execute the following command:

```
Set-Item WSMan:\Localhost\Service\Auth\Basic -value true
```

Option 3: Using a Group Policy to Configure Windows Remote Management

You can use a group policy object (GPO) to configure the following Windows Remote Management settings on Windows Server 2012 or Windows Server 2016:
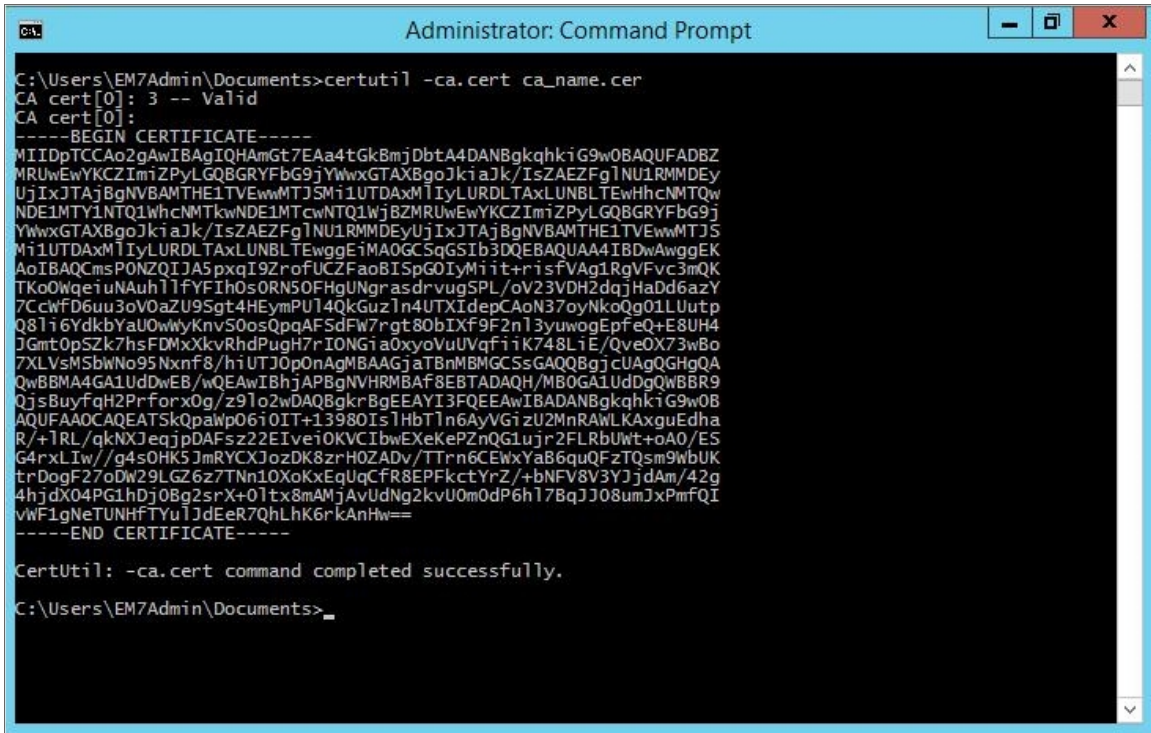
- A registry key to enable Local Account access to Windows Remote Management
- Firewall rules
- Certificates
- HTTP and HTTPS listeners, including authentication and encryption settings
- Service start and recovery settings

To create the group policy object, perform the following steps:

1. Log in to the server as an administrator.
2. Right-click on the PowerShell icon in the taskbar and select *Run as Administrator*.
3. At the PowerShell prompt, use the change directory (CD) command to navigate to a folder where you can create new files.

4. Save the root Certification Authority certificate to the local directory by executing the following command:

```
certutil.exe -ca.cert ca_name.cer
```


TIP: You will import this certificate into the new group policy in step 21.

5. Exit the command prompt.

6. Log in to a domain controller in your Active Directory forest and navigate to the System Manager dashboard.

7. Click the **Tools** menu, then select *Group Policy Management*.
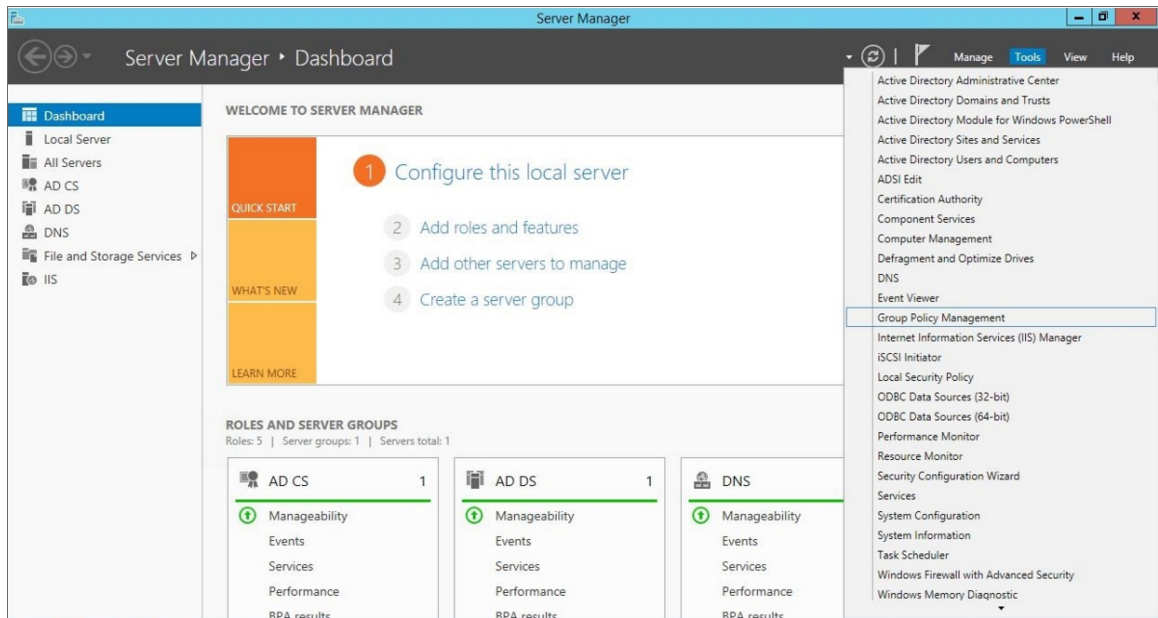
8. On the **Group Policy Management** page, in the left panel, right-click the domain name where you want the new group policy to resideand then select *Create a GPO in this domain and Link it here*.
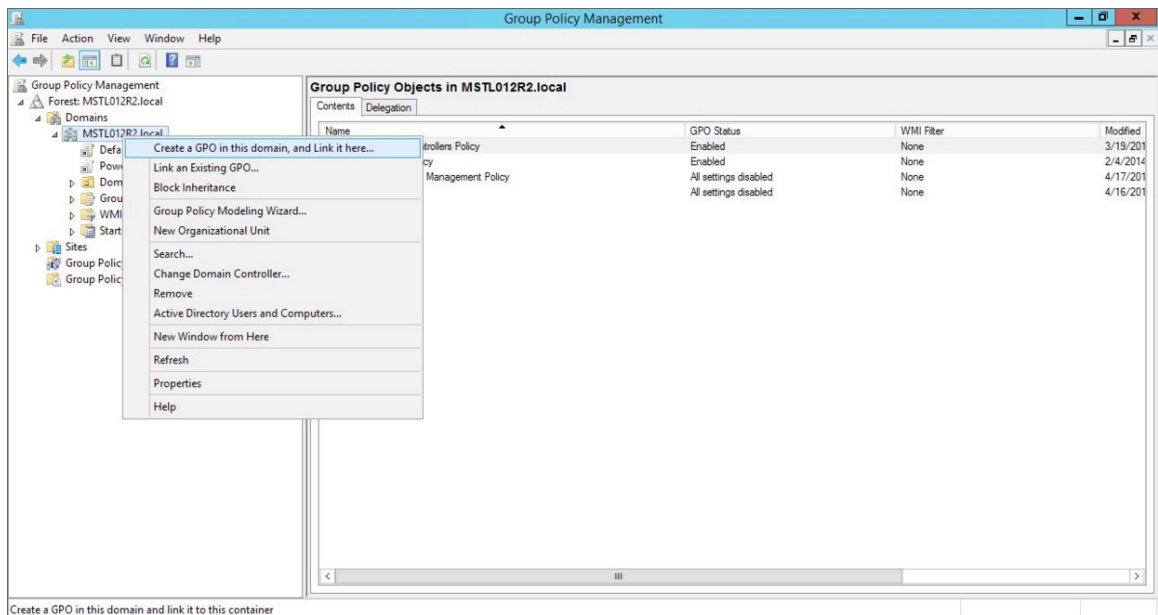
9. In the left panel, right-click the new group policy and select *Edit*. The **Group Policy Management Editor** page for the new Windows Remote Management group policy appears.
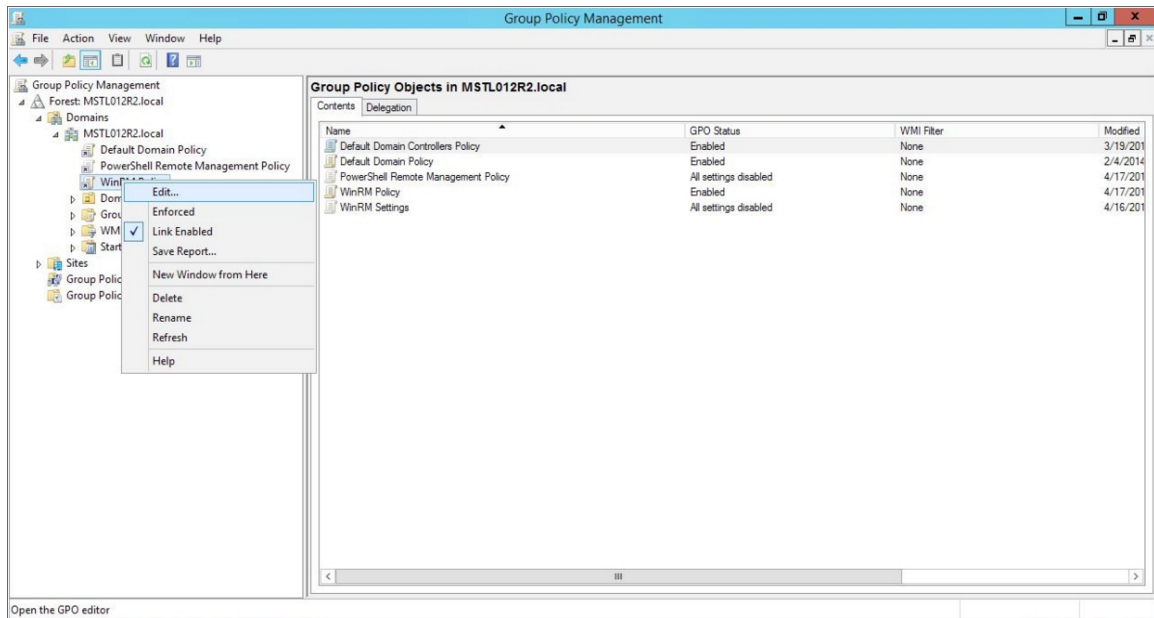
10. In the left panel, navigate to **Computer Configuration > Policies > Windows Settings > Security Settings > System Services**. In the right panel, locate the **Windows Remote Management (WS-Management)** service. Right-click the service, then select *Properties*.
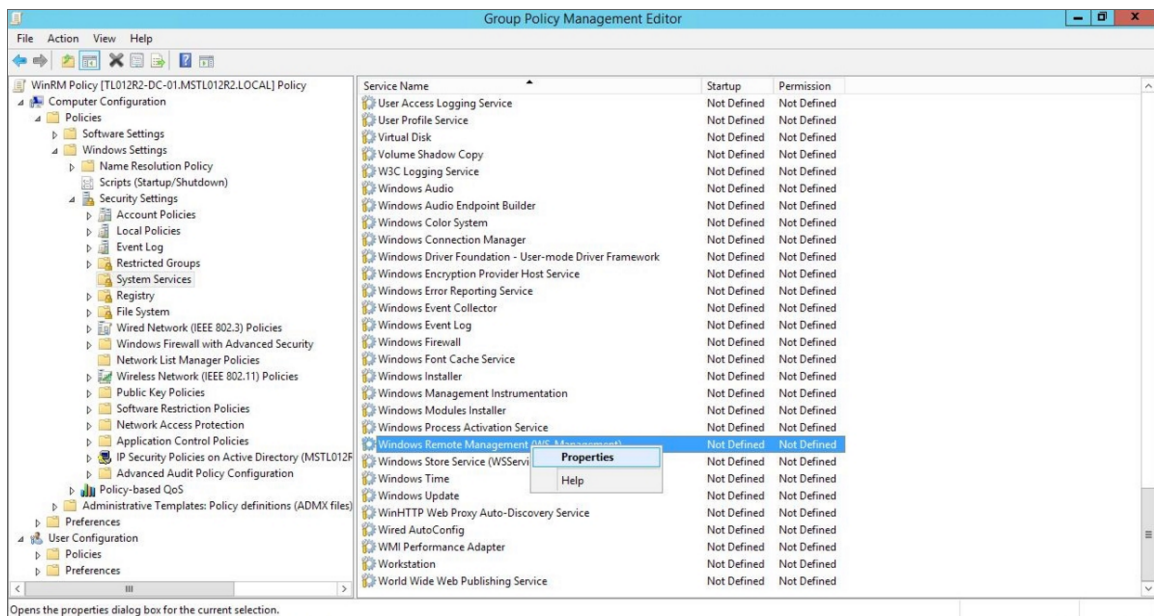
11. The **Windows Remote Management (WS-Management)** modal page appears. Select the **Define this policy setting** check box and the **Automatic** radio button, then click **[OK]**.
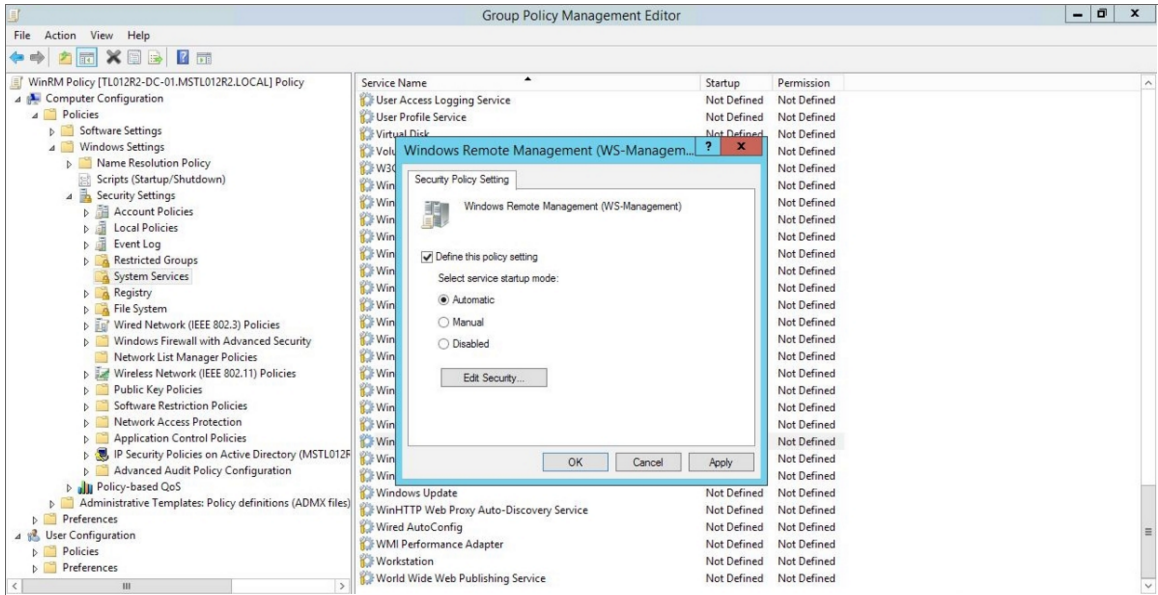
12. In the left panel of the **Group Policy Management Editor** page, navigate to **Computer Configuration > Policies > Windows Settings > Security Settings > Windows Firewall with Advanced Security > Windows Firewall with Advanced Security - LDAP > Inbound Rules**. In the right panel, right-click and select *New Rule*.
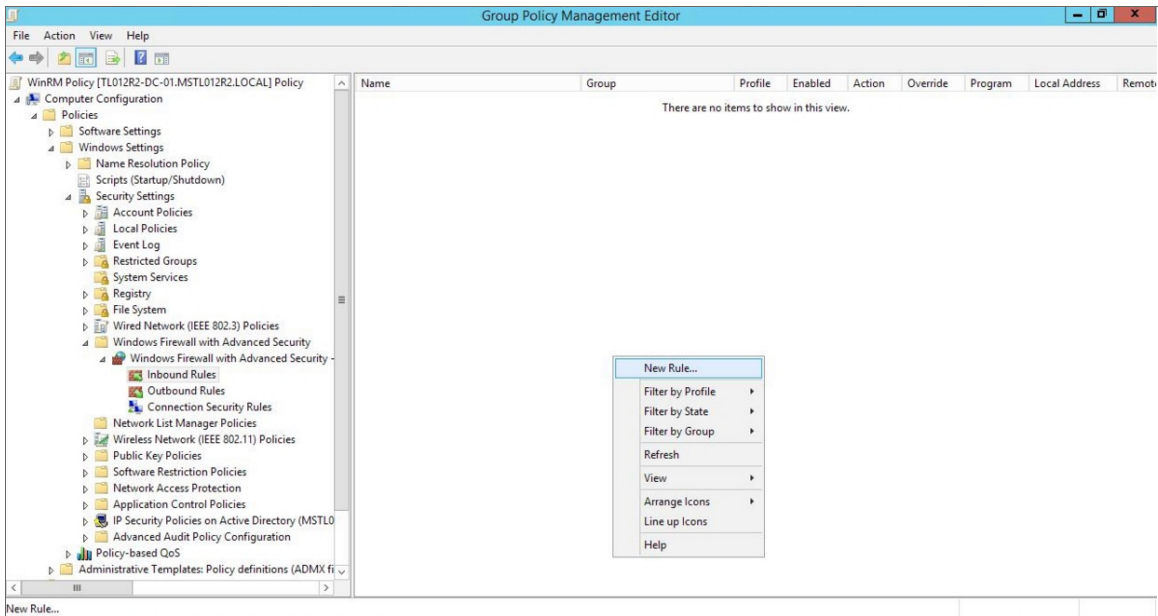
13. The **New Inbound Rule Wizard** modal page appears. Click the **Predefined** radio button, select *Windows Firewall Remote Management* from the list, and then click **[Next]**.
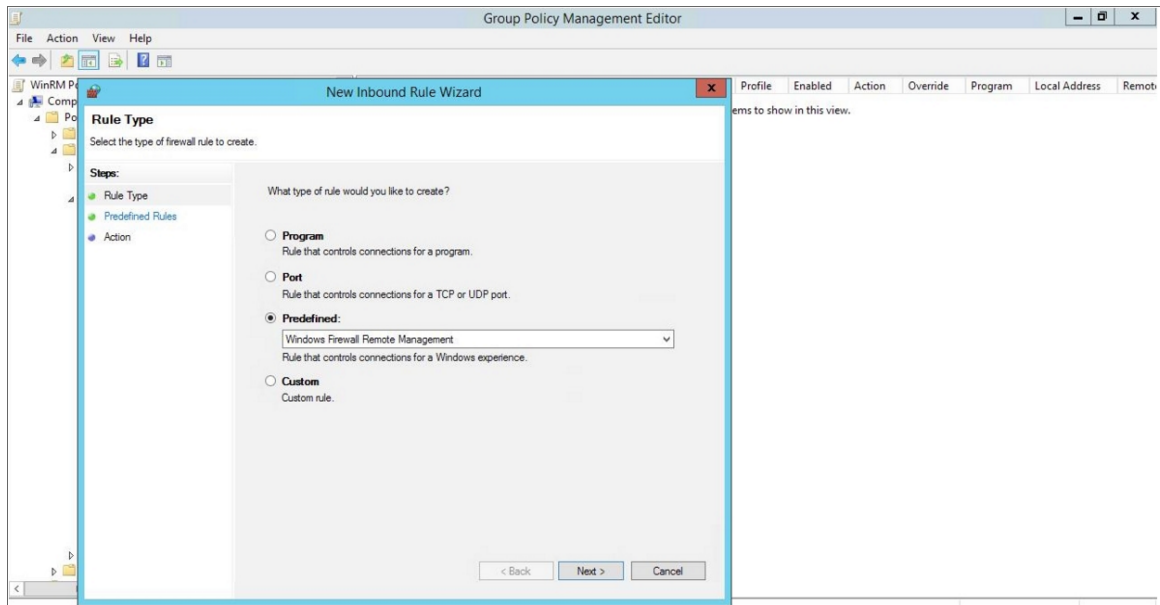
14. Select the *Windows Firewall Remote Management (RPC)* and *Windows Firewall Remote Management (RPC-EPMAP)* check boxes, then click **[Next]**.

15. Select the *Allow the connection* radio button, then click **[Finish]**.
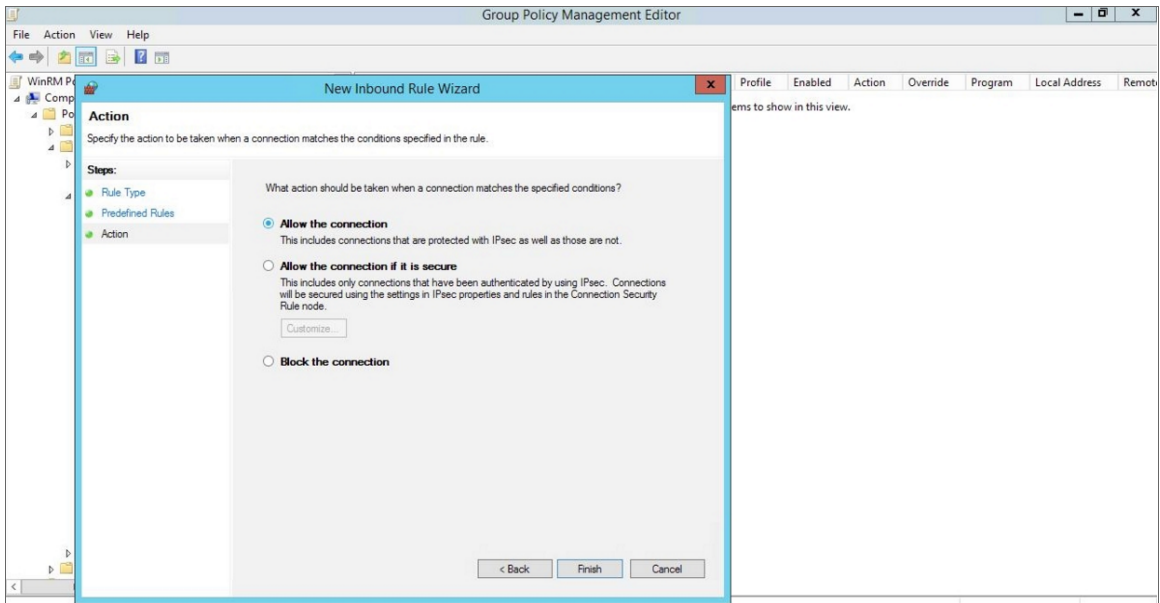
16. In the left panel of the **Group Policy Management Editor** page, navigate to **Computer Configuration > Policies > Windows Settings > Security Settings > Public Key Policies > Automatic Certificate Request Settings**. In the right panel, right-click and select *New > Automatic Certificate Request*.
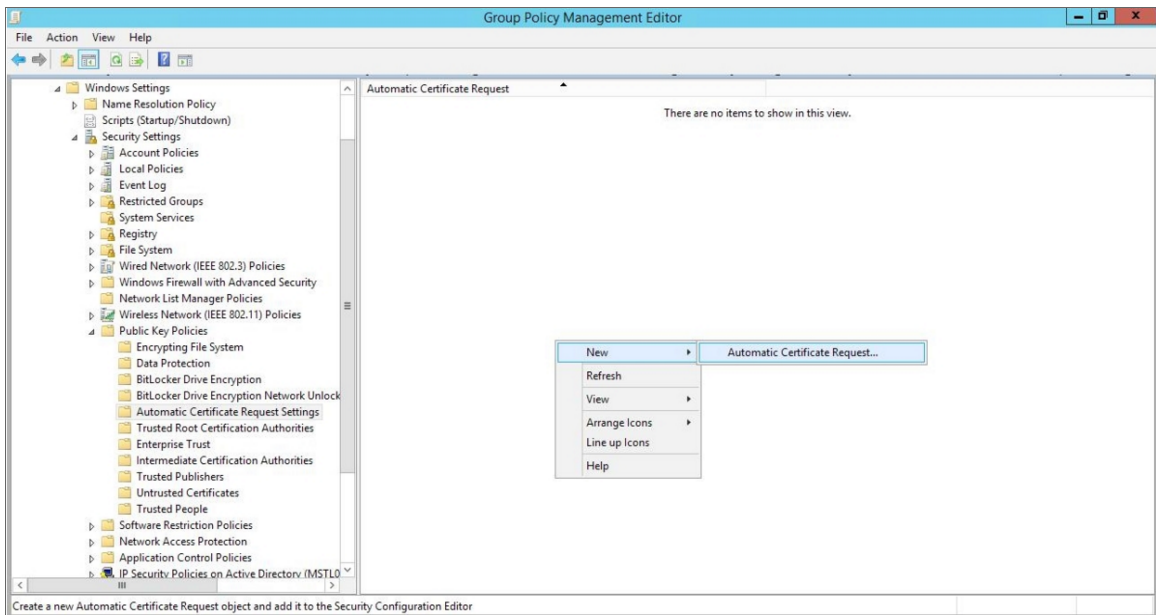
17. The **Automatic Certificate Request Setup Wizard** modal page appears. Click **[Next]**.
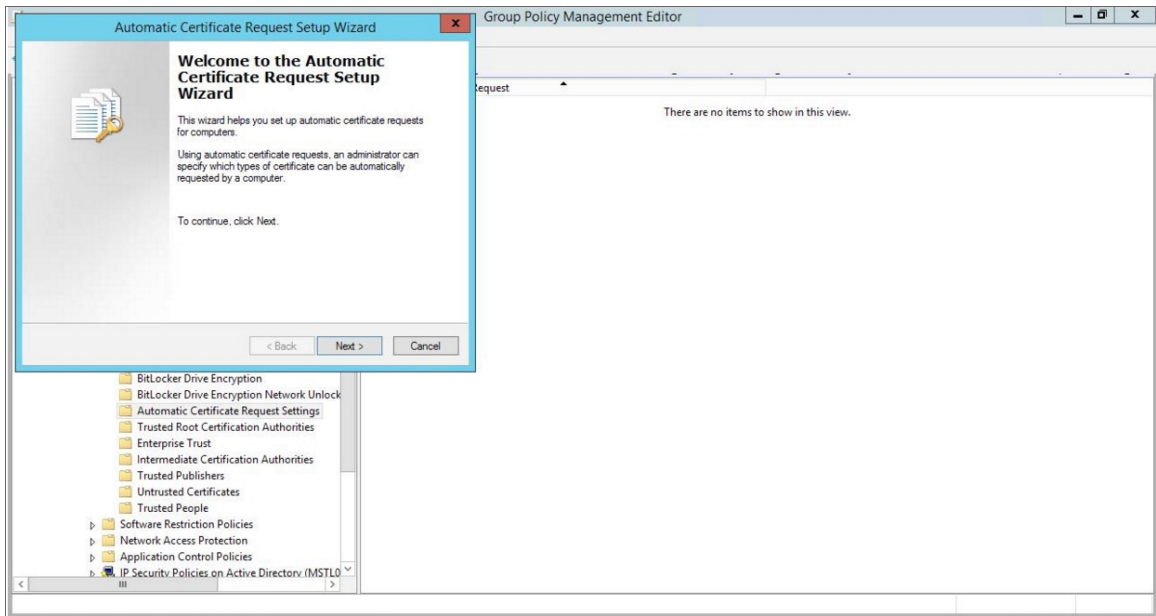
18. Select the *Computer* certificate template. Click **[Next]**, and then click **[Finish]**.
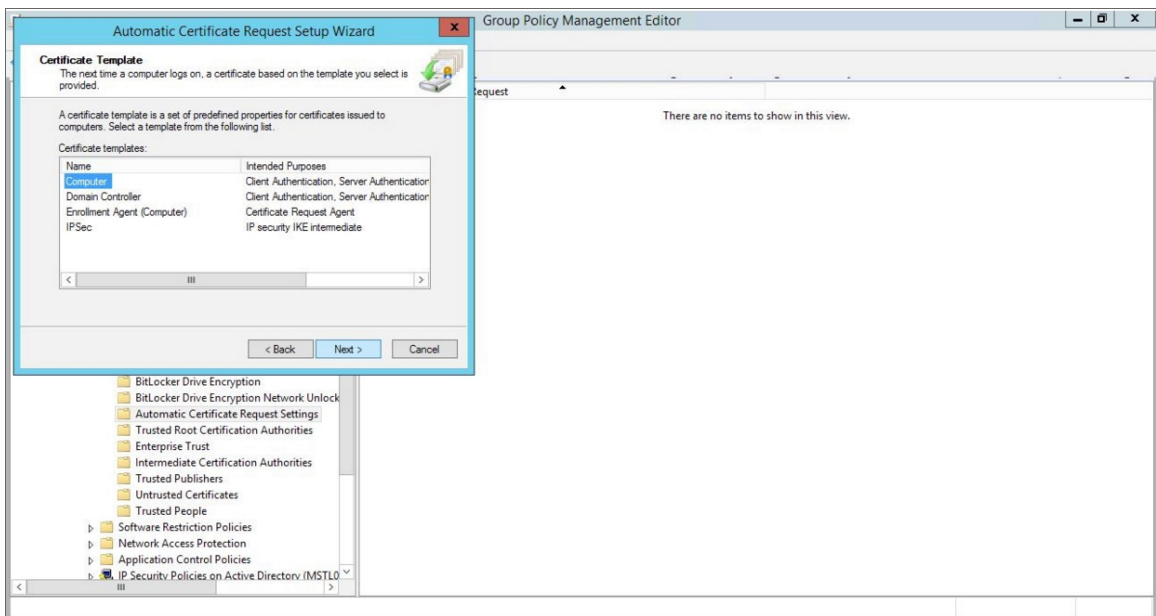
19. In the left panel of the **Group Policy Management Editor** page, navigate to **Computer Configuration > Policies > Windows Settings > Security Settings > Public Key Policies > Trusted Root Certification Authorities**. In the right panel, right-click and select *Import*.
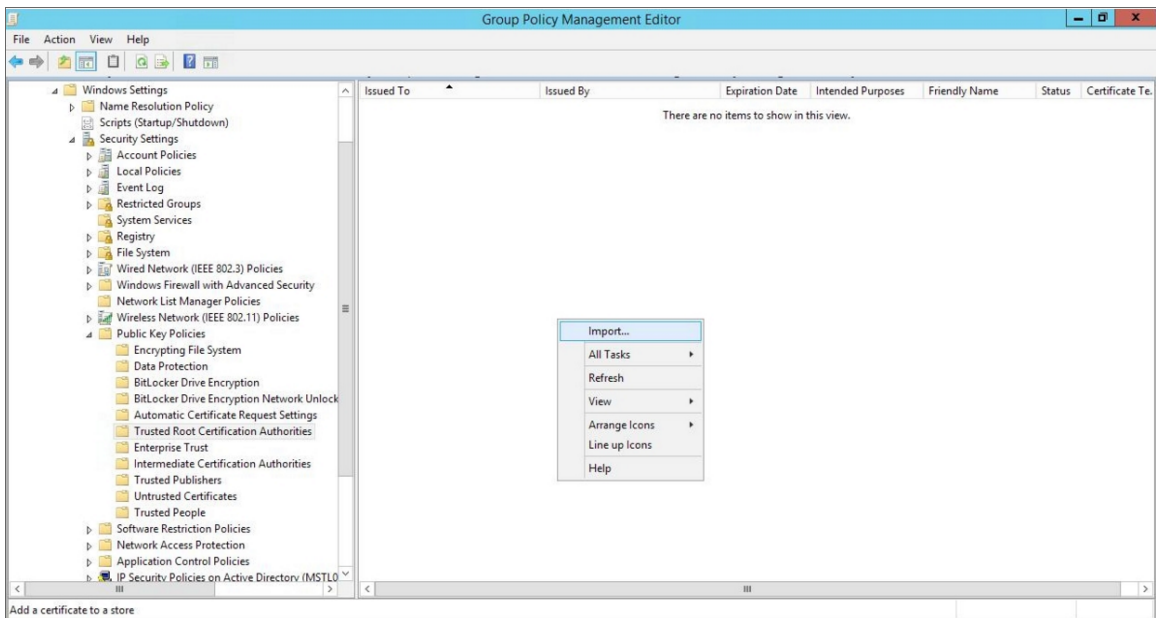
20. The **Certificate Import Wizard** modal page appears. Click **[Next]**.
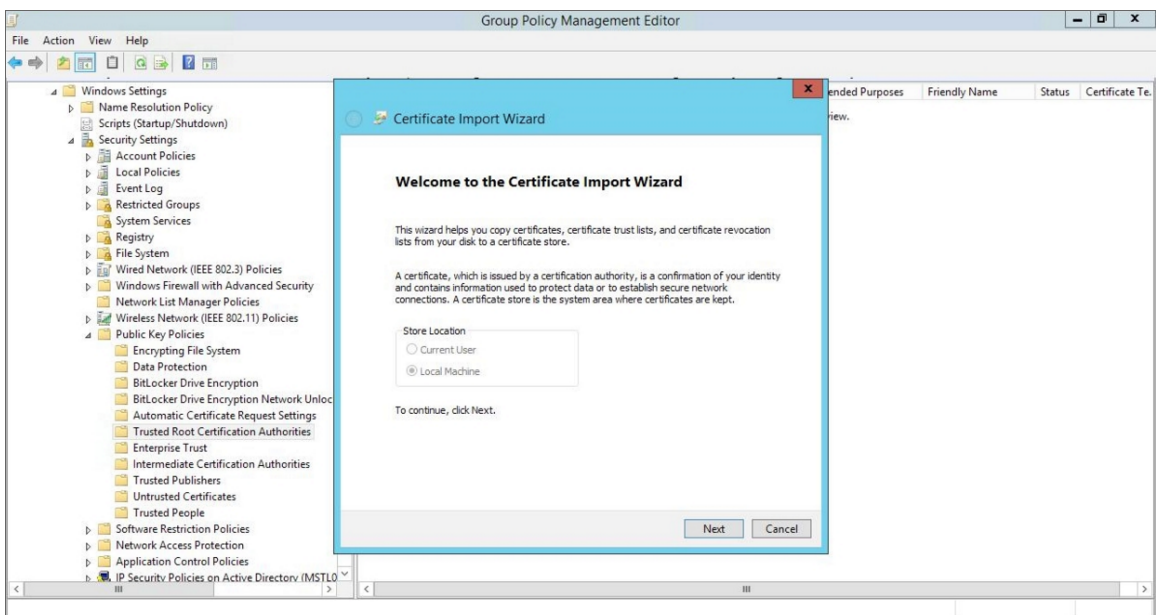
21. Browse to the Certification Authority certificate that you saved to your local directory in step 4, then click [Next].
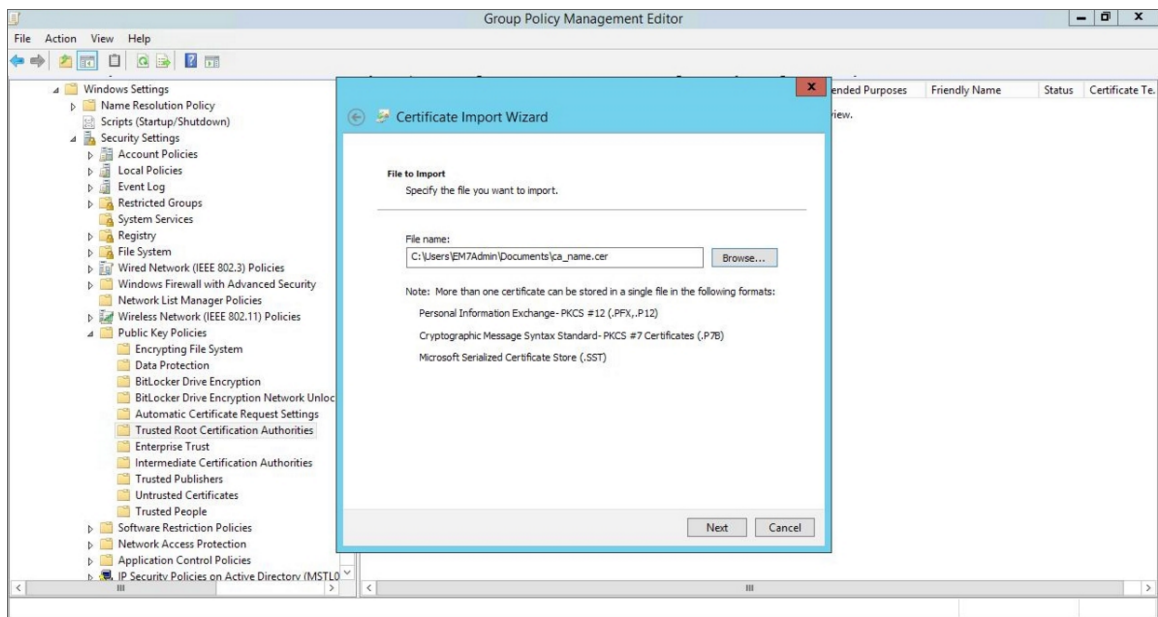
22. Select the **Place all certificates in the following store** radio button, then select the *Trusted Root Certification Authorities* certificate store and click [Next].
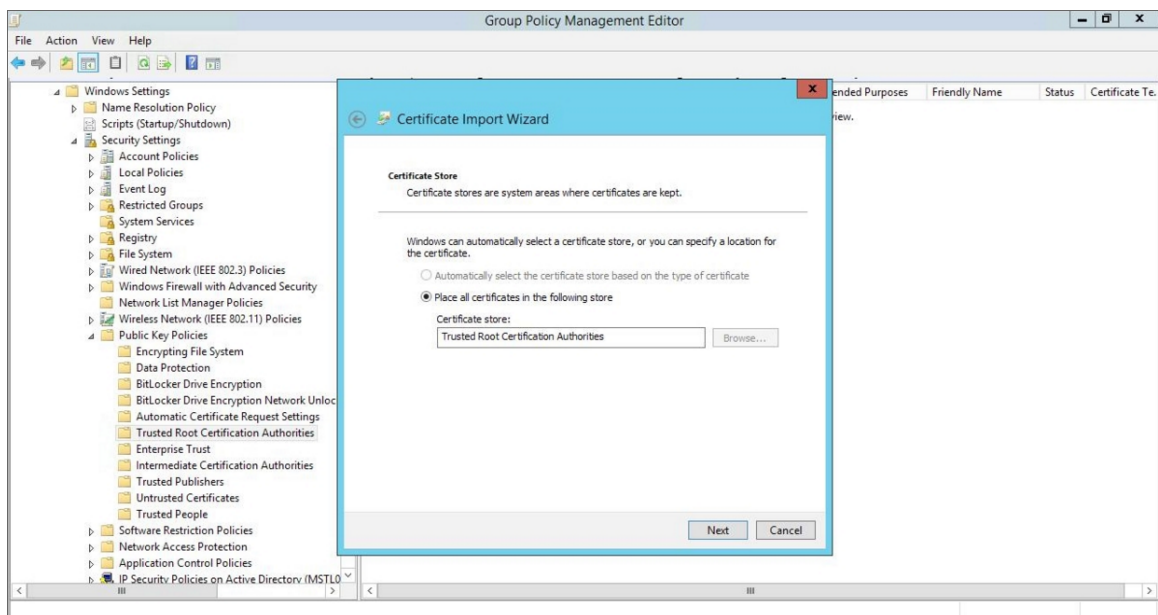
23. Click **[OK]** to confirm that the certificate was successfully imported, and then click **[Finish]**.
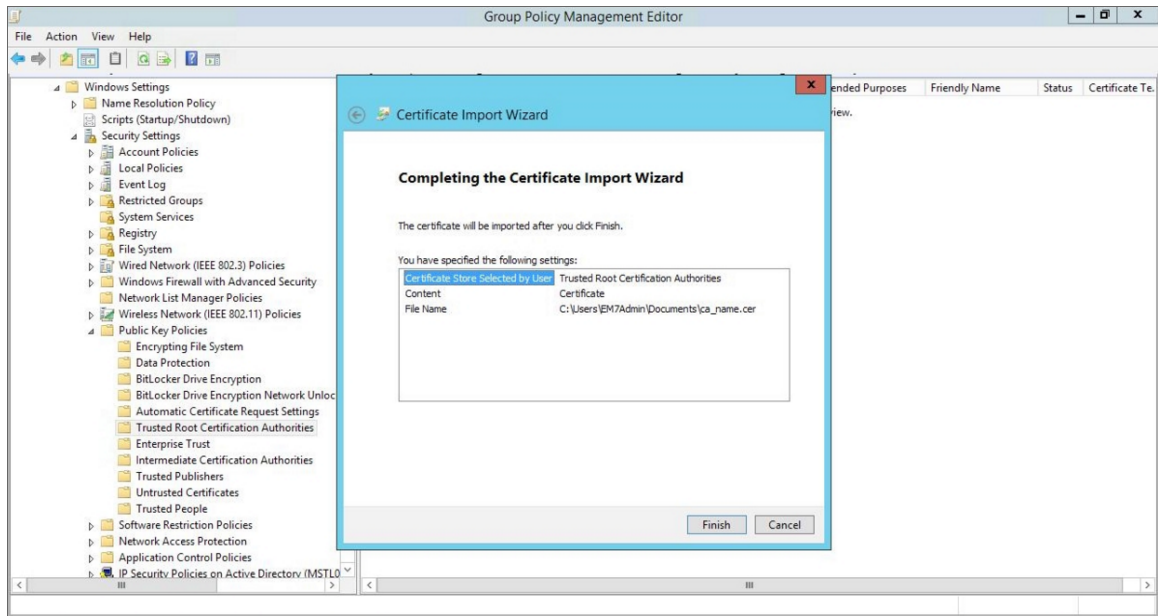
24. In the left panel of the **Group Policy Management Editor** page, navigate to **Computer Configuration > Policies > Administrative Templates > Network > Network Connections > Windows Firewall > Domain Profile**. In the right panel, right-click **Windows Firewall: Define inbound port exceptions** and select *Edit*.
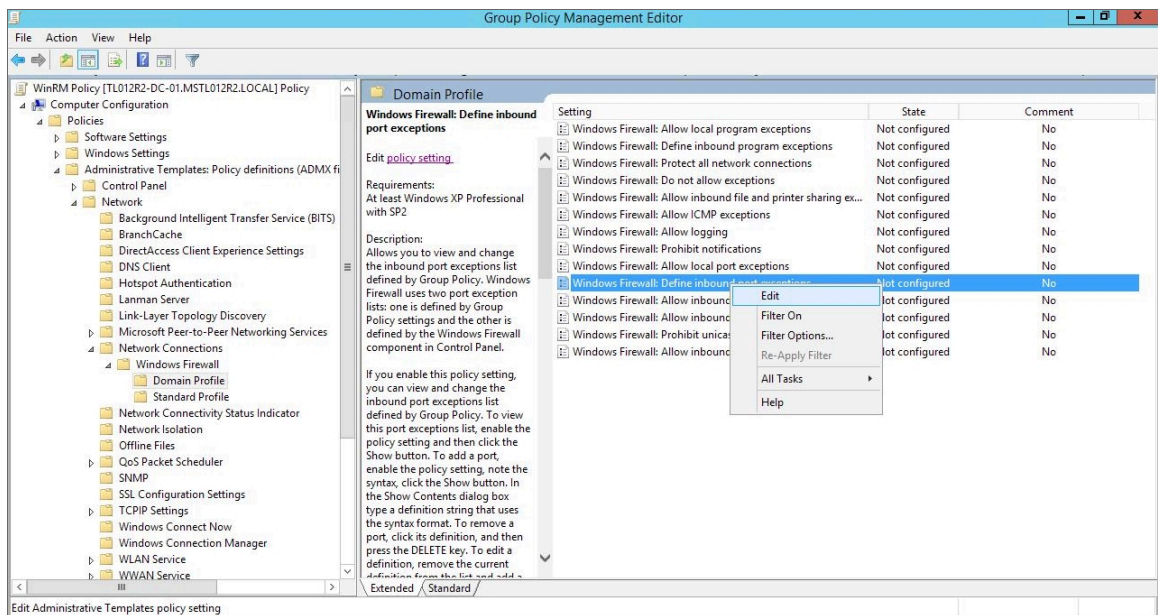
25. The **Windows Firewall: Define inbound port exceptions** modal page appears. Under **Options**, click **[Show]**.

26. The **Show Contents** modal page appears. Enter the following values:
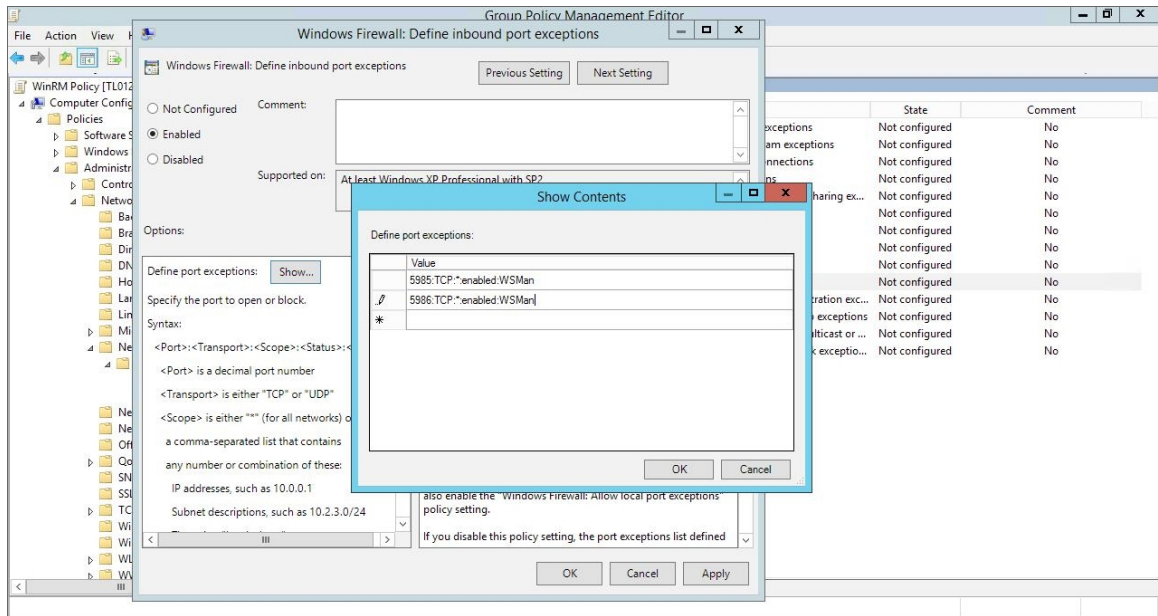
- 5985:TCP:*:enabled:WSMan
- 5986:TCP:*:enabled:WSMan

27. Click **[OK]**, then click **[OK]** again.

28. In the left panel of the **Group Policy Management Editor** page, navigate to **Computer Configuration > Policies > Administrative Templates > Windows Components > Windows Remote Management (WinRM) > WinRM Client**. In the right panel, double-click the **Allow Basic authentication** setting.
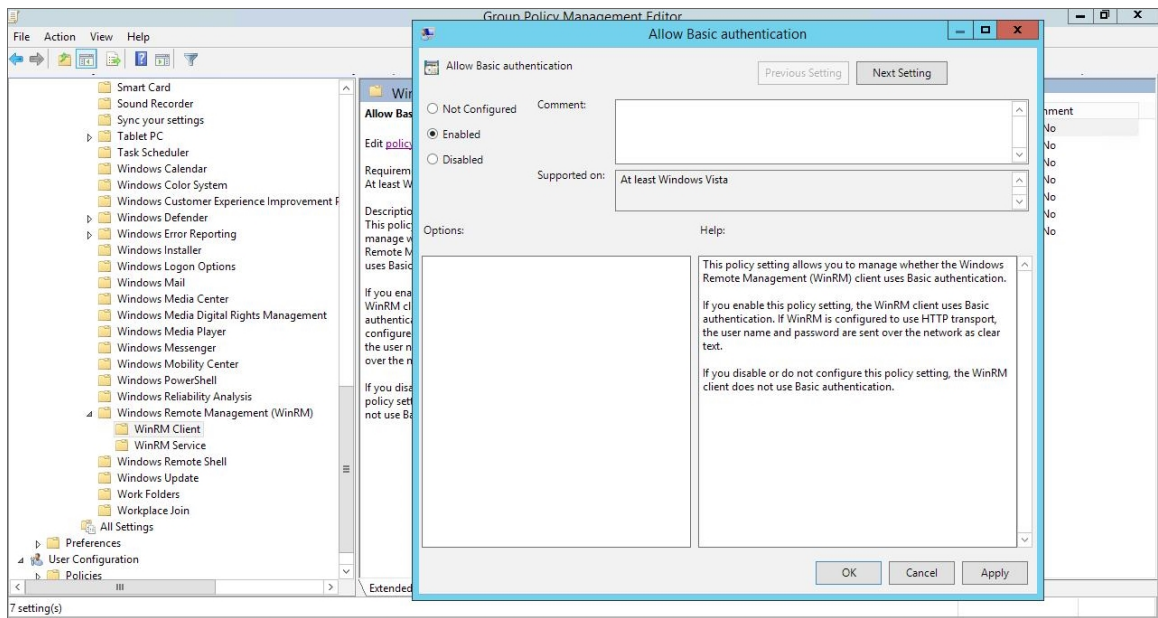
29. Select the **Enabled** radio button, then click **[OK]**.

30. Repeat steps 28 and 29 for the **Allow unencrypted traffic** setting.

31. Double-click the **Trusted Hosts** setting. Select the **Enabled** radio button, enter an asterisk (*) in the **TrustedHostsList** field (under **Options**), and then click **[OK]**.
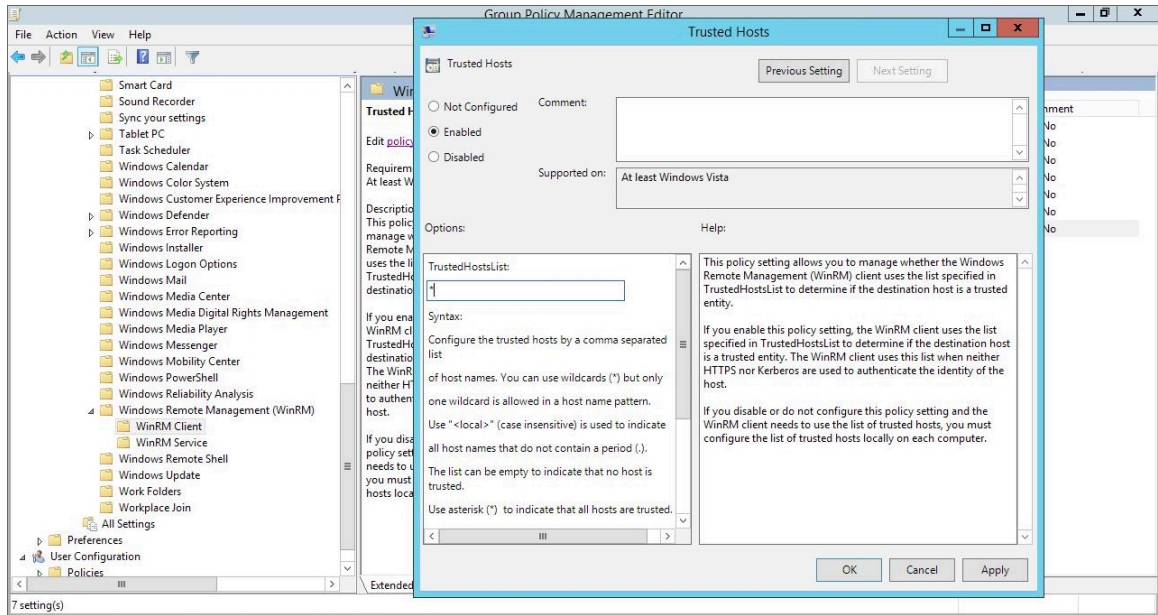
32. In the left panel of the **Group Policy Management Editor** page, navigate to **Computer Configuration > Policies > Administrative Templates > Windows Components > Windows Remote Management (WinRM) > WinRM Service**. In the right panel, double-click the **Allow Basic authentication** setting.
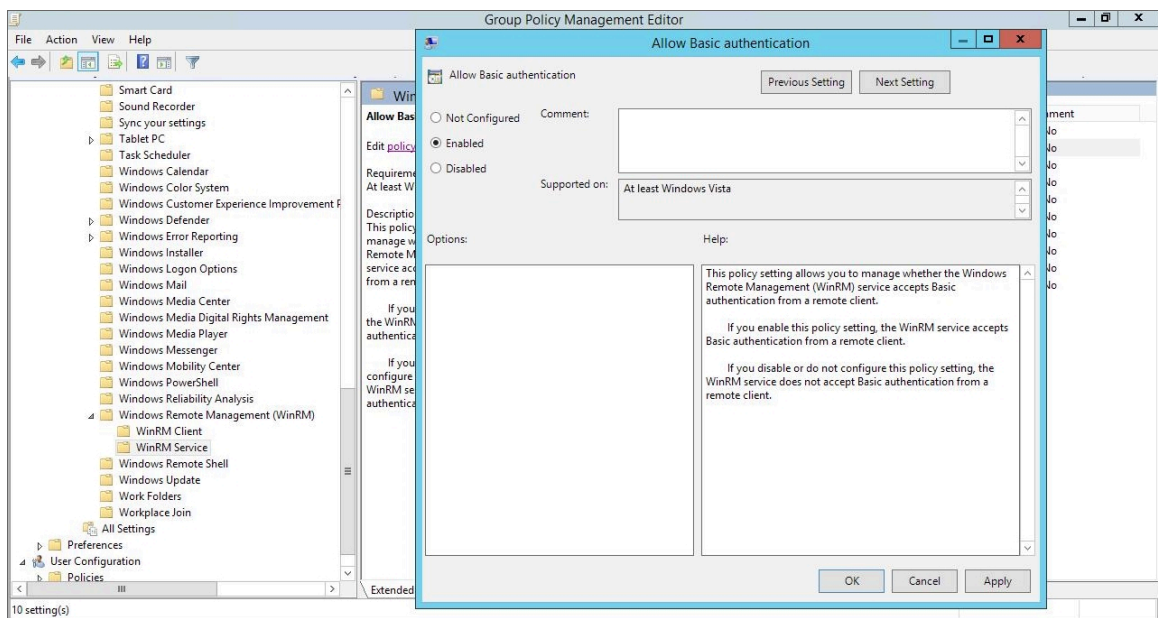
33. Select the **Enabled** radio button, then click **[OK]**.

34. Repeat steps 32 and 33 for the **Allow unencrypted traffic** setting.

35. Double-click the **Allow remote server management through WinRM** setting. Select the **Enabled** radio button, enter an asterisk (*) in the **Pv4 filter** and **Pv6 filter** fields (under **Options**), and then click **[OK]**.
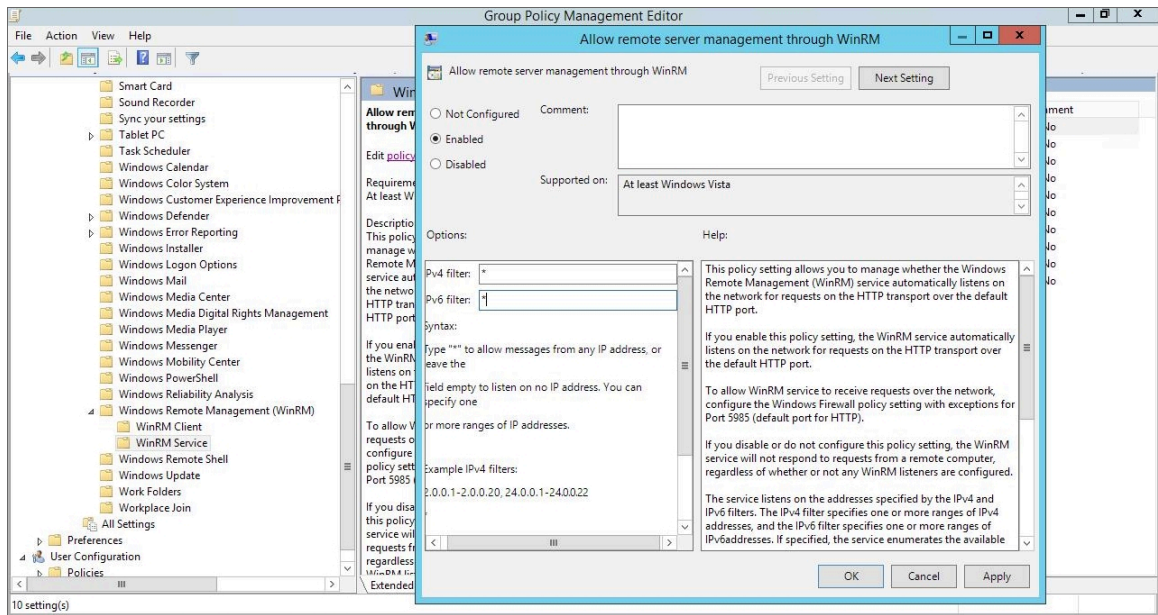
36. In the left panel of the **Group Policy Management Editor** page, navigate to **Computer Configuration > Preferences > Windows Settings > Registry**. In the right panel, right-click and select *New > Registry Item*.
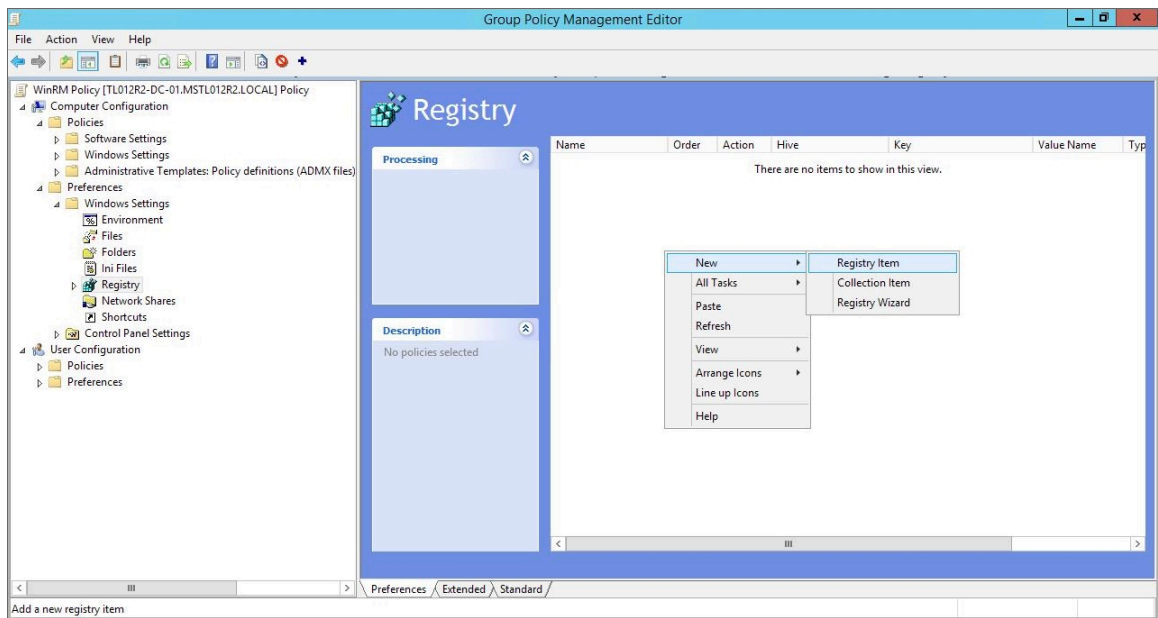
37. In the **New Registry Properties** modal page, edit the values in one or more of the following fields:
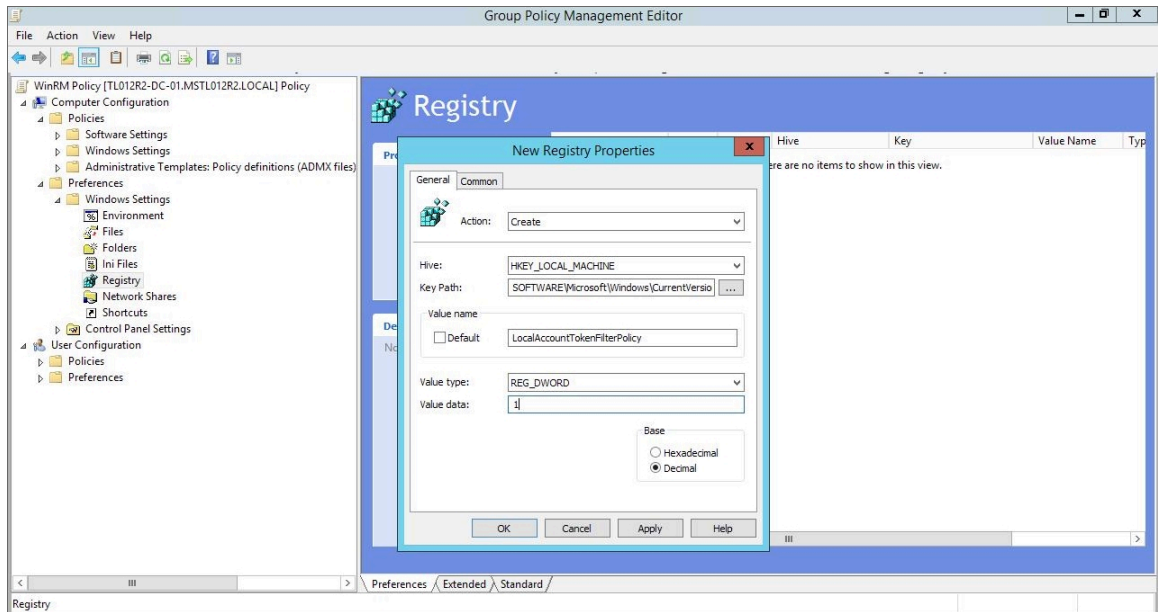
- *Action*. Select *Create*.
- *Hive*. Select *HKEY_LOCAL_MACHINE*.
- *Key Path*. Enter "SOFTWARE\Microsoft\Windows\CurrentVersion\policies\system".
- *Value name*. Enter "WinRS!MaxConcurrentUsers".
- *Value type*. Enter "REG_DWORD".
- *Value data*. Enter "1".
- *Base*. Select *Decimal*.

38. Click the **[OK]** button.

39. Repeat steps 36-38 to make an additional registry change to the local account. In the **New Registry Properties** modal page, edit the following values:

> NOTE: This step is required only if the user account is **not** a domain account and **not** the built-in local administrator account.

- *Action*. Select *Create*.
- *Hive*. Select *HKEY_LOCAL_MACHINE*.
- *Key Path*. Enter "SOFTWARE\Policies\Microsoft\Windows\WinRM\Service\".
- *Value name*. Enter "LocalAccountTokenFilterPolicy".
- *Value type*. Enter "REG_DWORD".

- *Value data*. Enter "1".

- *Base*. Select *Decimal*.

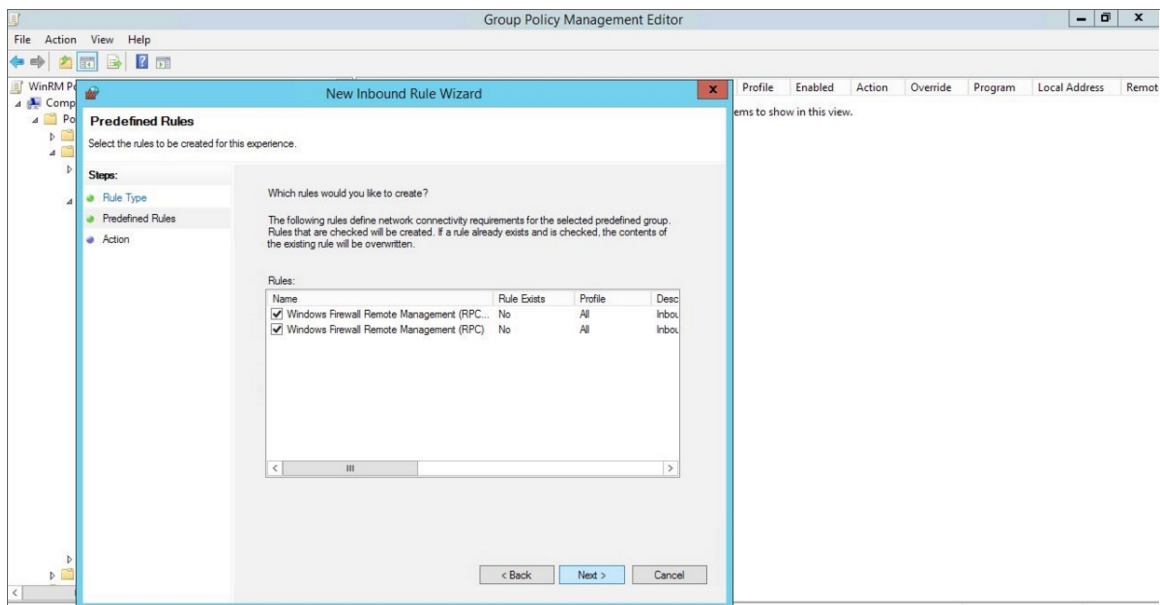40. In the left panel of the **Group Policy Management Editor** page, navigate to **Computer Configuration > Preferences > Control Panel Settings > Services**. In the right panel, right-click and select *New > Service*.
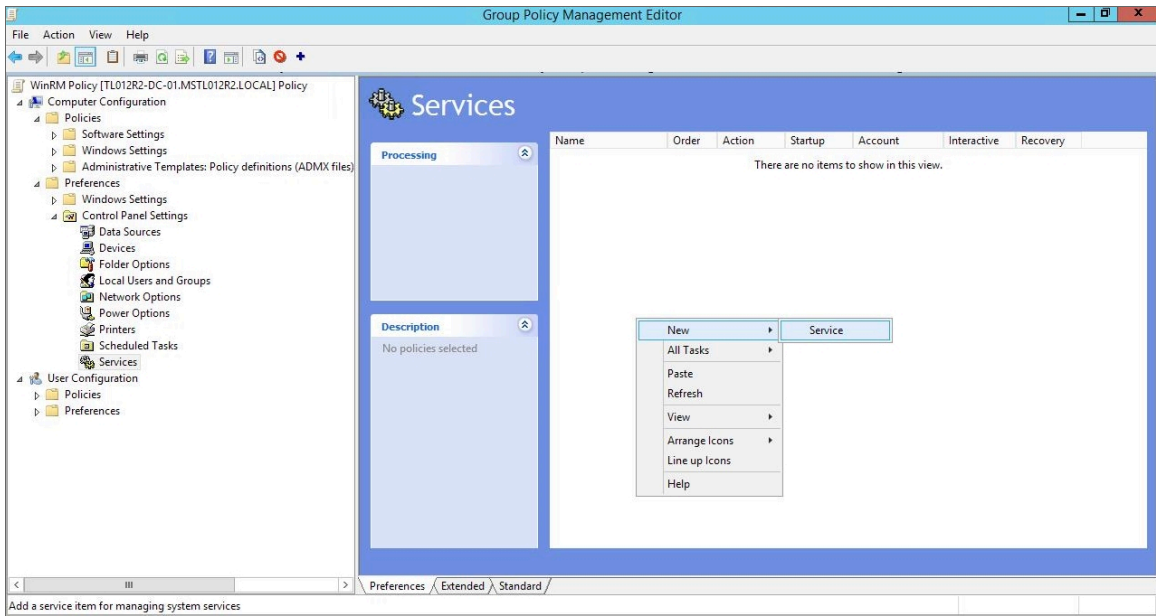
41. In the **New Service Properties** modal page, edit the values in one or more of the following fields:
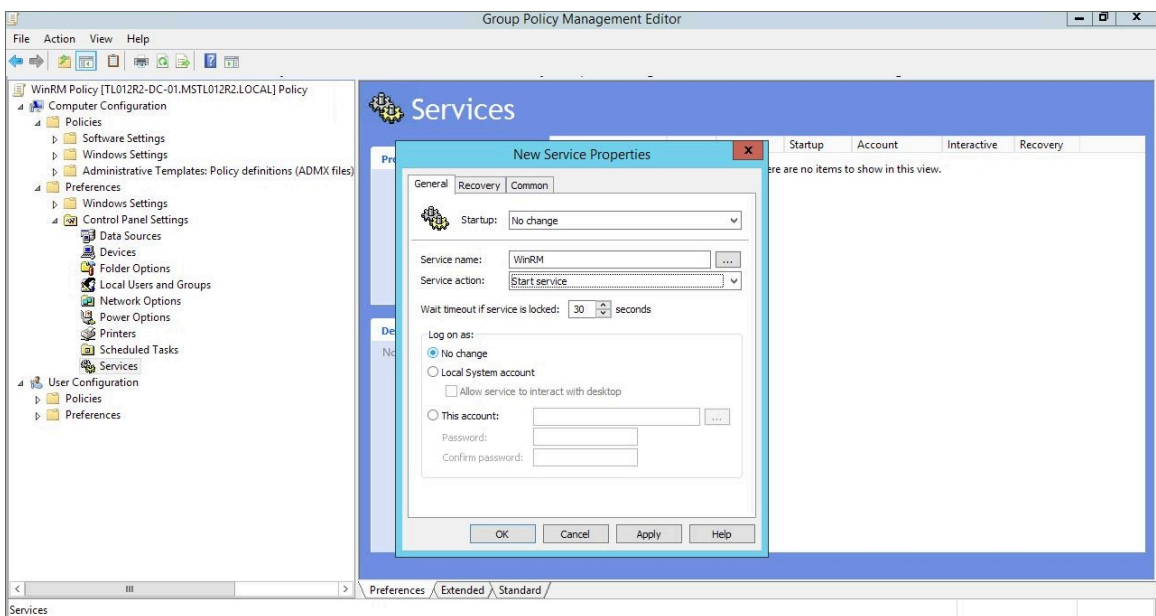
- *Startup*. Select *No change*.

- ***Service name***. Enter "WinRM".
- ***Service action***. Select *Start service*.
- ***Wait timeout if service is locked***. Select *30* seconds.
- ***Log on as***. Select *No change*.

42. Click the **[Recovery]** tab, then edit the values in one or more of the following fields:
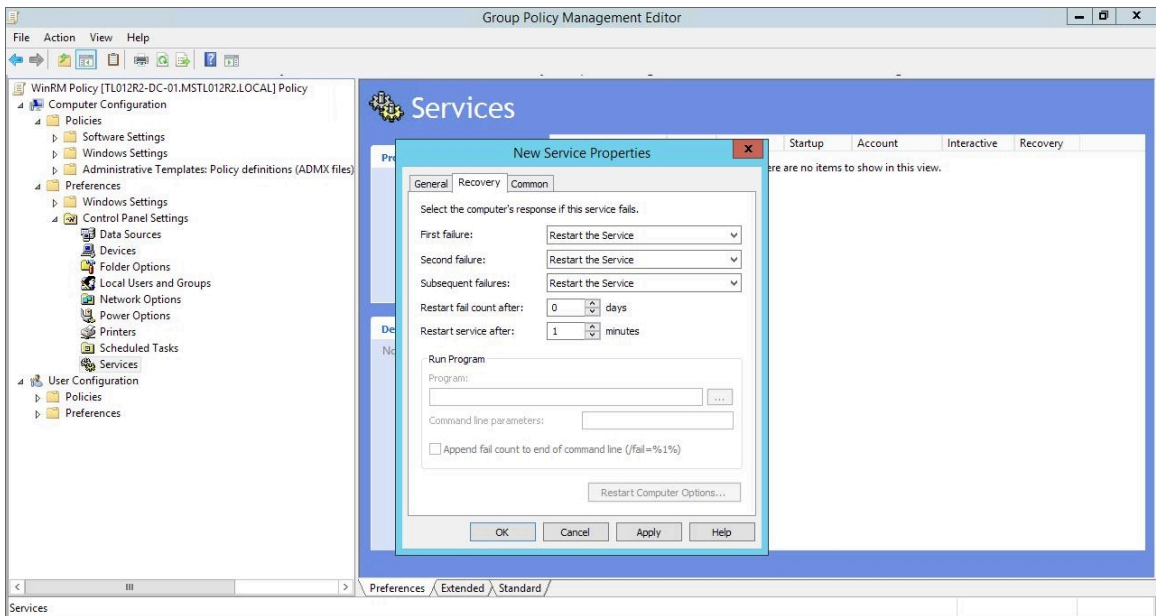
- ***First failure***. Select *Restart the Service*.
- ***Second failure***. Select *Restart the Service*.
- ***Subsequent failures***. Select *Restart the Service*.
- ***Restart fail count after***. Select *0* days.
- ***Restart service after***. Select *1* minute.

43. Click the **[OK]** button.

44. To enforce your group policy, in the left panel of the **Group Policy Management Editor** page, navigate to **Forest > Domains > [your local domain] > PowerShell Remote Management Policy**. In the **PowerShell Remote Management Policy** panel on the right, right-click the local domain name under *The following sites, domains, and OUs are linked to this GPO* and select *Enforced*.
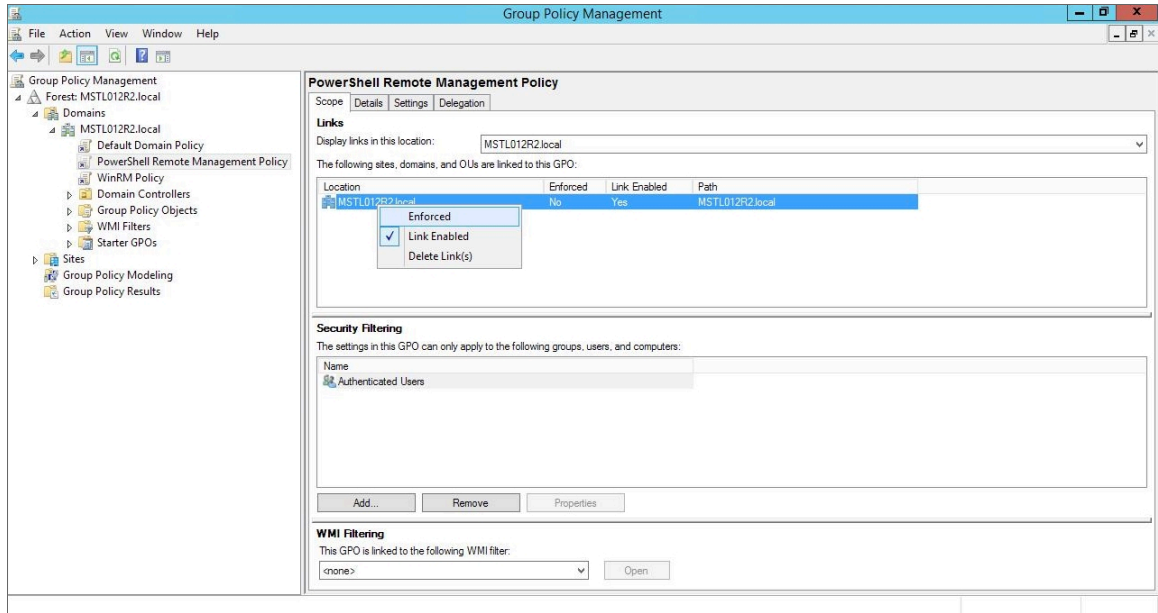
45. To enable your group policy, in the left panel of the **Group Policy Management Editor** page, navigate to **Forest > Domains > [your local domain] > Group Policy Objects > WinRM Policy**. Right-click **WinRM Policy**, then select *GPO Status > Enabled*.
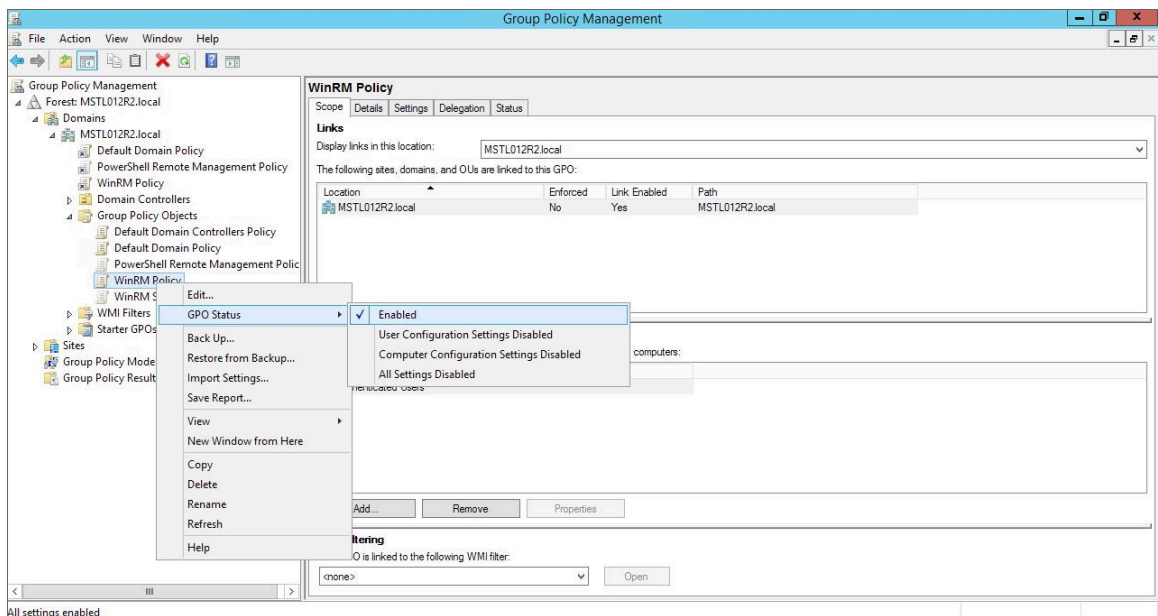
Step 4: Configuring a Windows Management Proxy

If SL1 cannot execute PowerShell requests directly on a Windows server, you can optionally configure an additional Windows server to act as a proxy for those PowerShell requests. To use a proxy, you must configure at least two Windows servers:

- A target server that SL1 cannot communicate with directly.
- A proxy server that SL1 will communicate with to execute PowerShell requests on the target server.

To configure the target and proxy servers, perform the following steps:

1. Configure a user account that SL1 will use to connect to the proxy server and the proxy server will use to connect to the target server. The user account can either be a local account or an Active Directory account; however, the user account must have the same credentials on the target and proxy servers and be in the Local Administrator's group on both servers.

2. If you have created a local user account on the Windows Server instead of an Active Directory account, you must configure encrypted communication between SL1 and the Windows server. To do this, you must *configure a Server Authentication certificate*.

3. *Configure Windows Remote Management* on the target server and the proxy server.

4. Log in to the proxy server as an administrator.

5. Open the PowerShell command window.

6. Right-click on the PowerShell icon in the taskbar and select *Run as Administrator*.

7. Execute one of the following commands on the proxy server to allow the proxy server to trust one or more target servers:

    - To allow the proxy server to trust all servers (not recommended), execute the following command:

      ```
      Set-Item WSMan:\Localhost\Client\TrustedHosts -value *
      ```

    - To allow the proxy server to trust only specific target servers, execute the following command, inserting a list that includes the IP address for each target server. Separate the list of IP addresses with commas.

      ```
      Set-Item WSMan:\Localhost\Client\TrustedHosts -value <comma-delimited-list-
      of-target-server-IPs>
      ```

> **NOTE:** The following step is required only if the user account is **not** a domain account and **not** the built-in local administrator account.

8. Execute the following command on the proxy server to configure the LocalAccountTokenFilterPolicy:

    ```
    New-ItemProperty
    "HKLM:\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System" -Name
    "LocalAccountTokenFilterPolicy" -Value 1 -PropertyType "DWORD"
    ```

> **NOTE**: If the proxy server is in a different Windows domain (domain A) than the target servers (domain B), and the proxy server uses a user account from Active Directory, and Active Directory is in the same Windows domain as the target servers (domain B), you must perform the following to allow the proxy server to send PowerShell commands to the target servers:
>
> - On the domain controller for each domain (domain A and domain B), create new forward-lookup zones and reverse-lookup zones that allow name resolution to work between the two domains.
> - On the domain controller for each domain (domain A and domain B), create a non-transitive realm trust between the two domains.
> - Login to the proxy server and add the Active Directory account (from domain A) to the Local Administrator's group for the proxy server. You should be able to select the account on the proxy server after you create the non-transitive realm trust between the two domains.

Step 5: Increasing the Number of PowerShell Dynamic Applications That Can Run Simultaneously

You can optionally execute a series of commands that will allow SL1 to increase the default maximum number of PowerShell Dynamic Applications that can run simultaneously.

To do so:

1. Determine the number of Dynamic Applications that will be used to monitor the Windows server. Multiply this number by three.
2. Open a PowerShell command prompt. Log in as an Administrator.
3. At the prompt, execute the following commands:

    ```
    Set-Item WSMan:\Localhost\Shell\MaxShellsPerUser -value <number you
    calculated in step 1>

    Set-Item WSMan:\Localhost\Service\MaxConcurrentOperationsPerUser -value
    <number you calculated in step 1>

    Restart-Service WinRM
    ```

4. Repeat these steps on each Windows server that will be monitored by SL1.

Chapter

4

SNMP and PowerShell Dynamic Applications for Windows Devices

Overview

Use the following menu options to navigate the SL1 user interface:

- To view a pop-out list of menu options, click the menu icon (▤).

- To view a page containing all the menu options, click the Advanced menu icon (⋯).

The following sections describe the SNMP and PowerShell Dynamic Applications that SL1 uses to monitor Windows devices:

SNMP Dynamic Applications for Windows Devices

If you configure your Windows system to respond to SNMP requests from SL1, you can discover your Windows system as an SNMP device. When SL1 discovers a Windows system as an SNMP device, the platform will automatically collect the same data from the Windows system that the platform collects from most network devices. This data includes interface usage, file system usage, CPU usage, memory usage, and hardware configuration information.

In addition to the common SNMP data collection, you can install an optional agent that reports WMI information through SNMP. The following SNMP Dynamic Applications can be used to collect the information reported by the optional agent:

- MSSQL: General
- MSSQL: Memory
- MSSQL: SQL Stats

PowerShell Dynamic Applications

If you configure your Windows system to respond to PowerShell requests from SL1, you can use PowerShell Dynamic Applications to collect information from your Windows system.

All of the PowerShell Dynamic Applications include a discovery object. If you include a credential for PowerShell Dynamic Applications in the discovery session that includes your Windows system, SL1 will automatically align the appropriate PowerShell Dynamic Applications to the Windows system. For more information about creating a discovery session, see the **Discovery & Credentials** manual.

The following PowerPacks include PowerShell Dynamic Applications for Microsoft Servers.

Microsoft: Active Directory Server

NOTE: The Dynamic Applications in this PowerPack support Windows Server 2012 R2.

The following PowerShell Dynamic Applications can be used to collect performance data from Active Directory servers:

- Microsoft: Active Directory Address Book Performance

- Microsoft: Active Directory Async Thread Queue Performance
- Microsoft: Active Directory Database Performance
- Microsoft: Active Directory Directory Services Reads Performance
- Microsoft: Active Directory Directory Services Searches Performance
- Microsoft: Active Directory Directory Services Writes Performance
- Microsoft: Active Directory DRA Performance
- Microsoft: Active Directory LDAP Performance
- Microsoft: Active Directory Security Account Management Performance
- Microsoft: Active Directory Services General Performance
- Microsoft: Active Directory Web Service Performance

Microsoft: DHCP Server

> **NOTE:** The Dynamic Applications in this PowerPack support Windows Server 2012.

The following PowerShell Dynamic Applications can be used to collect performance data from DHCP servers:

- Microsoft: DHCP IPv4 Performance
- Microsoft: DHCP IPv4 Scope Performance
- Microsoft: DHCP Service Performance

The following PowerShell Dynamic Applications can be used to collect configuration data from DHCP servers:

- Microsoft: DHCP IPv4 Lease Configuration
- Microsoft: DHCP IPv6 Lease Configuration
- Microsoft: DHCP Server Performance

Microsoft: DNS Server

> **NOTE:** The Dynamic Applications in this PowerPack support Windows Server 2008 R2, 2012, and 2012 R2.

The following PowerShell Dynamic Applications can be used to collect performance data from DNS servers:

- Microsoft: DNS AXFR Performance
- Microsoft: DNS Dynamic Update Performance
- Microsoft: DNS IXFR Performance
- Microsoft: DNS Memory Performance
- Microsoft: DNS Notification Performance

- Microsoft: DNS Recursion Performance
- Microsoft: DNS Secure Dynamic Update Performance
- Microsoft: DNS TCP Performance
- Microsoft: DNS Total Overall Performance
- Microsoft: DNS UDP Performance
- Microsoft: DNS WINS Performance
- Microsoft: DNS Zone Transfer Performance

Microsoft: Exchange Server

The following PowerShell Dynamic Applications can be used to collect performance data from Exchange 2013 and Exchange 2016 servers:

- Microsoft: Exchange CAS ActiveSync Performance
- Microsoft: Exchange CAS Address Book Load Performance
- Microsoft: Exchange CAS Address Book Service Performance
- Microsoft: Exchange CAS Availability Service Performance
- Microsoft: Exchange CAS OWA Performance
- Microsoft: Exchange CAS Performance
- Microsoft: Exchange CAS RPC Client Access Load Performance
- Microsoft: Exchange CAS RPC Client Access Performance
- Microsoft: Exchange MBS Database Performance
- Microsoft: Exchange MBS Info Store RPC Processing Stats
- Microsoft: Exchange MBS Information Store Performance
- Microsoft: Exchange MBS Replay Log I/O Latency Requirements
- Microsoft: Exchange TPS Disk Performance
- Microsoft: Exchange TPS Transport Database Performance
- Microsoft: Exchange TPS Transport Load Assessment Stats
- Microsoft: Exchange UMS General Performance

Microsoft: Exchange Server 2010

The following PowerShell Dynamic Applications can be used to collect performance data from Exchange 2010 servers:

- Microsoft: Exchange 2010 CAS Address Book Load Performance
- Microsoft: Exchange 2010 CAS Address Book Service Performance
- Microsoft: Exchange 2010 CAS Availability Service Performance
- Microsoft: Exchange 2010 CAS OWA Performance

- Microsoft: Exchange 2010 CAS Performance
- Microsoft: Exchange 2010 CAS RPC Client Access Load Performance
- Microsoft: Exchange 2010 CAS RPC Client Access Performance
- Microsoft: Exchange 2010 MBS Client-Related Search Performance
- Microsoft: Exchange 2010 MBS Database Performance
- Microsoft: Exchange 2010 MBS Info Store RPC Processing Stats
- Microsoft: Exchange 2010 MBS Information Store Performance
- Microsoft: Exchange 2010 MBS Message Queuing Performance
- Microsoft: Exchange 2010 MBS Replay Log I/O Latency Requirements
- Microsoft: Exchange 2010 MBS RPC Client Throttling Performance
- Microsoft: Exchange 2010 MBS Store Client Request Performance
- Microsoft: Exchange 2010 TPS Disk Performance
- Microsoft: Exchange 2010 TPS Transport Database Performance
- Microsoft: Exchange 2010 TPS Transport Load Assessment Stats
- Microsoft: Exchange 2010 TPS Transport Queue Length Performance
- Microsoft: Exchange 2010 UMS General Performance

Microsoft: Hyper-V Server

NOTE: The Dynamic Applications in this PowerPack support Hyper-V Server 2008 R2, 2012, and 2012 R2.

The following PowerShell Dynamic Applications can be used to collect performance data from Hyper-V servers:

- Microsoft: Hyper-V Component Count
- Microsoft: Hyper-V Logical Processor Performance
- Microsoft: Hyper-V Overall Guest CPU Performance
- Microsoft: Hyper-V Process Performance
- Microsoft: Hyper-V Root Virtual Processor Performance
- Microsoft: Hyper-V Virtual Processor Performance
- Microsoft: Hyper-V Virtual Storage Device Performance
- Microsoft: Hyper-V Virtual Switch Performance

The following PowerShell Dynamic Applications can be used to collect configuration data from Hyper-V servers:

- Microsoft: Hyper-V Component Count Configuration
- Microsoft: Hyper-V Host Configuration

This PowerPack also includes Snippet Dynamic Applications that discover virtual machines managed by the Hyper-V host. Although the Dynamic Applications are of type "Snippet", the snippets themselves perform PowerShell requests to collect data and use PowerShell credentials. See the *Discovering Component Devices on Hyper-V Systems* section for more information.

- Microsoft: Hyper-V Guest Configuration
- Microsoft: Hyper-V Guest Configuration Cache
- Microsoft: Hyper-V Guest Discovery

This PowerPack also includes Snippet Dynamic Applications that retrieve performance data from virtual machines managed by the Hyper-V host. Although the Dynamic Applications are of type "Snippet", the snippets themselves perform PowerShell requests to collect data and use PowerShell credentials:

- Microsoft: Hyper-V Connected Clients
- Microsoft: Hyper-V Guest CPU Performance
- Microsoft: Hyper-V Guest IDE Controller Performance
- Microsoft: Hyper-V Guest Interface Performance
- Microsoft: Hyper-V Guest Memory Performance

Microsoft: IIS Server

NOTE: The Dynamic Applications in this PowerPack support Internet Information Services (ISS) versions 7.5, 8.0, 8.5, and 10.0.

The following PowerShell Dynamic Applications can be used to collect performance data from IIS servers:

- Microsoft: IIS Active Server Pages Performance
- Microsoft: IIS Core Performance
- Microsoft: IIS Web Service Performance

The following PowerShell Dynamic Applications can be used to collect configuration data from IIS servers:

- Microsoft: IIS Server Configuration

Microsoft: Lync Server 2010

The following PowerShell Dynamic Applications can be used to collect performance data from Lync 2010 servers:

- Microsoft: Lync 2010 Announcement Service Performance
- Microsoft: Lync 2010 AS MCU Performance
- Microsoft: Lync 2010 Auto Attendant Performance
- Microsoft: Lync 2010 AV MCU Performance
- Microsoft: Lync 2010 AV SIP/MRAS/QOE Performance

- Microsoft: Lync 2010 Call Park Service Performance
- Microsoft: Lync 2010 Conferencing Compatibility Performance
- Microsoft: Lync 2010 Data Conferencing Performance
- Microsoft: Lync 2010 IM MCU Performance
- Microsoft: Lync 2010 Response Group Performance
- Microsoft: Lync 2010 SIP Load Management Performance
- Microsoft: Lync 2010 SIP Networking Performance
- Microsoft: Lync 2010 SIP Peers Performance
- Microsoft: Lync 2010 SIP Protocol Performance
- Microsoft: Lync 2010 SIP Response Performance
- Microsoft: Lync 2010 SipEps Incoming Message Performance
- Microsoft: Lync 2010 User Services Performance
- Microsoft: Lync 2010 Web Services Performance

The following PowerShell Dynamic Applications can be used to collect configuration data from Lync 2010 servers:

- Microsoft: Lync 2010 AS MCU Configuration
- Microsoft: Lync 2010 AV MCU Configuration
- Microsoft: Lync 2010 Conferencing Compatibility Configuration
- Microsoft: Lync 2010 Data Conferencing Configuration
- Microsoft: Lync 2010 Service Health Configuration
- Microsoft: Lync 2010 User Services Configuration

Microsoft: SharePoint Server

NOTE: The Dynamic Applications in this PowerPack support SharePoint Server 2010 SE.

The following PowerShell Dynamic Applications can be used to collect performance data from SharePoint servers:

- Microsoft: SharePoint Core Performance
- Microsoft: SharePoint Indexer Performance
- Microsoft: SharePoint Query Performance

Microsoft: Skype for Business

NOTE: This PowerPack was previously named *Microsoft: Lync Server 2013*.

The following PowerShell Dynamic Applications can be used to collect performance data from Lync 2013 servers:

- Microsoft: Lync 2013 AS MCU Performance
- Microsoft: Lync 2013 AV MCU Performance
- Microsoft: Lync 2013 AV SIP/MRAS/QOE Performance
- Microsoft: Lync 2013 Bandwidth Services Performance
- Microsoft: Lync 2013 Call Park Service Performance
- Microsoft: Lync 2013 Data Conferencing Performance
- Microsoft: Lync 2013 IM MCU Performance
- Microsoft: Lync 2013 Mediation Server Performance
- Microsoft: Lync 2013 Response Group Performance
- Microsoft: Lync 2013 SIP Load Management Performance
- Microsoft: Lync 2013 SIP Networking Performance
- Microsoft: Lync 2013 SIP Peers Performance
- Microsoft: Lync 2013 SIP Protocol Performance
- Microsoft: Lync 2013 SIP Response Performance
- Microsoft: Lync 2013 SipEps Incoming Message Performance
- Microsoft: Lync 2013 User Services Performance
- Microsoft: Lync 2013 Web Services Performance

The following PowerShell Dynamic Applications can be used to collect configuration data from Lync 2013 servers:

- x Microsoft: Lync 2013 AS MCU Configuration
- x Microsoft: Lync 2013 AV MCU Configuration
- x Microsoft: Lync 2013 Data Conferencing Configuration
- x Microsoft: Lync 2013 Service Health Configuration
- x Microsoft: Lync 2013 User Services Configuration

Microsoft: SQL Server

> **NOTE:** The Dynamic Applications in this PowerPack support SQL Server 2008, 2012, 2014, and 2016.

The following PowerShell Dynamic Applications can be used to collect performance data from SQL servers:

- Microsoft: SQL 2008 Buffer Pages Performance
- Microsoft: SQL Buffer Performance
- Microsoft: SQL Database Performance
- Microsoft: SQL Memory Performance

- Microsoft: SQL Plan Cache Performance
- Microsoft: SQL Query Performance
- Microsoft: SQL Session Performance
- Microsoft: SQL Table Lock/Latch Performance

Microsoft: Windows Server

> **NOTE:** The Dynamic Applications in this PowerPack support Windows Server 2008 R2, 2012, 2012 R2, and 2016, as well as Windows 10.

The following PowerShell Dynamic Applications can be used to collect configuration data from Windows servers:

- Microsoft: Print Server
- Microsoft: Windows Server BIOS Configuration
- Microsoft: Windows Server Configuration Cache
- Microsoft: Windows Server CPU Configuration
- Microsoft: Windows Server Device Discovery
- Microsoft: Windows Server Disk Configuration
- Microsoft: Windows Server Interface Configuration
- Microsoft: Windows Server Memory Configuration
- Microsoft: Windows Server OS Configuration
- Microsoft: Windows Server Software Configuration

> **NOTE:** The "Microsoft: Windows Server Configuration Cache" Dynamic Application caches data that is consumed by all of the other configuration Dynamic Applications in the list.

> **NOTE:** When the "Microsoft: Windows Server OS Configuration" or "Microsoft: Windows Server Device Discovery" Dynamic Applications automatically align to Windows servers, they trigger events and Run Book Actions that classify the server.

The following PowerShell Dynamic Applications can be used to collect performance data from Windows servers:

- Microsoft: Windows Server CPU Performance
- Microsoft: Windows Server Disk Performance
- Microsoft: Windows Server Interface Performance
- Microsoft: Windows Server IPStats Performance
- Microsoft: Windows Server Memory Performance

- Microsoft: Windows Server Performance Cache
- Microsoft: Windows Server TCPStats Performance
- Microsoft: Windows Server UDPStats Performance

> **NOTE:** The "Microsoft: Windows Server Performance Cache" Dynamic Application caches data that is consumed by all of the other performance Dynamic Applications in the list.

The following Snippet Dynamic Application, which uses PowerShell requests to collect data, can be used to collect journal data from Windows servers:

- Microsoft: Windows Server Process List

The following Snippet Dynamic Application creates a DCM+R relationship for AppDynamics, Dynatrace, and New Relic component devices with the physical Windows server device:

- Microsoft: Windows Server DCM+R Relationship

The following Dynamic Applications use PowerShell to collect data as a supplement to SL1's internal collection capabilities:

- Microsoft: Windows Server IC Cache Trigger
- Microsoft: Windows Server IC Detail
- Microsoft: Windows Server IC Filesystem Inventory
- Microsoft: Windows Server IC Filesystem Performance
- Microsoft: Windows Server IC Interface Inventory
- Microsoft: Windows Server IC Interface Performance

Microsoft: Windows Server Event Logs

The following Snippet Dynamic Applications can be used to collect data from system, application, and security event logs on Microsoft Windows servers:

- Microsoft: Windows Server Application Events
- Microsoft: Windows Server Security Events
- Microsoft: Windows Server System Events

To customize how the *Microsoft: Windows Server Event Logs* Dynamic Applications filter event logs, perform the following steps for each Dynamic Application:

1. Go to the **Dynamic Applications Manager** page (System > Manage > Applications) and search for the Dynamic Application you want to customize in the **Dynamic Application Name** column.

2. Click the wrench icon ( ) for the Dynamic Application you want to edit.

3. In the **[Snippets]** tab, click the wrench icon ( ) next to the item in the **Snippet Registry** pane.

4. In the Snippet Editor, you can edit the following fields:

   - *EVENT_ID_FILTER_INCLUDE_LIST*. Enter a list of Event IDs to include in your event logs.

   - *EVENT_TYPE_FILTER_INCLUDE_LIST*. Enter a list of Event Types to include in your event logs.

   - *EVENT_MSG_FILTER_INCLUDE_LIST*. Enter a list of Event Descriptions to include in your event logs. This field supports the use of the * wildcard character.

   - *EVENT_SRC_FILTER_INCLUDE_LIST*. Enter a list of Event Provider names to include in your event logs. This field supports the use of the * wildcard character.

5. Click the **[Save]** button.
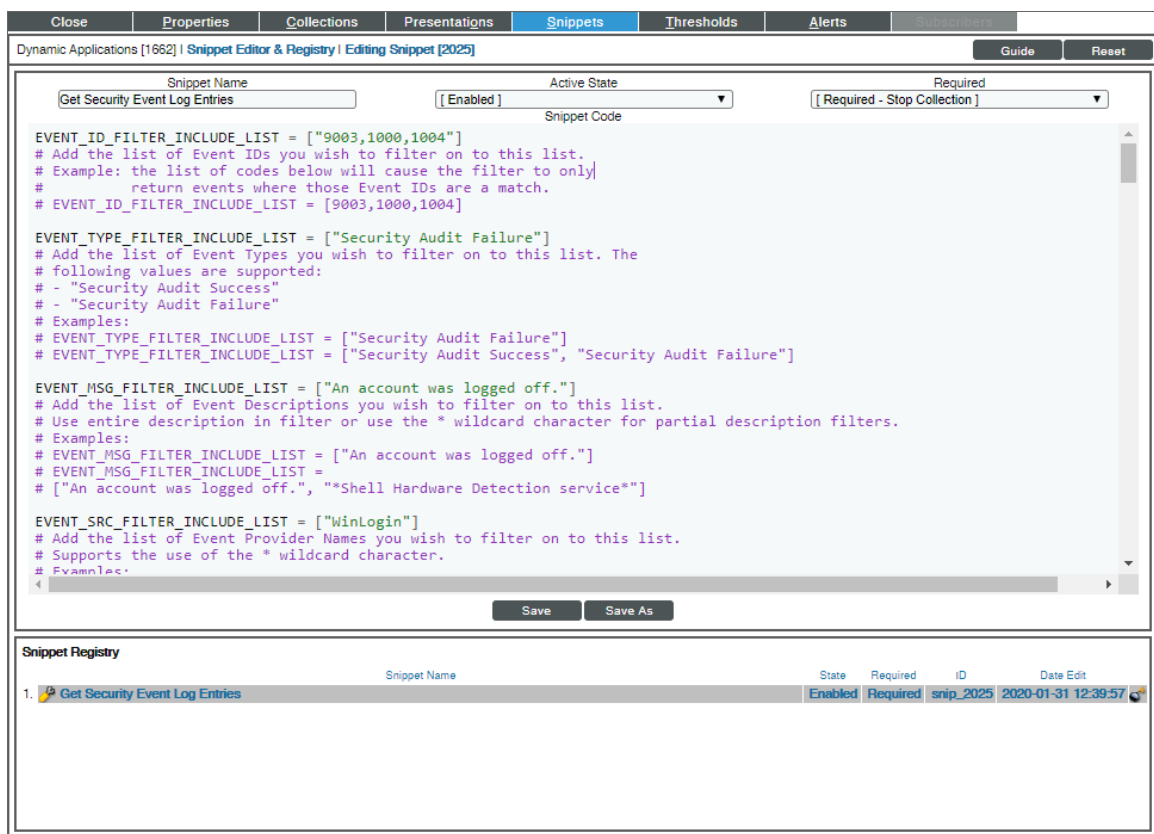
Microsoft: Windows Server Services

NOTE: The Dynamic Applications in this PowerPack support Windows Server 2008 R2, 2012, and 2012 R2.

The following PowerShell Dynamic Applications can be used to collect configuration data from a Windows server about each Windows Service running on the Windows server:

- Microsoft: Windows Server Service Configuration

Run Book Automations and Actions Associated with PowerShell Dynamic Applications for Windows Servers

You can use the following Run Book Automation Policy and Run Book Action Policy to assign a device class to each Windows device that does not support SNMP:

- Microsoft: Windows Server Device Class Alignment (Run Book Automation Policy)
- Microsoft: Windows Server Device Class Alignment (Run Book Action Policy)

Devices that do not support SNMP are assigned a device class of type "pingable".

The automation policy is configured to trigger when the "Microsoft: Windows Server OS Configuration" or "Microsoft: Windows Server Device Discovery" Dynamic Applications are aligned with a device during discovery. These Dynamic Applications collect the name of the Windows operating system and store the name in a collection object named "Edition". The Run Book Automation policy and Run Book Action policy use the value of the collection object named "Edition" to assign a device class to each Windows device that does not support SNMP.

For example, if the collection object named "Edition" contains the value "Microsoft Windows Server 2012 R2 Datacenter", the Run Book Automation policy and the Run Book Action policy will assign the device to the device class "Microsoft Windows Server 2012 R2".

Error Messages for PowerShell Collection

The following table lists error messages that SL1 can generate during PowerShell collection.

Error Message	Possible Issue(s)
Preauthentication failed while getting initial credentials	Incorrect Password (Active Directory Accounts only)
Client not found in Kerberos database	Username does not exist in Active Directory (Active Directory Accounts only)

Error Message	Possible Issue(s)
KRB5 error code 68 while getting initial credentials	Incorrect domain name (Active Directory Accounts only)
Bad HTTP response returned from server. Code 401, basic auth failed	Incorrect username/password or target server does not allow user account to perform WinRM operations.
ParseError	Incorrect port specified in credential
[Errno 111] Connection refused	Mismatch between server configuration and credential, e.g. encryption option selected but not enabled on server.
Hostname cannot be canonicalized	Forward and/or reverse name resolution are not working from the Data Collector or All-In-One Appliance
Cannot resolve network address for KDC in requested realm	Forward and/or reverse name resolution are not working from the Data Collector or All-In-One Appliance
Configuration file does not specify default realm	Forward and/or reverse name resolution are not working from the Data Collector or All-In-One Appliance
No credentials cache found	Forward and/or reverse name resolution are not working from the Data Collector or All-In-One Appliance
Server not found in Kerbers database	Forward and/or reverse name resolution are not working from the Data Collector or All-In-One Appliance

Relationships with Other Types of Component Devices

Additionally, the Dynamic Applications in the *Microsoft: Windows Server* PowerPack can automatically build relationships between Windows servers and other associated devices:

- If you discover Dynatrace devices using the Dynamic Applications in the *Dynatrace* PowerPack, SL1 will automatically create relationships between Windows servers and Dynatrace hosts.

- If you discover Cisco AppDynamics devices using the Dynamic Applications in the *Cisco: AppDynamics* PowerPack, SL1 will automatically create relationships between Windows servers and AppDynamics nodes.

- If you discover New Relic devices using the Dynamic Applications in the *New Relic APM Pro* PowerPack, SL1 will automatically create relationships between Windows servers and New Relic servers.

Chapter

5

Creating SNMP and PowerShell Credentials for Windows Devices

Overview

Use the following menu options to navigate the SL1 user interface:

- To view a pop-out list of menu options, click the menu icon ( ).

- To view a page containing all the menu options, click the Advanced menu icon ( ).

The following sections describe how to create SNMP and PowerShell credentials for Windows devices that you want to monitor with SL1, as well as how to discover component devices on Hyper-V systems:

Creating an SNMP Credential

SNMP credentials allow SL1 to access SNMP data on a managed device. SL1 uses SNMP credentials to perform discovery, run auto-discovery, and gather information from SNMP Dynamic Applications.

To create an SNMP credential:

1. Go to the **Credentials** page (Manage > Credentials).
2. Click the **[Create New]** button and then select *Create SNMP Credential*. The **Create Credential** modal page appears:

3. Supply values in the following fields:

   - *Name*. Name of the credential. Can be any combination of alphanumeric characters.
   - *All Organizations*. Toggle on (blue) to align the credential to all organizations, or toggle off (gray) and then select one or more specific organizations from the *What organization manages this service?* drop-down field to align the credential with those specific organizations.
   - *Timeout (ms)*. Time, in milliseconds, after which SL1 will stop trying to communicate with the device. The default value is *1500*.
   - *SNMP Version*. SNMP version. Choices are *SNMP V1*, *SNMP V2*, and *SNMP V3*. The default value is *SNMP V2*.
   - *Port*. The port SL1 will use to communicate with the external device or application. The default value is *161*.
   - *SNMP Retries*. Number of times SL1 will try to authenticate and communicate with the external device. The default value is *1*.

SNMP V1/V2 Settings

If you selected *SNMP V1* or *SNMP V2* in the **SNMP Version** field, complete these fields. These fields are inactive if you selected *SNMP V3*.

- **SNMP Community (Read-Only)**. The SNMP community string (password) required for read-only access of SNMP data on the remote device or application. For SNMP V1 and SNMP V2 credentials, you must supply a community string, either in this field or in the **SNMP Community (Read/Write)** field.

- **SNMP Community (Read/Write)**. The SNMP community string (password) required for read and write access of SNMP data on the remote device or application. For SNMP V1 and SNMP V2 credentials, you must supply a community string, either in this field or in the **SNMP Community (Read Only)** field.

SNMP V3 Settings

If you selected *SNMP V3* in the **SNMP Version** field, complete these fields. These fields are inactive if you selected *SNMP V1* or *SNMP V2*.

- **Security Name**. Name for SNMP authentication. This field is required.

- **Security Passphrase**. Password to authenticate the credential. This value must contain at least 8 characters. This value is required if you use a **Security Level** that includes authentication.

  In addition to alphanumeric characters, you **can** also use the following special characters in an SNMP V3 security passphrase: ? - _ = , . : # + % $ [ ] { }

  You **cannot** use the following special characters in an SNMP V3 security passphrase: & ! " ' ( ) | / \

- **Authentication Protocol**. Select an authentication algorithm for the credential. This field is required. Choices are:

  - *MD5*. This is the default value.
  - *SHA*
  - *SHA-224*
  - *SHA-256*
  - *SHA-384*
  - *SHA-512*

NOTE: The *SHA* option is SHA-128.

- ***Security Level***. Specifies the combination of security features for the credentials. This field is required. Choices are:

  - *No Authentication / No Encryption*.
  - *Authentication Only*. This is the default value.
  - *Authentication and Encryption*.

- ***Engine ID***. The unique engine ID for the SNMP agent you want to communicate with. (SNMPv3 authentication and encryption keys are generated based on the associated passwords and the engine ID.) This field is optional.

- ***Context***. A context is a mechanism within SNMPv3 (and AgentX) that allows you to use parallel versions of the same MIB objects. For example, one version of a MIB might be associated with SNMP Version 2 and another version of the same MIB might be associated with SNMP Version 3. For SNMP Version 3, specify the context name in this field. This field is optional.

- ***Privacy Protocol***. The privacy service encryption and decryption algorithm. This field is required. Choices are:

  - *DES*. This is the default value.
  - *AES-128*
  - *AES-192*
  - *AES-256*

- ***Privacy Protocol Passphrase***. Privacy password for the credential. This field is optional.

4. Click **[Save & Close]**.

NOTE: If you would like to test your credential using the Credential Tester panel, click **[Save & Test]**. For detailed instructions on using the Credential Tester panel, see the *Using the Credential Tester Panel* section.

Creating an SNMP Credential in the SL1 Classic User Interface

SNMP Credentials allow SL1 to access SNMP data on a managed device. SL1 uses SNMP credentials to perform discovery, run auto-discovery, and gather information from SNMP Dynamic Applications.

To create an SNMP credential:

1. Go to the **Credential Management** page (System > Manage > Credentials).

2. Click the [**Actions**] button and select *Create SNMP Credential*. The **Credential Editor** page appears.

3. Supply values in the following fields:

- *Profile Name*. Name of the credential. Can be any combination of alphanumeric characters.
- *SNMP Version*. SNMP version. Choices are *SNMP V1*, *SNMP V2*, and *SNMP V3*. The default value is *SNMP V2*.
- *Port*. The port SL1 will use to communicate with the external device or application. The default value is *161*.

- **Timeout (ms)**. Time, in milliseconds, after which SL1 will stop trying to communicate with the SNMP device. The default value is *1500*.

- **Retries**. Number of times SL1 will try to authenticate and communicate with the external device. The default value is *1*.

SNMP V1/V2 Settings

These fields appear if you selected *SNMP V1* or *SNMP V2* in the **SNMP Version** field. The fields are inactive if you selected SNMP V3.

- **SNMP Community (Read-Only)**. The SNMP community string (password) required for read-only access of SNMP data on the remote device or application. For SNMP V1 and SNMP V2 credentials, you must supply a community string, either in this field or in the **SNMP Community (Read/Write)** field.

- **SNMP Community (Read/Write)**. The SNMP community string (password) required for read and write access of SNMP data on the remote device or application. For SNMP V1 and SNMP V2 credentials, you must supply a community string, either in this field or in the **SNMP Community (Read Only)** field.

SNMP V3 Settings

These fields appear if you selected *SNMP V3* in the **SNMP Version** field. These fields are inactive if you selected SNMP V1 or SNMP V2.

- **Security Name**. Name for SNMP authentication. This field is required.

- **Security Passphrase**. Password to authenticate the credential. This value must contain at least 8 characters. This value is required if you use a **Security Level** that includes authentication.

- **Authentication Protocol**. Select an authentication algorithm for the credential. This field is required. Choices are:

  - *MD5*. This is the default value.
  - *SHA*
  - *SHA-224*
  - *SHA-256*
  - *SHA-384*
  - *SHA-512*

NOTE: The *SHA* option is SHA-128.

- **Security Level**. Specifies the combination of security features for the credentials. This field is required. Choices are:

    - *No Authentication / No Encryption*.
    - *Authentication Only*. This is the default value.
    - *Authentication and Encryption*.

- **SNMP v3 Engine ID**. The unique engine ID for the SNMP agent you want to communicate with. (SNMPv3 authentication and encryption keys are generated based on the associated passwords and the engine ID.) This field is optional.
- **Context Name**. A context is a mechanism within SNMPv3 (and AgentX) that allows you to use parallel versions of the same MIB objects. For example, one version of a MIB might be associated with SNMP Version 2 and another version of the same MIB might be associated with SNMP Version 3. For SNMP Version 3, specify the context name in this field. This field is optional.
- **Privacy Protocol**. The privacy service encryption and decryption algorithm. This field is required. Choices are:

    - *DES*. This is the default value.
    - *AES-128*
    - *AES-192*
    - *AES-256*

- **Privacy Protocol Passphrase**. Privacy password for the credential. This field is optional.

4. Click the **[Save]** button to save the new SNMP credential.
5. Repeat steps 1-4 for each SNMP-enabled device in your network that you want to monitor with SL1.

NOTE: When you define an SNMP Credential, SL1 automatically aligns the credential with all organizations of which you are a member.

Creating a PowerShell Credential

If you configure your Windows system to respond to PowerShell requests from SL1, you can use PowerShell Dynamic Applications to collect information from your Windows system.

All of the PowerShell Dynamic Applications include a discovery object. If you include a credential for PowerShell Dynamic Applications in the discovery session that includes your Windows system, SL1 will automatically align the appropriate PowerShell Dynamic Applications to the Windows system. For more information about creating a discovery session, see the **Discovery & Credentials** manual.

To define a PowerShell credential in SL1, you will need the following information:

- The username and password for a user on the Windows device.
- If the user is an Active Directory account, the hostname or IP address of the Active Directory server and the domain.
- Determine if an encrypted connection should be used.
- If you are using a Windows Management Proxy, the hostname or IP address of the proxy server.

To create an SSH/Key credential:

1. Go to the **Credentials** page (Manage > Credentials).
2. Click the **[Create New]** button and then select *Create Powershell Credential*. The **Create Credential** modal page appears:

3. Supply values in the following fields:

- *Name*. Name of the credential. Can be any combination of alphanumeric characters.
- *All Organizations*. Toggle on (blue) to align the credential to all organizations, or toggle off (gray) and then select one or more specific organizations from the *What organization manages this service?* drop-down field to align the credential with those specific organizations.
- *Timeout (ms)*. Time, in milliseconds, after which SL1 will stop trying to communicate with the authenticating server. For collection to be successful, SL1 must connect to the authenticating server, execute the PowerShell command, and receive a response within the amount of time specified in this field.
- *Account Type*. Type of authentication for the username and password in this credential. Choices are:
  ○ *Active Directory*. On the Windows device, Active Directory will authenticate the username and password in this credential.
  ○ *Local*. Local security on the Windows device will authenticate the username and password in this credential.

- **Hostname/IP**. Hostname or IP address of the device from which you want to retrieve data.

  - You can include the variable **%D** in this field. SL1 will replace the variable with the IP address of the device that is currently using the credential.

  - You can include the variable **%N** in this field. SL1 will replace the variable with the hostname of the device that is currently using the credential. If SL1 cannot determine the hostname, SL1 will replace the variable with the primary, management IP address for the current device.

  - You can include the prefix **HOST** or **WSMAN** before the variable **%D** in this field if the device you want to monitor uses a service principal name (for example, "HOST://%D" or "WSMAN://%D"). SL1 will use the WinRM service HOST or WSMan instead of HTTP and replace the variable with the IP address of the device that is currently using the credential.

- **Username**. Type the username for an account on the Windows device to be monitored or on the proxy server.

> NOTE: The user should not include the domain name prefix in the username for Active Directory accounts. For example, use "em7admin" instead of "MSDOMAIN\em7admin".

- **Password**. Type the password for the account on the Windows device to be monitored or on the proxy server.
- **Encrypted**. Select whether SL1 will communicate with the device using an encrypted connection:

  - Toggle on (blue) if SL1 will communicate with the device using an encrypted connection. If the connection is encrypted, when communicating with the Windows server, SL1 will use a local user account with authentication of type "Basic Auth". You must then use HTTPS and can use a Microsoft Certificate or a self signed certificate.

  - Toggle off (gray) if the connection is not encrypted. If the connection is not encrypted, when communicating with the Windows server, SL1 will not encrypt the connection.

- **Port**. Type the port number used by the WinRM service on the Windows device. This field is automatically populated with the default port based on the value you selected in the **Encrypted** field.
- **PowerShell Proxy Hostname/IP**. If you use a proxy server in front of the Windows devices you want to communicate with, type the fully-qualified domain name or the IP address of the proxy server in this field.
- **Active Directory Host/IP**. If you selected Active Directory in the **Account Type** field, type the hostname or IP address of the Active Directory server that will authenticate the credential.
- **Active Directory Domain**. If you selected Active Directory in the **Account Type** field, type the domain where the monitored Windows device resides.

4. Click **[Save & Close]**.

> NOTE: If you would like to test your credential using the Credential Tester panel, click **[Save & Test]**. For detailed instructions on using the Credential Tester panel, see the *Using the Credential Tester Panel* section.

Creating a PowerShell Credential in the SL1 Classic User Interface

To define a PowerShell credential in SL1:

1. Collect the information you need to create the credential:

   - The username and password for a user on the Windows device.
   - If the user is an Active Directory account, the hostname or IP address of the Active Directory server and the domain.
   - Determine if an encrypted connection should be used.
   - If you are using a Windows Management Proxy, the hostname or IP address of the proxy server.

2. Go to the **Credential Management** page (System > Manage > Credentials).

3. In the **Credential Management** page, click the **[Actions]** menu. Select *Create PowerShell Credential*.
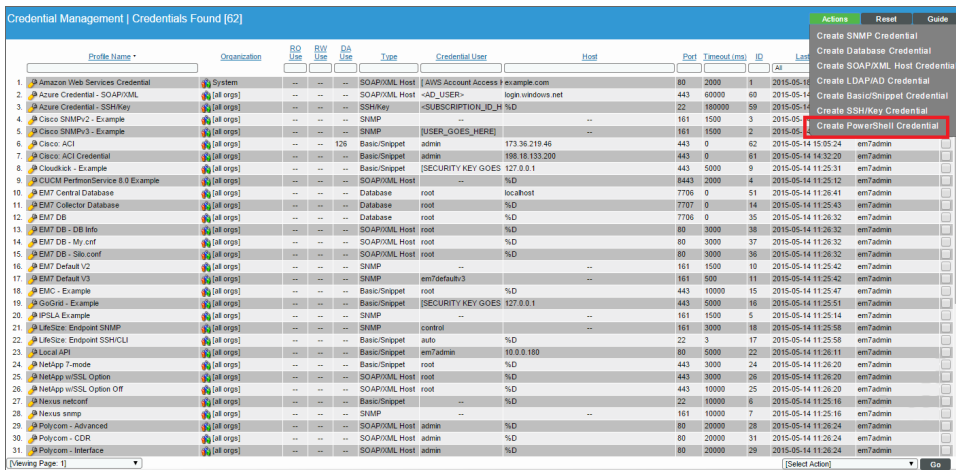
4. The **Credential Editor** page appears, where you can define the following fields:

- **Profile Name**. Name of the credential. Can be any combination of alphanumeric characters.

- **Hostname/IP**. Hostname or IP address of the device from which you want to retrieve data.

  ◦ You can include the variable **%D** in this field. SL1 will replace the variable with the IP address of the device that is currently using the credential.

  ◦ You can include the variable **%N** in this field. SL1 will replace the variable with the hostname of the device that is currently using the credential. If SL1 cannot determine the hostname, SL1 will replace the variable with the primary, management IP address for the current device.

  ◦ You can include the prefix **HOST** or **WSMAN** before the variable **%D** in this field if the device you want to monitor uses a service principal name (for example, "HOST://%D" or "WSMAN://%D"). SL1 will use the WinRM service HOST or WSMan instead of HTTP and replace the variable with the IP address of the device that is currently using the credential.

- **Username**. Type the username for an account on the Windows device to be monitored or on the proxy server.

NOTE: The user should not include the domain name prefix in the username for Active Directory accounts. For example, use "em7admin" instead of "MSDOMAIN\em7admin".

- **Encrypted**. Select whether SL1 will communicate with the device using an encrypted connection. Choices are:

  ◦ *yes*. When communicating with the Windows server, SL1 will use a local user account with authentication of type "Basic Auth". You must then use HTTPS and can use a Microsoft Certificate or a self-signed certificate.

  ◦ *no*. When communicating with the Windows server, SL1 will not encrypt the connection.

- **Port**. Type the port number used by the WinRM service on the Windows device. This field is automatically populated with the default port based on the value you selected in the **Encrypted** field.

- **Account Type**. Type of authentication for the username and password in this credential. Choices are:
  - *Active Directory*. On the Windows device, Active Directory will authenticate the username and password in this credential.
  - *Local*. Local security on the Windows device will authenticate the username and password in this credential.

- **Timeout (ms)**. Type the time, in milliseconds, after which SL1 will stop trying to collect data from the authenticating server. For collection to be successful, SL1 must connect to the authenticating server, execute the PowerShell command, and receive a response within the amount of time specified in this field.

- **Password**. Type the password for the account on the Windows device to be monitored or on the proxy server.

- **PowerShell Proxy Hostname/IP**. If you use a proxy server in front of the Windows devices you want to communicate with, type the fully-qualified domain name or the IP address of the proxy server in this field.

- **Active Directory Hostname/IP**. If you selected Active Directory in the **Account Type** field, type the hostname or IP address of the Active Directory server that will authenticate the credential.

- **Domain**. If you selected Active Directory in the **Account Type** field, type the domain where the monitored Windows device resides.

5. To save the credential, click the **[Save]** button. To clear the values you set, click the **[Reset]** button.

Testing Windows Credentials

Credential Tests define a series of steps that SL1 can execute on-demand to validate whether a credential works as expected. This section describes the SNMP and PowerShell Credential Tests that are included in the default installation of SL1.

SNMP Credential Test

The SNMP Credential Test can be used to test an SNMP credential for connectivity. The SNMP Credential Test performs the following steps:

- **Test Reachability**. Performs an ICMP ping request to the host specified in the credential.

- **Test Port Availability**. Performs an NMAP request to the UDP port specified in the credential on the host specified in the credential.

- **Test SNMP Availability**. Attempts an SNMP getnext request to .1.3.6.1 using the credential.

PowerShell Credential Test

The PowerShell Credential Test can be used to test a PowerShell credential for connectivity. The PowerShell Credential Test performs the following steps:

- **Test Reachability**. Performs an ICMP ping request to the host specified in the credential.

- **Test Port Availability**. Performs an NMAP request to the TCP port specified in the credential on the host specified in the credential.

- **Test Name Resolution**. Performs an nslookup request on the host specified in the credential.

- **Test Kerberos**. If the credential does not specify local authentication, attempts to acquire a kerberos ticket using the credential.

- **Test WinRM Connection**. Attempts a WinRM connection using the credential.

- **Execute PowerShell Cmdlet**. Attempts to execute the 'Get-WmiObject Win32_Process | Select Name' PowerShell Cmdlet using the credential.

Running a Windows Credential Test

You can test a credential from the **Credentials** page using a predefined credential test.

To run a credential test from the **Credentials** page:

1. Go to the **Credentials** page (Manage > Credentials).
2. Do one of the following:

   - Click the **[Test Credentials]** button, or
   - Click the **Actions** button (⚊) of the credential that you want to test, and then select *Test*.

3. The **Credential Test Form** modal page appears. Fill out the following fields on this page:

   - **Credential**. Select the credential to test. This drop-down list includes only credentials that you have access to. (If you clicked the **Actions** button (⚊) and then selected *Test* for a specific credential, then this field is read-only.)

   - **Select Credential Test**. Select a credential test to run. This drop-down list includes the *ScienceLogic Default Credential Tests*, credential tests included in any PowerPacks that have been optionally installed on your system, and credential tests that users have created on your system.

   - **Collector**. Select the All-In-One Appliance or Data Collector that will run the test.

   - **IP or Hostname to Test**. Type a hostname or IP address that will be used during the test. For example, if you are testing an SNMP credential, the hostname/IP address you supply will be used to perform a test SNMP request.

4. Click **[Run Test]** button to run the credential test. The **Testing Credential** window appears:

The **Testing Credential** window displays a log entry for each step in the credential test. The steps performed are different for each credential test. The log entry for each step includes the following information:

- *Step*. The name of the step.
- *Description*. A description of the action performed during the step.
- *Log Message*. The result of the step for this execution of the credential test.
- *Status*. Whether the result of this step indicates the credential and/or the network environment is configured correctly (Passed) or incorrectly (Failed).
- *Step Tip*. Mouse over the question mark icon (?) to display the tip text. The tip text recommends what to do to change the credential and/or the network environment if the step has a status of "Failed".

To run a Windows credential test using the Credential Tester panel:

1. While defining a credential, supply values in the required fields on the **Create Credential** page.
2. Click the **[Save & Test]** button. This activates the Credential Tester fields.
3. In the Credential Tester panel, supply values in the following fields:

   - *Select Credential Test*. Select a credential test to run. This drop-down list includes the ScienceLogic Default Credential Tests, credential tests included in any PowerPacks that have been optionally installed on your system, and credential tests that users have created on your system.
   - *Select Collector*. Select the All-In-One Appliance or Data Collector that will run the test.
   - *IP or Hostname to test*. Type a hostname or IP address that will be used during the test. For example, if you are testing an SNMP credential, the hostname/IP address you supply will be used to perform a test SNMP request.

4. Click **[Run Test]** button to run the credential test. The **Testing Credential** window appears:

The **Testing Credential** window displays a log entry for each step in the credential test. The steps performed are different for each credential test. The log entry for each step includes the following information:

- **Step**. The name of the step.
- **Description**. A description of the action performed during the step.
- **Log Message**. The result of the step for this execution of the credential test.
- **Status**. Whether the result of this step indicates the credential and/or the network environment is configured correctly (Passed) or incorrectly (Failed).
- **Step Tip**. Mouse over the question mark icon (?) to display the tip text. The tip text recommends what to do to change the credential and/or the network environment if the step has a status of "Failed".

Running a Windows Credential Test in the SL1 Classic User Interface

To run a Windows credential test from the **Credential Management** page:

1. Go to the **Credential Management** page (System > Manage > Credentials).
2. Click the **[Actions]** menu, and then select *Test Credential*. The **Credential Tester** modal page appears:

3. Supply values in the following fields:

   - **Test Type**. Select a credential test to run.
   - **Credential**. Select the credential you want to test. This drop-down list includes only credentials that you have access to that can be tested using the selected credential test.

- ***Hostname/IP***. Enter a hostname or IP address that will be used during the test. For example, if you are testing an SNMP credential, the hostname/IP address you supply will be used to perform a test SNMP request.
- ***Collector***. Select the All-In-One Appliance or Data Collector that will run the test.

4. Click the **[Run Test]** button to run the credential test. The **Test Credential** window appears:

	Step	Description	Log Message	Status	
**Test Credential	Test execution complete**					
1	Test Reachability	Check to see if the device is reachable using ICMP	The device is reachable using ICMP. The average response time is 0.397ms	Passed	
2	Test Port Availability	Check to see if the SNMP port is open	Port 161 is open	Passed	
3	Test SNMP Availability	Check to see if a walk of SNMP will return results	The SNMP SysName is ScienceLogic EM7 G3 - All-In-One	Passed	

The **Test Credential** window displays a log entry for each step in the credential test. The steps performed are different for each credential test. The log entry for each step includes the following information:

- ***Step***. The name of the step.
- ***Description***. A description of the action performed during the step.
- ***Log Message***. The result of the step for this execution of the credential test.
- ***Status***. Whether the result of this step indicates the credential and/or the network environment is configured correctly (Passed) or incorrectly (Failed).
- ***Step Tip***. Mouse over the question mark icon ( ) to display the tip text. The tip text recommends what to do to change the credential and/or the network environment if the step has a status of "Failed".

5. Optionally, you can click the **[Execute Discovery Session]** button to run a discovery session using the ***Credential***, ***Hostname/IP***, and ***Collector*** you selected in the **Credential Tester** modal page.

Discovering Component Devices on Hyper-V Systems

The *Microsoft: Hyper-V Server* PowerPack includes two Dynamic Applications that allow SL1 to collect information about the virtual machines running on a Hyper-V system.

To discover the virtual machines on a Hyper-V system as component devices, align the following two Dynamic Applications with a Hyper-V system:

- Microsoft: Hyper-V Guest Configuration Cache
- Microsoft: Hyper-V Gust Discovery
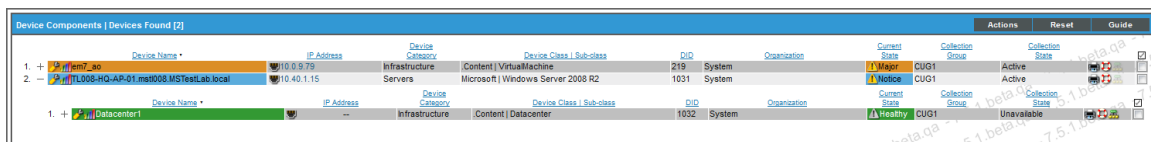
When these Dynamic Applications are aligned to a Hyper-V system, the platform will automatically create a device record for each virtual machine. The platform will also automatically align other Dynamic Applications from the *Microsoft: Hyper-V Server* PowerPack to each virtual machine.

Viewing Component Devices

When SL1 performs collection for the "Microsoft Hyper-V Guest Configuration Cache" and "Microsoft Hyper-V Guest Discovery" Dynamic Applications, SL1 will create component devices for the virtual machines on the Hyper-V and align other Dynamic Applications to those component devices. All component devices appear in the **Device Manager** page just like devices discovered using the ScienceLogic discovery process.

In addition to the **Device Manager** page, you can view the Hyper-V system and all associated component devices in the following places in the user interface:

- The **Device Components** page (Devices > Device Components or Registry > Devices > Device Components in the SL1 classic user interface) displays a list of all root devices and component devices discovered by the platform. The **Device Components** page displays all root devices and component devices in an indented view, so you can easily view the hierarchy and relationships between child devices, parent devices, and root devices. To view the component devices associated with a Hyper-V system, find the Hyper-V system and select its plus icon (**+**):

- The **Component Map** page (Classic Maps > Device Maps > Components or Views > Device Maps > Components in the SL1 classic user interface) allows you to view devices by root node and view the relationships between root nodes, parent components, and child components in a map. This makes it easy to visualize and manage root nodes and their components. SL1 automatically updates the **Component Map** as new component devices are discovered. SL1 also updates each map with the latest status and event information. To view the map for a Hyper-V system, select the Hyper-V system from the left NavBar. To learn more about the **Component Map** page, see the *Views* manual.
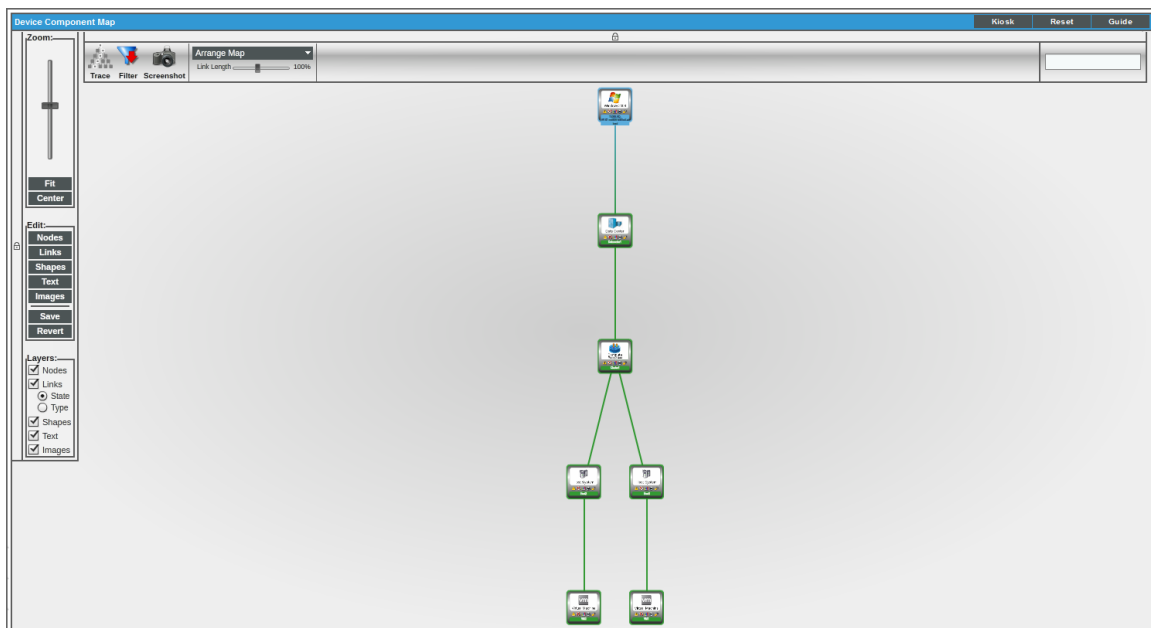
Manually Aligning the Microsoft: Print Server Dynamic Application

The Microsoft: Print Server Dynamic Application must be manually aligned. To do so, perform the following steps:

1. Find your Windows device in the **Device Manager** page (Registry > Devices > Device Manager and click its wrench icon ( ).

2. From the **Device Properties** page for the Windows system, click the **[Collections]** tab. The **Dynamic Application Collections** page appears.

3. Click the **[Action]** button and then select *Add Dynamic Application*. The **Dynamic Application Alignment** page appears:
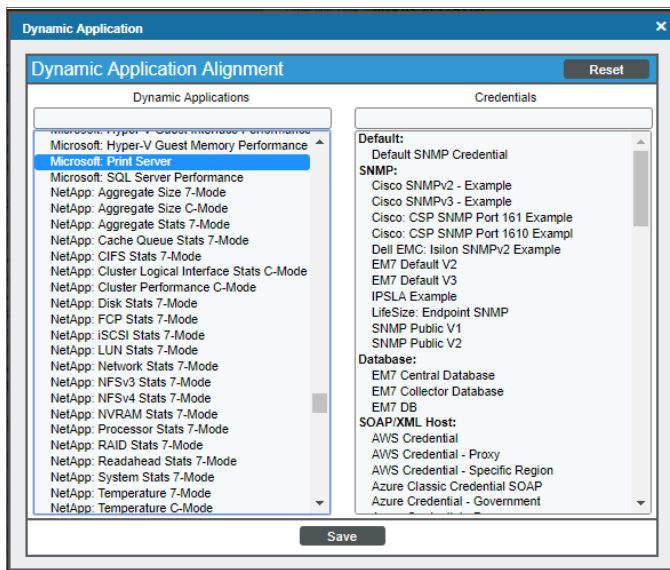
4. In the **Dynamic Applications** field, select the "Microsoft: Print Server" Dynamic Application.

5. In the **Credentials** field, select the credential you created for your Windows system.

6. Click the **[Save]** button.

Chapter

6

Concurrent PowerShell Collection

Overview

This chapter describes how to configure and use concurrent PowerShell collection. Concurrent PowerShell collection allows multiple collection tasks to run at the same time with a reduced load on Data Collectors. Concurrent PowerShell collection also reduces missed polls and data gaps because collection will execute more quickly. As a result, Data Collectors can collect more data using fewer system resources. The PowerShell Collector is an independent service running as a container on a Data Collector.

Use the following menu options to navigate the SL1 user interface:

- To view a pop-out list of menu options, click the menu icon (▤).

- To view a page containing all the menu options, click the Advanced menu icon (⋯).

This chapter covers the following topics:

Prerequisites

The following prerequisites are required to use concurrent PowerShell collection:

- SL1 version 10.2.0 or greater
- *Microsoft: Windows Server* PowerPack version 110 or greater
- *SL1: Concurrent PowerShell Monitoring* PowerPack version 100 or greater

Scope

When the concurrent PowerShell collection service is enabled, PowerShell Configuration and PowerShell Performance Dynamic Applications are sent to the service.

The following PowerPacks can use the concurrent PowerShell collection service:

- Microsoft: Active Directory Server
- Microsoft: DHCP Server
- Microsoft: DNS Server
- Microsoft: Exchange Server
- Microsoft: IIS Server
- Microsoft: Lync Server 2013
- Microsoft: SharePoint Server
- Microsoft: SQL Server
- Microsoft: Hyper-V Server (partially)
- Microsoft: Windows Server (partially)

Enabling and Disabling Concurrent PowerShell for Collector Groups

> **NOTE:** To enable and receive SL1 events related to concurrent PowerShell, you will need to install the *SL1: Concurrent PowerShell Monitor* PowerPack.

To improve the process of collecting data via PowerShell and to collect metrics, you can enable concurrent PowerShell collection. You can enable one or more collector groups to use concurrent PowerShell collection.

> **CAUTION:** By default, a loopback to 127.0.0.1 is configured on the collector with the line `localhost localhost.localdomain localhost4 localhost4.localdomain4` in the `/etc/hosts` file. If this line is removed, concurrent PowerShell collection will not function properly.

> **NOTE**: Concurrent PowerShell collection is for PowerShell Performance and Performance Configuration Dynamic Application types and does not include Snippet Dynamic Applications which happen to run PowerShell commands.

Enabling Concurrent PowerShell on All Collector Groups

To enable concurrent PowerShell collection service for all collector groups:

1. Go to the **Database Tool** page (System > Tools > DB Tool).
2. Enter the following in the *SQL Query* field:

```
INSERT INTO master.system_custom_config (`field`, `field_value`) VALUES ('enable_
powershell_service', '1');
```

Disabling Concurrent PowerShell on All Collector Groups

To disable concurrent PowerShell collection service for all collector groups:

1. Go to the **Database Tool** page (System > Tools > DB Tool).
2. Enter the following in the *SQL Query* field:

```
UPDATE master.system_custom_config SET field_value=0 where field='enable_powershell_
service';
```

Enabling Concurrent PowerShell on a Specific Collector Group

To enable concurrent PowerShell collection for a specific collector group:

1. Go to the **Database Tool** page (System > Tools > DB Tool).
2. Enter the following in the *SQL Query* field:

```
INSERT INTO master.system_custom_config (`field`, `field_value`, `cug_filter`)
VALUES ('enable_powershell_service_CUGx', '1', 'collector_group_ID');
```

where:

collector_group_ID is the collector group ID. You can find this value in the **Collector Group Management** page (System > Settings > Collector Groups).

Disabling Concurrent PowerShell on a Specific Collector Group

To disable concurrent PowerShell collection for a specific collector group:

1. Go to the **Database Tool** page (System > Tools > DB Tool).
2. Enter the following in the *SQL Query* field:

```
UPDATE master.system_custom_config SET field_value=0 where field='enable_powershell_
service_CUGx';
```

where:

collector_group_ID is the collector group ID. You can find this value in the **Collector Group Management** page (System > Settings > Collector Groups).

The SL1: Concurrent PowerShell Monitoring PowerPack
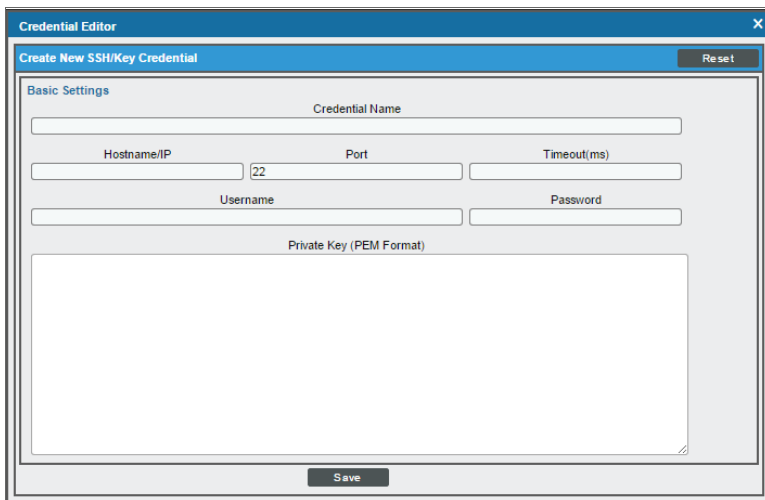
The *SL1: Concurrent PowerShell Monitoring* PowerPack includes a device template and two Dynamic Applications that use SSH to monitor collectors with concurrent PowerShell enabled.

- The "ScienceLogic: PowerShell Collector Performance" Dynamic Application is an optional Dynamic Application used for troubleshooting.
- The "ScienceLogic: PowerShell Service Log Parser" Dynamic Application parses the log file from the PowerShell servers and converts errors into events aligned to the related device.
- The "SL1: Concurrent PowerShell Monitoring" device template can be use to align multiple Data Collectors to the "ScienceLogic: PowerShell Service Log Parser" Dynamic Application.

Aligning the "ScienceLogic: PowerShell Service Log Parser" Dynamic Application

To align the "ScienceLogic: PowerShell Service Log Parser" Dynamic Application, first you must create an SSH/Key credential:

1. Go to the **Credential Management** page (System > Manage > Credentials).
2. In the **Credential Management** page, click the **[Actions]** menu. Select *Create SSH/Key Credential*.

3. The **Credential Editor** modal page appears. In this page, define the new SSH/Key credential using a valid username and password or SSH key for SL1 collectors:

- *Credential Name*. Name of the credential. Can be any combination of alphanumeric characters.
- *Hostname/IP*. Hostname or IP address of the device from which you want to retrieve data.

    ○ You can include the variable %D in this field. SL1 will replace the variable with the IP address of the current device (device that is currently using the credential).

    ○ You can include the variable %N in this field. SL1 will replace the variable with hostname of the current device (device that is currently using the credential). If SL1 cannot determine the hostname, SL1 will replace the variable with the primary, management IP address for the current device.

- *Port*. Port number associated with the data you want to retrieve.

NOTE: The default TCP port for SSH servers is 22.

- *Timeout (ms)*. Time, in milliseconds, after which SL1 will stop trying to communicate with the authenticating server.
- *Username*. Username for the Data Collector to be monitored.
- *Password*. Password for the Data Collector to be monitored.
- *Private Key (PEM Format)*. Enter an SSH private key for the SL1 Data Collector, in PEM format.

4. Click the **[Save]** button to save the new SSH/Key credential.

Configuring the Device Template

After creating the SSH/Key credential, you will need to configure the device template included in the PowerPack.

To configure the device template:

1. Go to the **Configuration Templates** page (Registry > Devices > Templates).

2. Locate the "SL1: Concurrent PowerShell Monitoring" sample template and click its wrench icon ( ). The **Device Template Editor** modal page appears.

3. Type a new name for the device template in the **Template Name** field so the sample template is not overwritten.

4. Click the **[Dyn Apps]** tab. The **Editing Dynamic Application Subtemplates** page appears.

5. In the **Subtemplate Selection** pane, select the "ScienceLogic: PowerShell Service Log Parser" Dynamic Application.

6. In the **Credentials** drop-down list, select the SSH/Key credential that you created.

7. Click **[Save As]**.

Applying the Device Template

If your Data Collector devices already exist on your SL1 system, perform the following steps to apply the device template:

1. Go to the **Device Manager** page (Registry > Devices > Device Manager) and select the checkbox for each of your Data Collector devices.

2. In the **Select Action** menu, select *MODIFY By Template* and then click **[Go]**.

3. In the **Device Template Editor**, select the template you created in the **Template** field.

4. Click **[Apply]**.

If your devices have not yet been discovered, perform the following steps to discover the devices and apply the device template:

1. Go to the **Discovery Control Panel** page (System > Manage > Classic Discovery) and click **[Create]**.

2. Supply values in the following fields:

    - *Name*. Type a name for the discovery session.

    - *Description*. Optionally, type a description of the discovery session.

    - *IP Address/Hostname Discovery List*. Provide a list of IP addresses for your Data Collectors.

    - *SNMP Credentials*. Select *EM7 Default V2*.

    - *Model Devices*. Select this checkbox.

    - *Apply Device Template*. Selec the device template that you created.

    - *Log All*. Select this checkbox.

4. Click the **[Save]** button **to save the discovery session**. Close the **Discovery Session Editor** page.

5. In the **Discovery Control Panel** page, click the **[Reset]** button. The new discovery session will appear in the **Session Register** pane.

6. To launch the new discovery session, click its **Queue this Session** icon ( ).

7. If no other discovery sessions are currently running, the session will be executed immediately. If another discovery session is currently running, your discovery session will be queued for execution.

Scale Recommendations

The following recommendations increase the number of Windows Servers the concurrent PowerShell collector can support:

- In the **Device Properties** page for all Windows Server devices (Registry > Devices > wrench icon), unselect the *Dynamic Discovery* checkbox. Alternatively, this can be set in bulk using a device template and device group. This prevents nightly discovery from attempting to align Dynamic Applications with a discovery object to all the devices on the collector, which does not use the concurrent PowerShell collector and will dramatically limit the number of Windows Server devices that can be monitored.

- Do not select any credentials in the discovery session used to discover new Windows Servers. Instead, use a template that includes unselecting the **Dynamic Discovery** checkbox and includes the desired Dynamic Applications with the appropriate credential aligned. When a credential is selected in the Discovery Session, it will attempt to align Dynamic Applications that include a discovery object, which does not use the concurrent PowerShell collector and will dramatically limit the number of Windows Server devices that can be monitored. The *Microsoft: Windows Server* PowerPack includes the "Microsoft: Windows Server Discovery Template" that you can use be used to create your template.

For information on creating and using device templates, see the **Device Groups and Device Templates** manual.

Additional Scale Tips

- Limit the use of the "Microsoft: Windows Server Services" PowerPack, as using this Dynamic Application can reduce the number of servers a collector can support by 40%. If you do use this PowerPack, consider slowing down the **Poll Frequency** of the "Microsoft: Windows Server Services" Dynamic Application.

- Limit the use of the "Microsoft: Windows Server Event Logs" PowerPack as it does not work with the concurrent PowerShell collector.

- Use the *Microsoft: SQL Server* PowerPack instead of the *Microsoft: SQL Server Enhanced* PowerPack. The *Microsoft: SQL Server Enhanced* PowerPack does not work with the concurrent PowerShell collector.

- Disable Dynamic Applications that are not providing information required to meet your Service Level Agreements. There is an enhancement in caching included with concurrent PowerShell collection that will not send a PowerShell request from a cache-producing Dynamic Application unless at least one Dynamic Application is asking for that data. Disabling a cache-consuming Dynamic Application will also disable the cache producer from collecting that data. For example, the following Dynamic Applications are now disabled by default, as they are more diagnostic in nature and may not be required for routine monitoring:

  - Microsoft: Windows Server IPStats Performance

  - Microsoft: Windows Server TCPStats Performance

  - Microsoft: Windows Server UDPStats Performance

- Slow down the **Poll Frequency** for Dynamic Applications that do not include events. For example, the *Microsoft: Windows Server* PowerPack's Configuration Dynamic Applications used to be set to run every two hours and are now set to run every 12 hours.

Chapter

7

Executing the SL1 Agent with Windows PowerShell

Overview

Use the following menu options to navigate the SL1 user interface:

- To view a pop-out list of menu options, click the menu icon (▤).

- To view a page containing all the menu options, click the Advanced menu icon (⋯).

> **NOTE:** Monitoring with the SL1 Agent is available only on SL1 Extended.

The following sections provide an overview of local Agent execution on Windows devices with PowerShell:

What is an SL1 Agent?

The **SL1 agent** is a program that you can install on a device monitored by SL1. The SL1 agent collects data from the device and pushes that data back to SL1.

Similar to a Data Collector or Message Collector, the SL1 Agent collects data about infrastructure and applications.

The agent can be configured to communicate with either the Message Collector or Compute Cluster.

For more information, see the **Monitoring with the SL1 Agent** manual .

Agent-Compatible PowerPacks

The following PowerPacks include the SL1 Agent PowerShell Default credential and SL1 Agent device template, which you can use to execute the SL1 Agent on Windows devices with PowerShell:

- Microsoft: Windows Server
- SL1 Agent Templates for Microsoft PowerPacks, which includes templates for the following:

  - Microsoft: DHCP Server
  - Microsoft: DNS Server
  - Microsoft: Exchange Server
  - Microsoft: IIS Server
  - Microsoft: Lync Server
  - Microsoft: SharePoint Server
  - Microsoft: SQL Server
  - Microsoft: Windows Server

The Credential for the SL1 Agent

The "SL1 Agent PowerShell Default" credential does not need to be configured and can be used as-is. You can find the credential in the **Credentials** page (Manage > Credentials), or the **Credential Management** page (System > Manage > Credentials) in the SL1 classic user interface:

Configuring the SL1 Agent Device Template

A *device template* allows you to save a device configuration and apply it to multiple devices. Windows PowerPacks include a device template for executing the SL1 Agent with PowerShell. If you apply this device template during discovery, SL1 aligns the appropriate Dynamic Applications to the discovered PowerShell device.

This device template does not need to be edited and will work as-is, unless you would like to remove a Dynamic Application from the template. To remove any Dynamic Applications you may not need:

1. Go to the **Configuration Templates** page (Devices > Templates, or Registry > Devices > Templates in the SL1 classic user interface).

2. Locate the SL1 Agent template (for example, "SL1 Agent for Microsoft: Windows Server Template") and click its wrench icon ( ). The **Device Template Editor** page appears.

3. Click the **[Dyn Apps]** tab. The **Editing Dynamic Application Subtemplates** page appears:
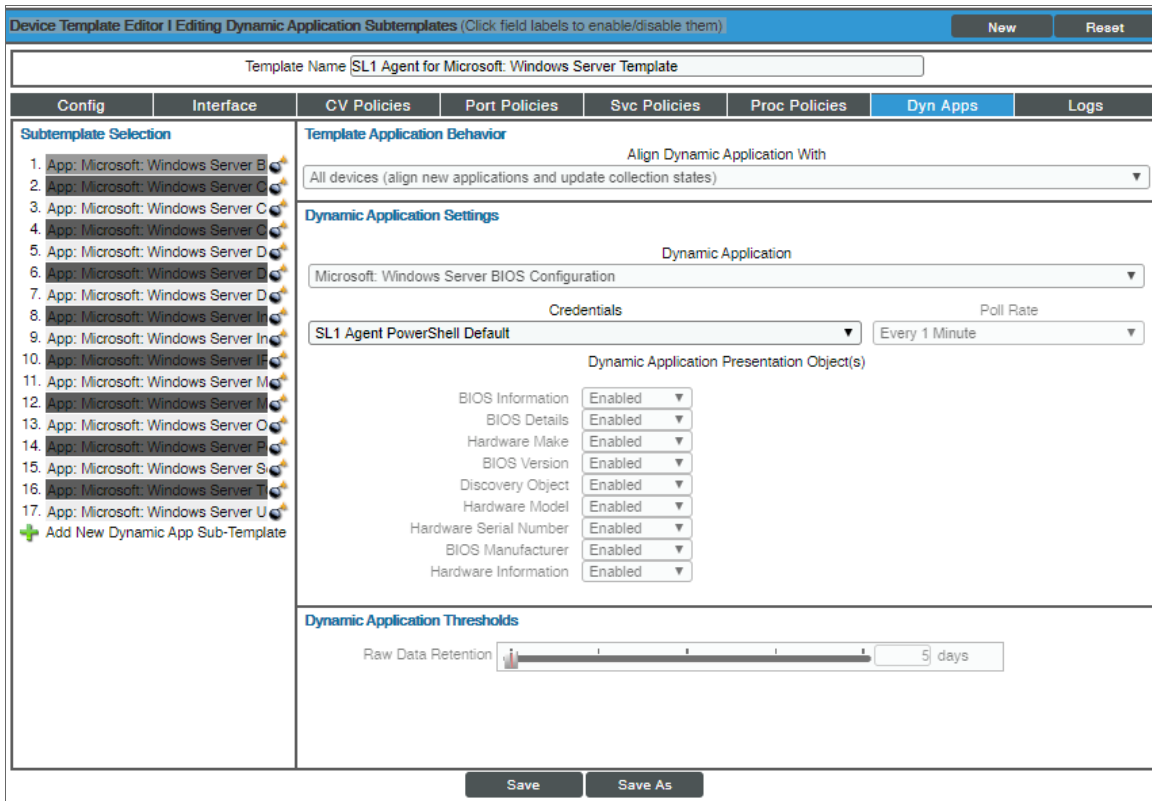
5. To remove a Dynamic Application listed in the **Subtemplate Selection** section on the left side of the page, click it's bomb icon ( ) and then click **[OK]** when asked to confirm.

6. Click **[Save]**.

Chapter

8

Windows Dashboards

Overview

Use the following menu options to navigate the SL1 user interface:

- To view a pop-out list of menu options, click the menu icon (▤).

- To view a page containing all the menu options, click the Advanced menu icon (⋯).

The following sections describe how to install the dashboards included in SL1 for Microsoft servers and a description of each:

Installing the Microsoft Server Dashboards

The following PowerPacks contain dashboards for Microsoft servers:

- Microsoft: Active Directory Server Dashboards
- Microsoft: DNS Server Dashboards
- Microsoft: Exchange Server 2010 Dashboards
- Microsoft: Exchange Server 2013 Dashboards
- Microsoft: IIS Server Dashboards
- Microsoft: Lync Server 2010 Dashboards
- Microsoft: Skype for Business Dashboards
- Microsoft: SQL Server Dashboards

To view these dashboards in SL1, you must first install the corresponding PowerPack. To do so:

1. Go to the **PowerPack Manager** page (System > Manage > PowerPacks).
2. Click the **[Actions]** button, then select *Install PowerPack*. The **Imported PowerPacks** modal page appears.
3. Use the search filter in the **PowerPack Name** column heading to locate the PowerPack you want to install. To do so, enter text to match, including special characters, and the **Imported PowerPacks** modal page displays only PowerPacks that have a matching name.
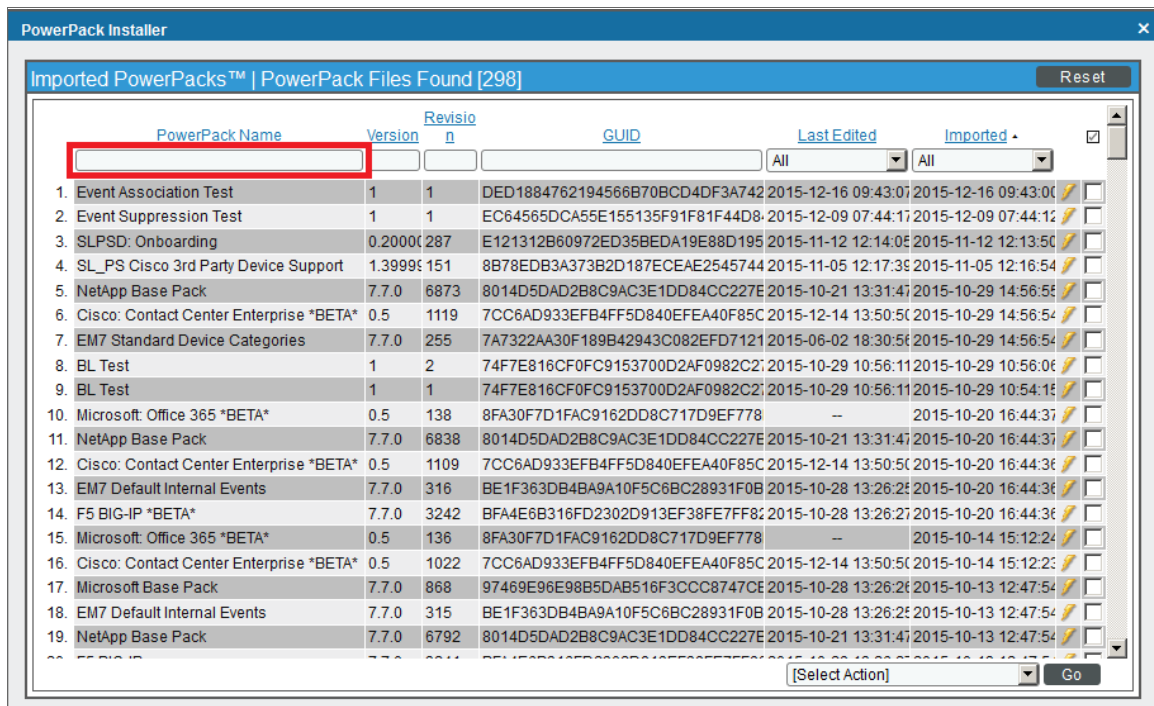
4. Click the lightning-bolt icon ( ) for the PowerPack that you want to install.

5. The **Install PowerPack** modal page appears. To install the PowerPack, click **[Install]**.
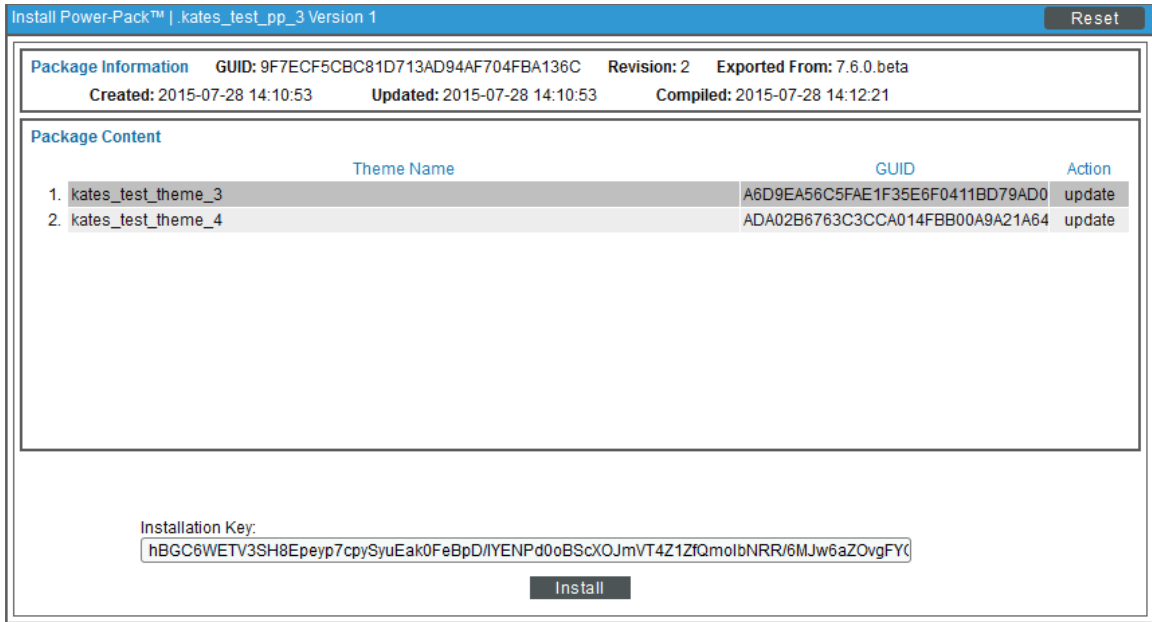
6. The PowerPack now appears in the **PowerPack Manager** page. The contents of the PowerPack are automatically installed in your SL1 System.

Microsoft: Active Directory Server Performance

The Microsoft: Active Directory Server Performance dashboard provides an overview of the health and performance of a selected Active Directory server.
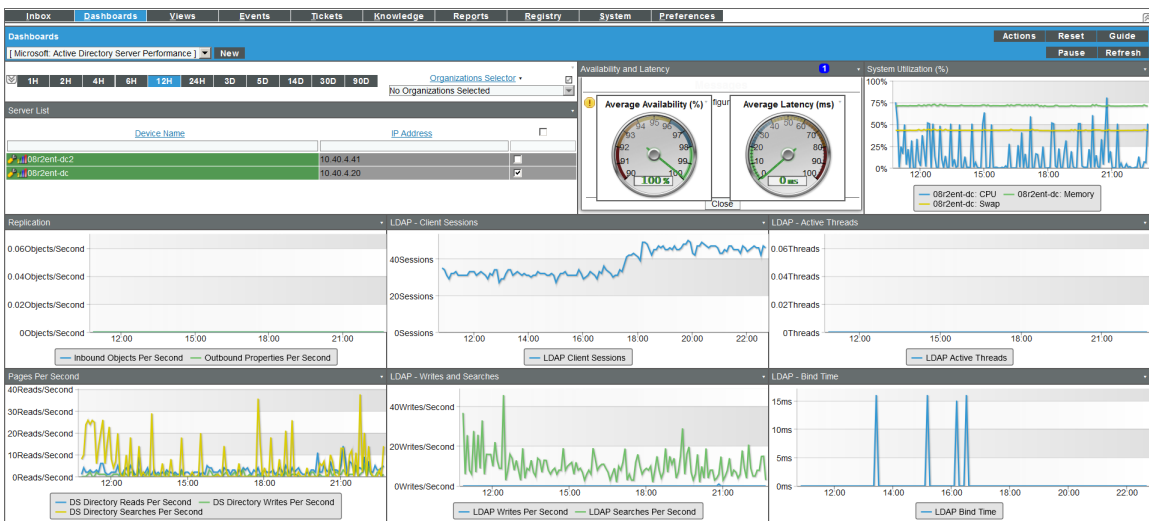
Context Quick Selector. This widget contains buttons for time span preset and the Organizations Selector.

- *Time span presets*. Users select the time span over which they want to view data. Selections range from one hour to 90 days.
- *Organizations Selector*. This drop-down list allows a user to select specific organizations for which they want to view data. This field filters the list of Active Directory servers that appear in the **Server List** widget.

Server List. This widget displays a list of Active Directory servers. Selecting a server drives the context for the other widgets in the dashboard.

Availability and Latency. This widget displays two gauges:

- The availability of the selected Active Directory server, in percent.
- The latency of the selected Active Directory server, in milliseconds.

System Utilization (%). This widget displays a line graph. The line graph displays memory usage, virtual-memory usage, and CPU usage for the selected Active Directory server during the selected duration. Each parameter is represented by a color-coded line.

- The y axis displays usage, in percent.
- The x axis displays time. The increments vary, depending upon the date range selected in the **Context Quick Selector**widget.
- Mousing over any point in any line displays the average value at that time-point.
- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected Active Directory server.

Replication. Replication is the process by which the changes that are made on one domain controller are synchronized with and written to all other domain controllers in the domain or forest. The Replication widget displays a line graph. The line graph displays information about data that is replicated from the current Active Directory server to other Active Directory servers (the Outbound Properties Per Second) and information about data that is replicated from other Active Directory server to the current Active Directory server (Inbound Objects Per Second).

- The y axis displays objects per second.
- The x axis displays time. The increments vary, depending upon the date ranges selected in the**Context Quick Selector**widget.
- Mousing over any point in any line displays the average value at that time-point.
- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected Active Directory server.

LDAP - Client Sessions. This widget displays the number of connected LDAP client sessions over time.

- The y axis displays number of sessions .
- The x axis displays time. The increments vary, depending upon the date range selected in the **Context Quick Selector** widget.
- Mousing over any point in any line displays the average value at that time-point.

- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected Active Directory server.

LDAP - Active Threads. This widget displays the number of threads in use by the LDAP subsystem of the local directory service.

- The y axis displays number of threads.
- The x axis displays time. The increments vary, depending upon the date range selected in the **Context Quick Selector** widget.
- Mousing over any point in any line displays the average value at that time-point.
- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected Active Directory server.

Pages Per Second. This widget displays a line graph. The line graph displays DS (domain server) directory reads per second, DS directory writes per second, and DS directory searches per second. Each parameter is represented by a color-coded line.

- The y axis displays usage, in percent.
- The x axis displays time. The increments vary, depending upon the date range selected in the **Context Quick Selector** widget.
- Mousing over any point in any line displays the average value at that time-point.
- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected Active Directory server.

LDAP - Writes and Searches. This widget displays a line graph. The line graph displays LDAP writes per second and LDAP searches per second. Each parameter is represented by a color-coded line.

- The y axis displays writers per second and searches per second.
- The x axis displays time. The increments vary, depending upon the date range selected in the **Context Quick Selector** widget.
- Mousing over any point in any line displays the average value at that time-point.
- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected Active Directory server.

LDAP - Bind Time. This widget displays a line graph. The line graph displays the time required for completion of each successful LDAP binding.

- The y axis displays duration in milliseconds.
- The x axis displays time. The increments vary, depending upon the date range selected in the **Context Quick Selector** widget.
- Mousing over any point in any line displays the average value at that time-point.
- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected Active Directory server.

Microsoft: DNS Server Performance

The Microsoft: DNS Server Performance dashboard provides an overview of the health and performance of a selected DNS server.
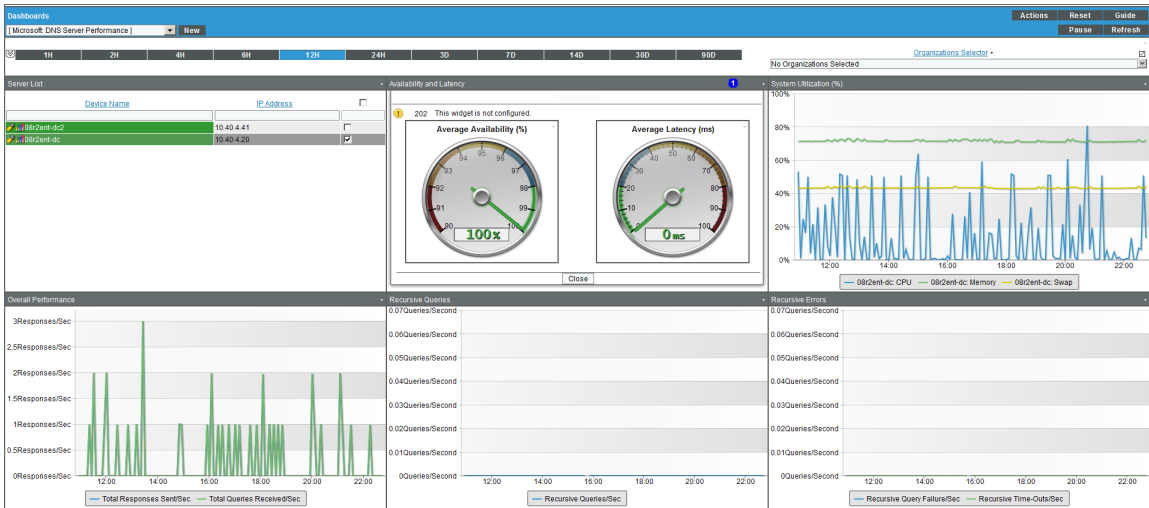
Context Quick Selector. This widget contains buttons for time span presets and the Organizations Selector.

- *Time span presets*. Users select the time span over which they want to view data. Selections range from one hour to 90 days.
- *Organizations Selector*. This drop-down list allows a user to select specific organizations for which they want to view data. This field filters the list of DNS servers that appear in the **Server List** widget.

Server List. This widget displays a list of DNS servers. Selecting a server drives the context for the other widgets in the dashboard.

Availability and Latency. This widget displays two gauges:

- The availability of the selected DNS server, in percent.
- The latency of the selected DNS server, in milliseconds.

System Utilization (%). This widget displays a line graph. The line graph displays memory usage, virtual-memory usage, and CPU usage for the selected DNS server during the selected duration. Each parameter is represented by a color-coded line.

- The y axis displays usage, in percent.
- The x axis displays time. The increments vary, depending upon the date range selected in the **Context Quick Selector** widget.
- Mousing over any point in any line displays the average value at that time-point.

- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected DNS server.

Overall Performance. This widget displays a line graph. The line graph displays Total Responses Sent per Second and Total Queries Received per Second. Each parameter is represented by a color-coded line.

- The y axis displays responses per second and queries per second.
- The x axis displays time. The increments vary, depending upon the date range selected in the **Context Quick Selector** widget.
- Mousing over any point in any line displays the average value at that time-point.
- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected DNS server.

Recursive Queries. This widget displays a line graph. The line graph displays Recursive Queries per Second.

- The y axis displays number of queries per second.
- The x axis displays time. The increments vary, depending upon the date range selected in the **Context Quick Selector** widget.
- Mousing over any point in any line displays the average value at that time-point.
- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected DNS server.

Recursive Errors. This widget displays a line graph. The line graph displays Recursive Query Failures per Second and Recursive Time-Outs per Second. Each parameter is represented by a color-coded line..

- The y axis displays number of queries per second.
- The x axis displays time. The increments vary, depending upon the date range selected in the **Context Quick Selector** widget.
- Mousing over any point in any line displays the average value at that time-point.
- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected DNS server.

Microsoft: Exchange Server 2010 Performance

The Microsoft: Exchange Server 2010 Performance dashboard provides an overview of the health and performance of a selected Exchange 2010 server.

Context Quick Selector. This widget contains buttons for time span presets and the Organizations Selector.

- *Time span presets*. Users select the time span over which they want to view data. Selections range from one hour to 90 days.
- *Organizations Selector*. This drop-down list allows a user to select specific organizations for which they want to view data. This field filters the list of Exchange 2010 servers that appear in the **Server List** widget.

Server List. This widget displays a list of Exchange 2010 servers. Selecting a server drives the context for the other widgets in the dashboard.

Availability and Latency. This widget displays two gauges:

- The availability of the selected Exchange 2010 server, in percent.
- The latency of the selected Exchange 2010 server, in milliseconds.

System Utilization (%). This widget displays a line graph. The line graph displays memory usage, swap memory usage, and CPU usage for the selected Exchange 2010 server during the selected duration. Each parameter is represented by a color-coded line.

- The y axis displays usage, in percent.
- The x axis displays time. The increments vary, depending upon the date range selected in the **Context Quick Selector** widget.
- Mousing over any point in any line displays the average value at that time-point.

- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected Exchange 2010 server.

User Active Connections. This widget displays a line graph. The line graph displays the number of active user connections for the selected Exchange 2010 server during the selected duration.

- The y axis displays the number of users.

- The x axis displays time. The increments vary, depending upon the date range selected in the **Context Quick Selector** widget.

- Mousing over any point in the line displays the average value at that time-point.

- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected Exchange 2010 server.

OWA Requests. This widget displays a line graph. The line graph displays two lines: One for the frequency of Outlook Web Access requests for the selected Exchange 2010 server during the selected duration and another for the frequency of Web Services requests for the selected Exchange 2010 server during the selected duration.

- The y axis displays the number of requests per second.

- The x axis displays time. The increments vary, depending upon the date range selected in the **Context Quick Selector** widget.

- Mousing over any point in any line displays the average value at that time-point.

- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected Exchange 2010 server.

RPC Averaged Latency. This widget displays a line graph. The line graph displays the average latency of remote procedure calls (RPCs) for the selected Exchange 2010 server during the selected duration.

- The y axis displays the average RPC latency, in milliseconds.

- The x axis displays time. The increments vary, depending upon the date range selected in the **Context Quick Selector** widget.

- Mousing over any point in any line displays the average value at that time-point.

- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected Exchange 2010 server.

MBS Databases. This widget displays a line graph. The line graph displays two lines: One for I/O write latency for the mailbox server database for the selected Exchange 2010 and one for I/O read latency to the mailbox server for the selected Exchange 2010 server during the selected duration.

- The y axis displays the write and read latency statistics in milliseconds.

- The x axis displays time. The increments vary, depending upon the date range selected in the **Context Quick Selector** widget.

- Mousing over any point in any line displays the average value at that time-point.

- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected Exchange 2010 server.

Mailbox Messages.This widget displays a line graph. The line graph displays two lines: One for the number of mailbox messages sent to the selected Exchange 2010 server and one for the number of mailbox message sent from the selected Exchange 2010 server during the selected duration.

- The y axis displays the number of messages per second.
- The x axis displays time. The increments vary, depending upon the date range selected in the *Context Quick Selector* widget.
- Mousing over any point in any line displays the average value at that time-point.
- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected Exchange 2010 server.

Total Queue Messages. This widget displays a line graph. The line graph includes three lines: One for the number of messages in the submission queue, one for the number of messages in the delivery queue, and one for the number of queued message that were delivered for the selected Exchange 2010 server during the selected duration.

- The y axis displays the number of messages per second.
- The x axis displays time. The increments vary, depending upon the date range selected in the *Context Quick Selector* widget.
- Mousing over any point in any line displays the average value at that time-point.
- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected Exchange 2010 server.

SMTP Messages. This widget displays a line graph. The line graphs includes two lines: One for the number of SMTP messages sent from the selected Exchange 2010 server and one for the number of SMTP messages received by the selected Exchange 2010 server during the selected duration.

- The y axis displays the number of messages per second.
- The x axis displays time. The increments vary, depending upon the date range selected in the *Context Quick Selector* widget.
- Mousing over any point in any line displays the average value at that time-point.
- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected Exchange 2010 server.

Buckets Allocated. This widget displays a line graph. The line graph displays the number of buckets of version store memory used by the selected Exchange 2010 server during the selected duration.

- The y axis displays the number of allocated buckets.
- The x axis displays time. The increments vary, depending upon the date range selected in the *Context Quick Selector* widget.
- Mousing over any point in any line displays the average value at that time-point.
- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected Exchange 2010 server.

Microsoft: Exchange Server 2013 Performance

The Microsoft: Exchange Server 2013 Performance dashboard provides an overview of the health and performance of a selected Exchange 2013 server.

Context Quick Selector. This widget contains buttons for time span presets and the Organizations Selector.

- *Time span presets*. Users select the time span over which they want to view data. Selections range from one hour to 90 days.
- *Organizations Selector*. This drop-down list allows a user to select specific organizations for which they want to view data. This field filters the list of Exchange 2013 servers that appear in the *Server List* widget.

Server List. This widget displays a list of Exchange 2013 servers. Selecting a server drives the context for the other widgets in the dashboard.

Availability and Latency. This widget displays two gauges:

- The availability of the selected Exchange 2013 server, in percent.
- The latency of the selected Exchange 2013 server, in milliseconds.

System Utilization (%). This widget displays a line graph. The line graph displays three lines: One for memory usage, one for swap memory usage, and one for CPU usage for the selected Exchange 2013 server during the selected duration. Each parameter is represented by a color-coded line.

- The y axis displays usage, in percent.
- The x axis displays time. The increments vary, depending upon the date range selected in the *Context Quick Selector* widget.
- Mousing over any point in any line displays the average value at that time-point.

- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected Exchange 2013 server.

User Active Connections. This widget displays a line graph. The line graph displays the number of active user connections for the selected Exchange 2013 server during the selected duration.

- The y axis displays the number of users.
- The x axis displays time. The increments vary, depending upon the date range selected in the **Context Quick Selector** widget.
- Mousing over any point in the line displays the average value at that time-point.
- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected Exchange 2013 server.

OWA Requests. This widget displays a line graph. The line graph displays two lines: One for the frequency of Outlook Web Access requests and one for the frequency of Web Services requests for the selected Exchange 2013 server during the selected duration.

- The y axis displays the number of requests per second.
- The x axis displays time. The increments vary, depending upon the date range selected in the **Context Quick Selector** widget.
- Mousing over any point in any line displays the average value at that time-point.
- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected Exchange 2013 server.

RPC Averaged Latency. This widget displays a line graph. The line graph displays the average latency for remote procedure calls (RPCs) for the selected Exchange 2013 server during the selected duration.

- The y axis displays the average RPC latency, in milliseconds.
- The x axis displays time. The increments vary, depending upon the date range selected in the **Context Quick Selector** widget.
- Mousing over any point in any line displays the average value at that time-point.
- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected Exchange 2013 server.

MBS Databases. This widget displays a line graph. The line graph displays two lines: One for I/O write latency to the mailbox server database and one for I/O read latency to the mailbox server database for the selected Exchange 2013 server during the selected duration.

- The y axis displays the average write and read latency in milliseconds.
- The x axis displays time. The increments vary, depending upon the date range selected in the **Context Quick Selector** widget.
- Mousing over any point in any line displays the average value at that time-point.
- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected Exchange 2013 server.

Mailbox Messages. This widget displays a line graph. The line graph displays two lines: One for the number of mailbox messages sent from the selected Exchange 2013 and one for the number of mailbox messages delivered to the selected Exchange 2013 server during the selected duration.
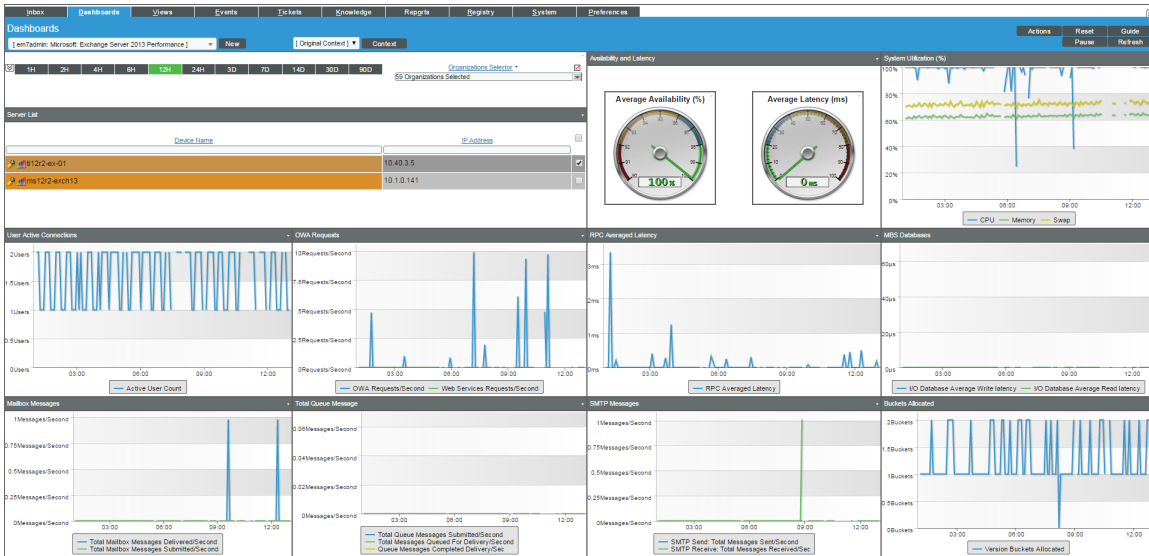
- The y axis displays the number of messages per second.
- The x axis displays time. The increments vary, depending upon the date range selected in the ***Context Quick Selector*** widget.
- Mousing over any point in any line displays the average value at that time-point.
- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected Exchange 2013 server.

Total Queue Messages. This widget displays a line graph. The line graph displays three lines: One for the the number of messages in the submission queue, one for the number of messages in the delivery queue, and one for the number of queued message that were delivered for the selected Exchange 2013 server during the selected duration.

- The y axis displays the number of messages per second.
- The x axis displays time. The increments vary, depending upon the date range selected in the ***Context Quick Selector*** widget.
- Mousing over any point in any line displays the average value at that time-point.
- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected Exchange 2013 server.

SMTP Messages. This widget displays a line graph. The line graph displays two lines: One for the number of SMTP messages sent from the selected Exchange 2013 server and one for the number of SMTP messages received by the selected Exchange 2013 server during the selected duration.

- The y axis displays the number of messages per second.
- The x axis displays time. The increments vary, depending upon the date range selected in the ***Context Quick Selector*** widget.
- Mousing over any point in any line displays the average value at that time-point.
- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected Exchange 2013 server.

Buckets Allocated. This widget displays a line graph. The line graph displays the number of buckets of version store memory used by the selected Exchange 2013 server during the selected duration.

- The y axis displays the number of allocated buckets.
- The x axis displays time. The increments vary, depending upon the date range selected in the ***Context Quick Selector*** widget.
- Mousing over any point in any line displays the average value at that time-point.
- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected Exchange 2013 server.

Microsoft: IIS Server Performance

The Microsoft: IIS Server Performance dashboard provides an overview of the health and performance of a selected IIS server.
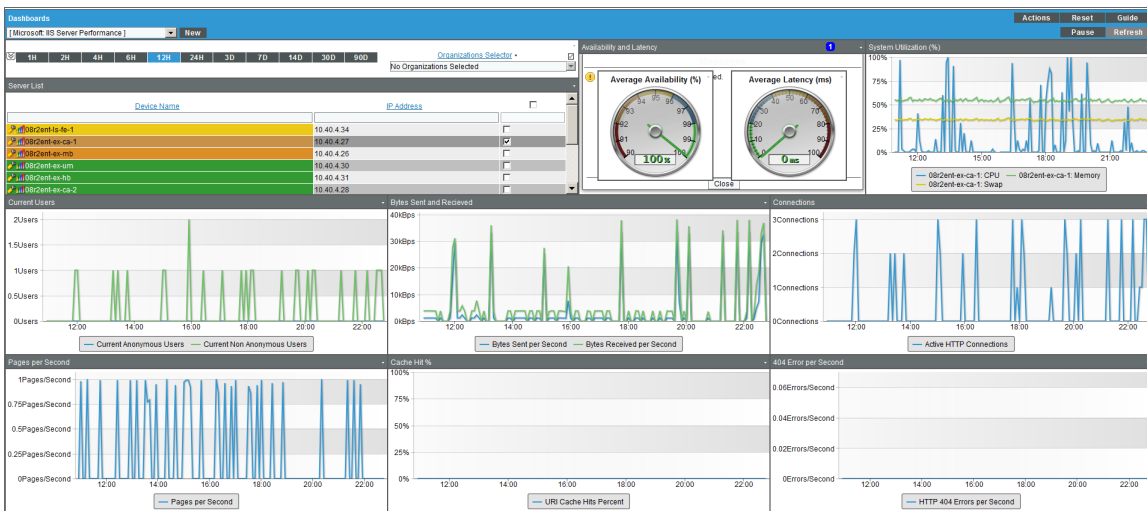
Context Quick Selector. This widget contains buttons for time span presets and the Organizations Selector.

- *Time span presets*. Users select the time span over which they want to view data. Selections range from one hour to 90 days.
- *Organizations Selector*. This drop-down list allows a user to select specific organizations for which they want to view data. This field filters the list of IIS servers that appear in the **Server List** widget.

Server List. This widget displays a list of IIS servers. Selecting a server drives the context for the other widgets in the dashboard.

Availability and Latency. This widget displays two gauges:

- The availability of the selected IIS server, in percent.
- The latency of the selected IIS server, in milliseconds.

System Utilization (%). This widget displays a line graph. The line graph displays memory usage, virtual-memory usage, and CPU usage for the selected IIS server during the selected duration. Each parameter is represented by a color-coded line.

- The y axis displays usage, in percent.
- The x axis displays time. The increments vary, depending upon the date range selected in the **Context Quick Selector** widget.
- Mousing over any point in any line displays the average value at that time-point.

- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected IIS server.

Current Users. This widget displays a line graph. The line graph displays Current Anonymous Users and Current Non Anonymous Users. Each parameter is represented by a color-coded line.

- The y axis displays number of users.
- The x axis displays time. The increments vary, depending upon the date range selected in the *Context Quick Selector* widget.
- Mousing over any point in any line displays the average value at that time-point.
- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected IIS server.

Bytes Sent and Received. This widget displays a line graph. The line graph displays Bytes Sent Per Second and Bytes Received Per Second. Each parameter is represented by a color-coded line.

- The y axis displays kB of data per second..
- The x axis displays time. The increments vary, depending upon the date range selected in the *Context Quick Selector* widget.
- Mousing over any point in any line displays the average value at that time-point.
- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected IIS server.

Connections. This widget displays a line graph. The line graph displays the number of Active HTTP Connections.

- The y axis displays number of connections.
- The x axis displays time. The increments vary, depending upon the date range selected in the *Context Quick Selector* widget.
- Mousing over any point in any line displays the average value at that time-point.
- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected IIS server.

*Pages Per Second*. This widget displays a line graph. The line graph displays the number of Pages (served) Per Second.

- The y axis displays number of pages per second..
- The x axis displays time. The increments vary, depending upon the date range selected in the *Context Quick Selector* widget.
- Mousing over any point in any line displays the average value at that time-point.
- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected IIS server.

Cache Hit %. The IIS server caches (in memory) frequently requested files. This widget displays a line graph. The line graph displays the ratio of kernel URI cache hits to total cache requests.

- The y axis displays percent of URI cache hits.

- The x axis displays time. The increments vary, depending upon the date range selected in the **Context Quick Selector** widget.

- Mousing over any point in any line displays the average value at that time-point.

- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected IIS server.

404 Errors Per Second. This widget displays a line graph. The line graph displays the number of errors due to requests that couldn't be satisfied by the server because the requested document couldn't be found, per second.

- The y axis displays number of errors per second.

- The x axis displays time. The increments vary, depending upon the date range selected in the **Context Quick Selector** widget.

- Mousing over any point in any line displays the average value at that time-point.

- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected IIS server.

Microsoft: Lync Server 2010 Dashboards

The *Microsoft: Lync Server 2010 Dashboards* PowerPack includes the following dashboards:

- Microsoft: Lync Server 2010 Performance
- Microsoft: Lync Server 2010 Utilization

Microsoft: Lync Server 2010 Performance

The Microsoft: Lync 2010 Server Performance dashboard provides an overview of the health and performance of a selected Lync 2010 server.

Context Quick Selector. This widget contains buttons for time span presets and the Organizations Selector.

- *Time span presets*. Users select the time span over which they want to view data. Selections range from one hour to 90 days.
- *Organizations Selector*. This drop-down list allows a user to select specific organizations for which they want to view data. This field filters the list of Lync 2010 servers that appear in the **Server List** widget.

Server List. This widget displays a list of Lync 2010 servers. Selecting a server drives the context for the other widgets in the dashboard.

Availability and Latency. This widget displays two gauges:

- The availability of the selected Lync 2010 server, in percent.
- The latency of the selected Lync 2010 server, in milliseconds.

System Utilization (%). This widget displays a line graph. The line graph displays memory usage, virtual-memory usage, and CPU usage for the selected Lync 2010 server during the selected duration. Each parameter is represented by a color-coded line.

- The y axis displays usage, in percent.
- The x axis displays time. The increments vary, depending upon the date range selected in the **Context Quick Selector** widget.
- Mousing over any point in any line displays the average value at that time-point.
- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected Lync 2010 server.

Connections Established. This widget displays a line graph. The line graph displays Connections Established.

- The y axis displays number of connections.
- The x axis displays time. The increments vary, depending upon the date range selected in the **Context Quick Selector** widget.
- Mousing over any point in any line displays the average value at that time-point.
- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected Lync 2010 server.

SIP Message. SIP is a protocol for instant messaging and VOIP. This widget displays a line graph. The line graph displays Incoming Message and Outgoing Messages. Each parameter is represented by a color-coded line.

- The y axis displays number of messages.
- The x axis displays time. The increments vary, depending upon the date range selected in the **Context Quick Selector** widget.
- Mousing over any point in any line displays the average value at that time-point.
- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected Lync 2010 server.

Sproc Latency. Stored Procedure Call (sproc) latency is the time it takes for the Lync database to process the stored procedure call.

- The y axis displays the duration, in milliseconds.
- The x axis displays time. The increments vary, depending upon the date range selected in the ***Context Quick Selector*** widget.
- Mousing over any point in any line displays the average value at that time-point.
- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected Lync 2010 server.
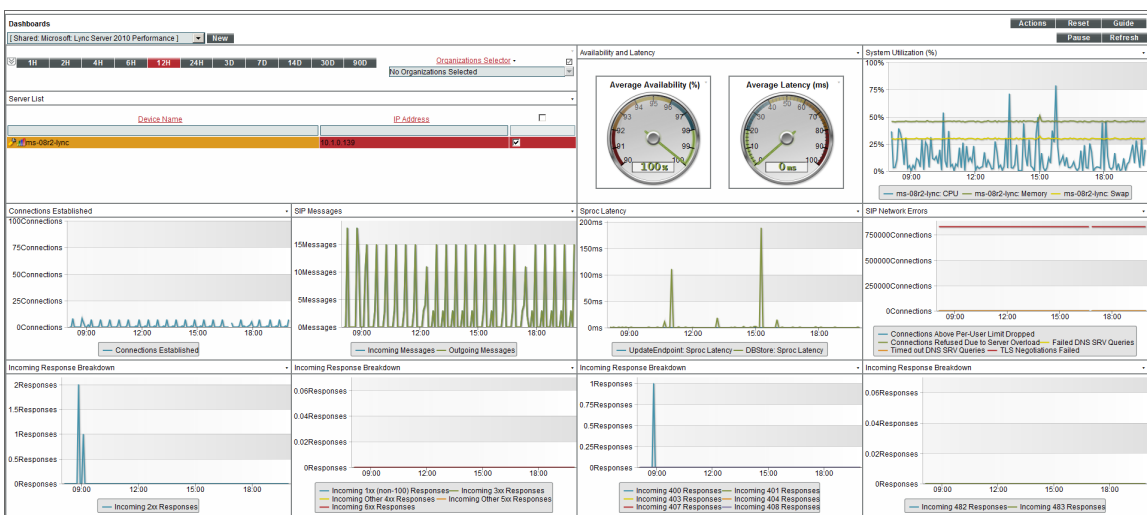
SIP Network Errors. This widget displays information about errors during instant messaging or VOIP. This widget displays a line graph. The line graph displays Connections Above Per-User Limit Dropped, Connections Refused Due to Server Overload, Failed DNS SRV Queries, Time Out DNS SRV Queries, and TLS Negotiations Failed. Each parameter is represented by a color-coded line.

- The y axis displays the number of connections that resulted in errors.
- The x axis displays time. The increments vary, depending upon the date range selected in the ***Context Quick Selector*** widget.
- Mousing over any point in any line displays the average value at that time-point.
- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected Lync 2010 server.

Incoming Response Breakdown. This widget displays information about the number of responses generated by the server. This widget displays a line graph. The line graph displays Incoming 2xx Responses. A 2xx Response means that a connection has been established.

- The y axis displays the number of responses.
- The x axis displays time. The increments vary, depending upon the date range selected in the ***Context Quick Selector*** widget.
- Mousing over any point in any line displays the average value at that time-point.
- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected Lync 2010 server.

Incoming Response Breakdown. This widget displays information about the number of responses generated by the server. This widget displays a line graph. The line graph displays Incoming 1xx (non-100) Responses, Incoming 3xx Responses, Incoming Other 4xx Responses, Incoming Other 5xx Responses, and Incoming 6xx Responses. Each parameter is represented by a color-coded line. For a description of SIP response codes, see the Wikipedia page http://en.wikipedia.org/wiki/List_of_SIP_response_codes.

- The y axis displays the number of responses.
- The x axis displays time. The increments vary, depending upon the date range selected in the ***Context Quick Selector*** widget.
- Mousing over any point in any line displays the average value at that time-point.
- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected Lync 2010 server.

Incoming Response Breakdown. This widget displays information about the number of responses generated by the server. This widget displays a line graph. The line graph displays Incoming 400 Responses, Incoming 401 Responses, Incoming Other 403 Responses, Incoming 404 Responses, Incoming 407 Responses, and Incoming 408 Responses. Each parameter is represented by a color-coded line. For a description of SIP response codes, see the Wikipedia page http://en.wikipedia.org/wiki/List_of_SIP_response_codes.

- The y axis displays the number of responses.
- The x axis displays time. The increments vary, depending upon the date range selected in the ***Context Quick Selector*** widget.
- Mousing over any point in any line displays the average value at that time-point.
- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected Lync 2010 server.

Incoming Response Breakdown. This widget displays information about the number of responses generated by the server. This widget displays a line graph. The line graph displays Incoming 482 Responses and Incoming 483 Responses. Each parameter is represented by a color-coded line. For a description of SIP response codes, see the Wikipedia page http://en.wikipedia.org/wiki/List_of_SIP_response_codes.

- The y axis displays the number of responses.
- The x axis displays time. The increments vary, depending upon the date range selected in the ***Context Quick Selector*** widget.
- Mousing over any point in any line displays the average value at that time-point.
- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected Lync 2010 server.

Microsoft: Lync Server 2010 Utilization

The Microsoft: Lync 2010 Server Utilization dashboard provides an overview of how users are using a selected Lync 2010 server.

Context Quick Selector. This widget contains the time span preset buttons and Organizations Selector.

- *Time span presets*. Users select the time span over which they want to view data. Selections range from one hour to 90 days.
- *Organizations Selector*. This drop-down list allows a user to select specific organizations for which they want to view data. This field filters the list of Lync 2010 servers that appear in the **Server List** widget.

Server List. This widget displays a list of Lync 2010 servers. Selecting a server drives the context for the other widgets in the dashboard.

Availability and Latency. This widget displays two gauges:

- The availability of the selected Lync 2010 server, in percent.
- The latency of the selected Lync 2010 server, in milliseconds.

System Utilization (%). This widget displays a line graph. The line graph displays memory usage, virtual-memory usage, and CPU usage for the selected Lync 2010 server during the selected duration. Each parameter is represented by a color-coded line.
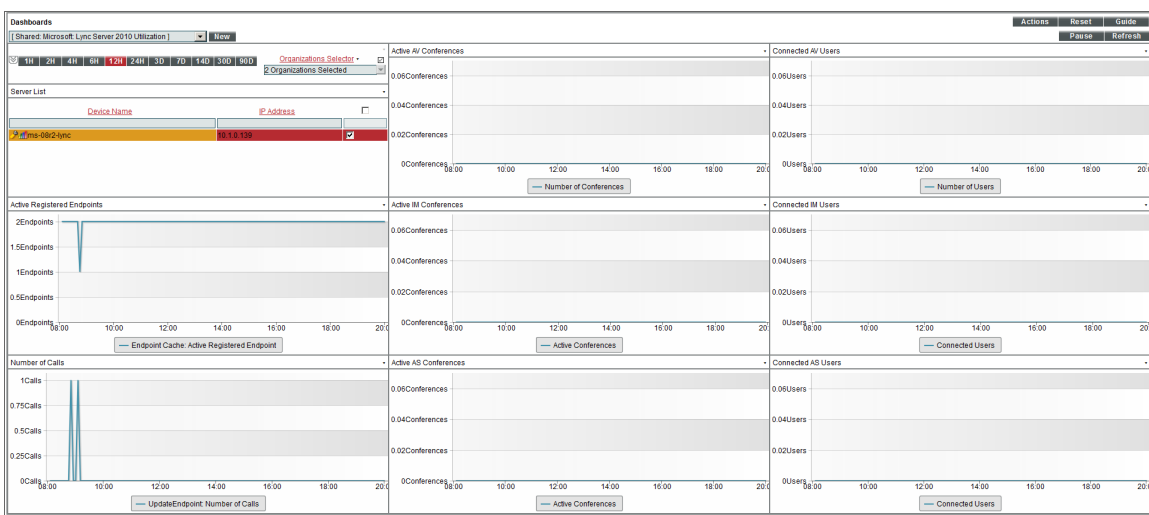
- The y axis displays usage, in percent t.
- The x axis displays time. The increments vary, depending upon the date range selected in the **Context Quick Selector** widget.
- Mousing over any point in any line displays the average value at that time-point.
- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected Lync 2010 server.

Active Registered Endpoints. Endpoints are devices that are connected to the Lync front-end server. This widget displays a line graph. The line graph displays Endpoint Cache: Active Registered Endpoints.

- The y axis displays numbered of registered endpoints.
- The x axis displays time. The increments vary, depending upon the date range selected in the **Context Quick Selector** widget.
- Mousing over any point in any line displays the average value at that time-point.
- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected Lync 2010 server.

Active IM Conferences. This widget displays the current number of IM conversations on the Lync server. Conferences usually include more than two users. This widget displays a line graph. The line graph displays Active Conferences.

- The y axis displays numbered of IM conferences.
- The x axis displays time. The increments vary, depending upon the date range selected in the ***Context Quick Selector*** widget.
- Mousing over any point in any line displays the average value at that time-point.
- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected Lync 2010 server.

Connected IM Users. This widget displays the current number of connected IM users. This widget displays a line graph. The line graph displays Connected Users.

- The y axis displays numbered of IM users.
- The x axis displays time. The increments vary, depending upon the date range selected in the ***Context Quick Selector*** widget.
- Mousing over any point in any line displays the average value at that time-point.
- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected Lync 2010 server.

Number of Calls. This widget displays the current number of voice calls on the Lync server. This widget displays a line graph. The line graph displays UpdateEndpoint: Number of Calls.

- The y axis displays numbered of calls.
- The x axis displays time. The increments vary, depending upon the date range selected in the ***Context Quick Selector*** widget.
- Mousing over any point in any line displays the average value at that time-point.
- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected Lync 2010 server.

Active AS Conferences. This widget displays the number of active conferences using Application Sharing (AS). This widget displays a line graph. The line graph displays Active Conferences.

- The y axis displays numbered of AS conferences.
- The x axis displays time. The increments vary, depending upon the date range selected in the ***Context Quick Selector*** widget.
- Mousing over any point in any line displays the average value at that time-point.
- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected Lync 2010 server.

Connected AS Users. This widget displays the number of users connected to conferences using Application Sharing (AS). This widget displays a line graph. The line graph displays Connected Users.

- The y axis displays numbered of AS users.
- The x axis displays time. The increments vary, depending upon the date range selected in the **Context Quick Selector** widget.
- Mousing over any point in any line displays the average value at that time-point.
- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected Lync 2010 server.

Microsoft: Skype for Business Dashboards

The *Microsoft: Skype for Business Dashboards* PowerPack includes the following dashboards:

- Microsoft: Lync Server 2013 Performance
- Microsoft: Lync Server 2013 Utilization

Microsoft: Lync Server 2013 Performance

The Microsoft: Lync 2013 Server Performance dashboard provides an overview of the health and performance of a selected Lync 2013 server.

.

Context Quick Selector. This widget contains the time span preset buttons and Organizations Selector.

- *Time span presets*. Users select the time span over which they want to view data. Selections range from one hour to 90 days.
- *Organizations Selector*. This drop-down list allows a user to select specific organizations for which they want to view data. This field filters the list of Lync 2013 servers that appear in the **Server List** widget.

Server List. This widget displays a list of Lync 2013 servers. Selecting a server drives the context for the other widgets in the dashboard.

Availability and Latency. This widget displays two gauges:

- The availability of the selected Lync 2013 server, in percent.
- The latency of the selected Lync 2013 server, in milliseconds.

System Utilization (%). This widget displays a line graph. The line graph displays memory usage, virtual-memory usage, and CPU usage for the selected Lync 2013 server during the selected duration. Each parameter is represented by a color-coded line.

- The y axis displays usage, in percent.
- The x axis displays time. The increments vary, depending upon the date range selected in the **Context Quick Selector** widget.
- Mousing over any point in any line displays the average value at that time-point.
- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected Lync 2013 server.

Connections Established. This widget displays a line graph. The line graph displays Connections Established.

- The y axis displays number of connections.
- The x axis displays time. The increments vary, depending upon the date range selected in the **Context Quick Selector** widget.
- Mousing over any point in any line displays the average value at that time-point.
- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected Lync 2010 server.

SIP Message. SIP is a protocol for instant messaging and VOIP. This widget displays a line graph. The line graph displays Incoming Message and Outgoing Messages. Each parameter is represented by a color-coded line.

- The y axis displays number of messages.
- The x axis displays time. The increments vary, depending upon the date range selected in the **Context Quick Selector** widget.
- Mousing over any point in any line displays the average value at that time-point.
- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected Lync 2013 server.

Sproc Latency. Stored Procedure Call (sproc) latency is the time it takes for the Lync database to process the stored procedure call.

- The y axis displays the duration, in milliseconds.
- The x axis displays time. The increments vary, depending upon the date range selected in the **Context Quick Selector** widget.
- Mousing over any point in any line displays the average value at that time-point.

- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected Lync 2013 server.

SIP Network Errors. This widget displays information about errors during instant messaging or VOIP. This widget displays a line graph. The line graph displays Connections Above Per-User Limit Dropped, Connections Refused Due to Server Overload, Failed DNS SRV Queries, Time Out DNS SRV Queries, and TLS Negotiations Failed. Each parameter is represented by a color-coded line.

- The y axis displays the number of connections that resulted in errors.

- The x axis displays time. The increments vary, depending upon the date range selected in the **Context Quick Selector** widget.

- Mousing over any point in any line displays the average value at that time-point.

- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected Lync 2013 server.

Incoming Response Breakdown. This widget displays information about the number of responses that are being generated by the server. This widget displays a line graph. The line graph displays Incoming 2xx Responses. A 2xx Response means that a connection has been established.

- The y axis displays the number of responses.

- The x axis displays time. The increments vary, depending upon the date range selected in the **Context Quick Selector** widget.

- Mousing over any point in any line displays the average value at that time-point.

- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected Lync 2013 server.

Incoming Respond Breakdown. This widget displays information about the number of responses that are being generated by the server. This widget displays a line graph. The line graph displays Incoming 1xx (non-100) Responses, Incoming 3xx Responses, Incoming Other 4xx Responses, Incoming Other 5xx Responses, and Incoming 6xx Responses. Each parameter is represented by a color-coded line. For a description of all SIP response codes, see the Wikipedia page http://en.wikipedia.org/wiki/List_of_SIP_response_codes.

- The y axis displays the number of responses.

- The x axis displays time. The increments vary, depending upon the date range selected in the **Context Quick Selector** widget.

- Mousing over any point in any line displays the average value at that time-point.

- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected Lync 2013 server.

Incoming Response Breakdown. This widget displays information about the number of responses that are being generated by the server. This widget displays a line graph. The line graph displays Incoming 400 Responses, Incoming 401 Responses, Incoming Other 403 Responses, Incoming 404 Responses, Incoming 407 Responses, and Incoming 408Responses. Each parameter is represented by a color-coded line. For a description of all SIP response codes, see the Wikipedia page http://en.wikipedia.org/wiki/List_of_SIP_response_codes.
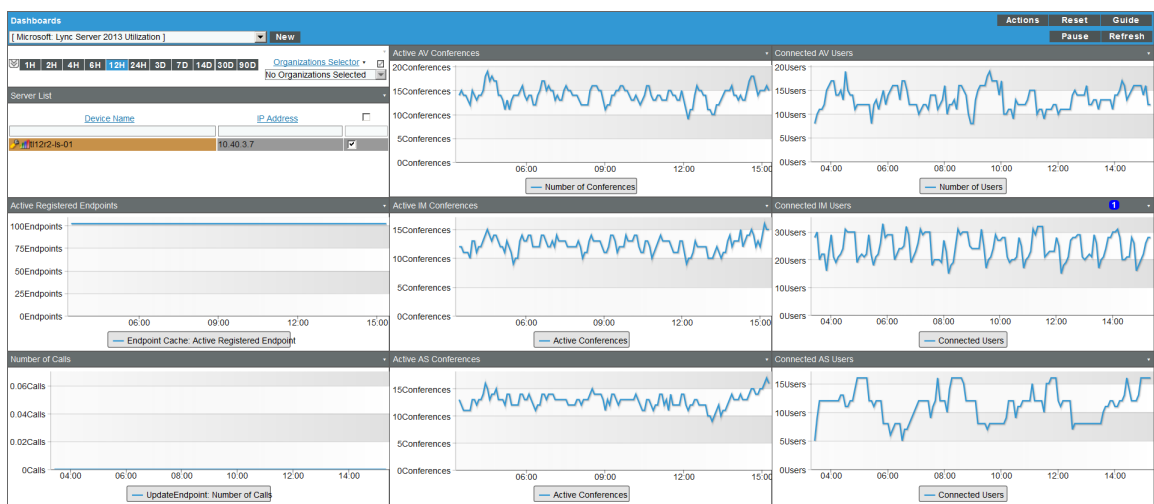
- The y axis displays the number of responses.
- The x axis displays time. The increments vary, depending upon the date range selected in the ***Context Quick Selector*** widget.
- Mousing over any point in any line displays the average value at that time-point.
- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected Lync 2013 server.

Incoming Response Breakdown. This widget displays information about the number of responses that are being generated by the server. This widget displays a line graph. The line graph displays Incoming 482 Responses and Incoming 483 Responses. Each parameter is represented by a color-coded line. For a description of all SIP responses codes, see the Wikipedia page http://en.wikipedia.org/wiki/List_of_SIP_response_codes.

- The y axis displays the number of responses.
- The x axis displays time. The increments vary, depending upon the date range selected in the ***Context Quick Selector*** widget.
- Mousing over any point in any line displays the average value at that time-point.
- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected Lync 2013 server.

Microsoft: Lync Server 2013 Utilization

The Microsoft: Lync 2013 Server Utilization dashboard provides an overview of how users are using a selected Lync 2013 server.

Context Quick Selector. This widget contains buttons for time span presets and the Organizations Selector.

- *Time span presets*. Users select the time span over which they want to view data. Selections range from one hour to 90 days.
- *Organizations Selector*. This drop-down list allows a user to select specific organizations for which they want to view data. This field filters the list of Lync 2013 servers that appear in the **Server List** widget.

Server List. This widget displays a list of Lync 2013 servers. Selecting a server drives the context for the other widgets in the dashboard.

Availability and Latency. This widget displays two gauges:

- The availability of the selected Lync 2013 server, in percent.
- The latency of the selected Lync 2013 server, in milliseconds.

System Utilization (%). This widget displays a line graph. The line graph displays memory usage, virtual-memory usage, and CPU usage for the selected Lync 2013 server during the selected duration. Each parameter is represented by a color-coded line.

- The y axis displays usage, in percent.
- The x axis displays time. The increments vary, depending upon the date range selected in the **Context Quick Selector** widget.
- Mousing over any point in any line displays the average value at that time-point.
- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected Lync 2013 server.

Active Registered Endpoints. Endpoints are devices that are connected to the Lync front-end server. This widget displays a line graph. The line graph displays Endpoint Cache: Active Registered Endpoints.

- The y axis displays the number of registered endpoints.
- The x axis displays time. The increments vary, depending upon the date range selected in the **Context Quick Selector** widget.
- Mousing over any point in any line displays the average value at that time-point.
- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected Lync 2013 server.

Active IM Conferences. This widget displays the number of IM conversations on the Lync server. Conferences usually include more than two users. This widget displays a line graph. The line graph displays Active Conferences.

- The y axis displays the number of IM conferences.
- The x axis displays time. The increments vary, depending upon the date range selected in the ***Context Quick Selector*** widget.
- Mousing over any point in any line displays the average value at that time-point.
- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected Lync 2013 server.

Connected IM Users. This widget displays the current number of connected IM users. This widget displays a line graph. The line graph displays Connected Users.

- The y axis displays the number of IM users.
- The x axis displays time. The increments vary, depending upon the date range selected in the ***Context Quick Selector*** widget.
- Mousing over any point in any line displays the average value at that time-point.
- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected Lync 2013 server.

Number of Calls. This widget displays the current number of voice calls on the Lync server. This widget displays a line graph. The line graph displays UpdateEndpoint: Number of Calls.

- The y axis displays the number of calls.
- The x axis displays time. The increments vary, depending upon the date range selected in the ***Context Quick Selector*** widget.
- Mousing over any point in any line displays the average value at that time-point.
- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected Lync 2013 server.

Active AS Conferences. This widget displays the number of active conferences using Application Sharing (AS). This widget displays a line graph. The line graph displays Active Conferences.

- The y axis displays the number of AS conferences.
- The x axis displays time. The increments vary, depending upon the date range selected in the ***Context Quick Selector*** widget.
- Mousing over any point in any line displays the average value at that time-point.
- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected Lync 2013 server.

Connected AS Users. This widget displays the number of users connected to conferences using Application Sharing (AS). This widget displays a line graph. The line graph displays Connected Users.

- The y axis displays the number of AS users.

- The x axis displays time. The increments vary, depending upon the date range selected in the *Context Quick Selector* widget.

- Mousing over any point in any line displays the average value at that time-point.

- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected Lync 2013 server.

Microsoft: SQL Server Performance

The Microsoft: SQL Server Performance dashboard provides an overview of the health and performance of a selected SQL server.
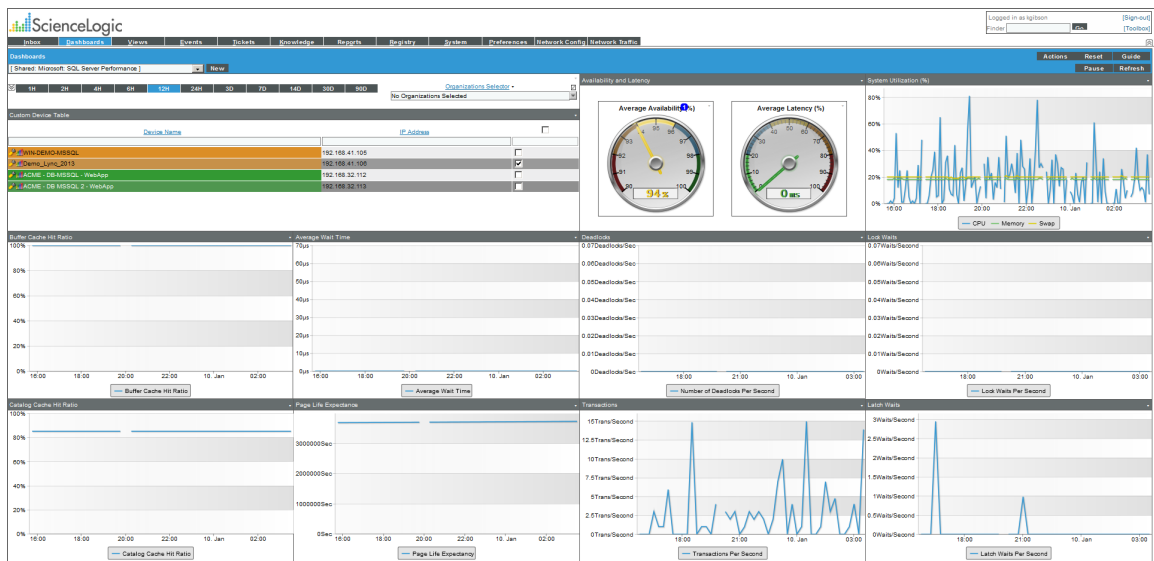
Context Quick Selector. This widget contains buttons for the time span presets and the Organizations Selector.

- *Time span presets*. Users select the time span over which they want to view data. Selections range from one hour to 90 days.

- *Organizations Selector*. This drop-down list allows a user to select specific organizations for which they want to view data. This field filters the list of SQL servers that appear in the *Server List* widget.

Server List. This widget displays a list of SQL servers. Selecting a server drives the context for the other widgets in the dashboard.

Availability and Latency. This widget displays two gauges:

- The availability of the selected SQL server, in percent.
- The latency of the selected SQL server, in milliseconds.

System Utilization (%). This widget displays a line graph. The line graph displays memory usage, virtual-memory usage, and CPU usage for the selected SQL server during the selected duration. Each parameter is represented by a color-coded line.

- The y axis displays usage, in percent.
- The x axis displays time. The increments vary, depending upon the date range selected in the **Context Quick Selector** widget.
- Mousing over any point in any line displays the average value at that time-point.
- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected SQL server.

Buffer Cache Hit Ratio. This widget displays information about the percentage of page requests that are satisfied by data pages from the buffer cache without having to read from disk. The ratio is the total number of pages found in the buffer divided by the total number of requests. This widget displays a line graph. The line graph displays Buffer Cache Hit Ratio.

- The y axis displays the ratio, in percent.
- The x axis displays time. The increments vary, depending upon the date range selected in the **Context Quick Selector** widget.
- Mousing over any point in any line displays the average value at that time-point.
- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected SQL server.

Average Wait Time. This widget displays information about the average wait time to acquire a lock. This widget displays a line graph. The line graph displays Average Wait Time.

- The y axis displays the wait time, in milliseconds.
- The x axis displays time. The increments vary, depending upon the date range selected in the **Context Quick Selector** widget.
- Mousing over any point in any line displays the average value at that time-point.
- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected SQL server.

Deadlocks. This widget displays information about deadlocks. A deadlock occurs when two or more tasks permanently block each other because each task tries to lock a resource which the other tasks are also trying to lock. This widget displays a line graph. The line graph displays Number of Deadlocks Per Second.

- The y axis displays the number of deadlocks per second.
- The x axis displays time. The increments vary, depending upon the date range selected in the *Context Quick Selector* widget.
- Mousing over any point in any line displays the average value at that time-point.
- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected SQL server.

Lock Waits. This widget displays information about the number of lock requests per second that require the requester to wait. This widget displays a line graph. The line graph displays Lock Waits Per Second.

- The y axis displays the number of waits per second.
- The x axis displays time. The increments vary, depending upon the date range selected in the *Context Quick Selector* widget.
- Mousing over any point in any line displays the average value at that time-point.
- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected SQL server.

Catalog Cache Hit Ratio. This widget displays information about the ratio between catalog metadata cache hits and lookups. The ratio is the total number of pages found in the catalog metadata cache divided by the total number of lookups. This widget displays a line graph. The line graph displays Catalog Cache Hit Ratio.

- The y axis displays the ratio.
- The x axis displays time. The increments vary, depending upon the date range selected in the *Context Quick Selector* widget.
- Mousing over any point in any line displays the average value at that time-point.
- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected SQL server.

Page Life Expectancy. This widget displays information about the number of seconds a page will stay in the buffer pool (memory cache) without references. This widget displays a line graph. The line graph displays Page Life Expectancy.

- The y axis displays the number of seconds a page will stay in the buffer pool.
- The x axis displays time. The increments vary, depending upon the date range selected in the *Context Quick Selector* widget.
- Mousing over any point in any line displays the average value at that time-point.
- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected SQL server.

Transactions. A transaction is a sequence of operations that make up a single logical unit of work, usually a change to the database. This widget displays information about the number of transactions per second to the SQL server. This widget displays a line graph. The line graph displays Transactions Per Second.

- The y axis displays the number of transactions per second.
- The x axis displays time. The increments vary, depending upon the date range selected in the ***Context Quick Selector*** widget.
- Mousing over any point in any line displays the average value at that time-point.
- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected SQL server.

Latch Waits. A latch is an object that ensures data integrity for objects in the buffer pool (memory cache). This widget displays a line graph. The line graph displays Latch Waits Per Second.

- The y axis displays the number of waits per second.
- The x axis displays time. The increments vary, depending upon the date range selected in the ***Context Quick Selector*** widget.
- Mousing over any point in any line displays the average value at that time-point.
- Clicking on a data point displays the Device Performance graph for the selected parameter on the selected SQL server.

Chapter

9

Troubleshooting

Overview

Use the following menu options to navigate the SL1 user interface:

- To view a pop-out list of menu options, click the menu icon (▤).

- To view a page containing all the menu options, click the Advanced menu icon (⋯).

The following sections describe some of the error messages that you might see when configuring SL1 to monitor Windows devices:

Troubleshooting WinRM Error Messages

SL1 can generate the following error messages when problems occur in Windows Remote Management (WinRM). For each error message, the top-most cause listed is the most likely reason for the error message.

Error / Message	Cause / Resolution
Incorrect username and/or password provided in the PowerShell Credential.	Bad HTTP response returned from server. Basic authentication failed. Code 401. (For more information, see the section *Debugging Code 401 Errors*.)
	Pre-authentication failed while getting initial credentials.
	Client not found in Kerberos database.
The device cannot respond to WinRM requests or the PowerShell credential settings do not match the device's WinRM configuration.	Kerberos-based authentication failed. Code 500. (For more information, see the section *Debugging Code 500 Errors*.)
	[Errno 111] Connection refused.
	ParseError.
Server is offline.	Increase the **Timeout** value on your ScienceLogic credential.

> **NOTE:** If you receive an error message that is a combination of the first two error messages, then you must run debugging steps for both Code 401 and Code 500.

Debugging Code 401 Errors

If you encounter a Code 401 error, perform the following troubleshooting steps to debug the error:

- Determine if the error is caused by an issue with the Kerberos ticket:

    - Ensure forward and reverse DNS are configured correctly when using Active Directory authentication:

    ```
    # nslookup [IP address]
    # nslookup [hostname]
    ```

    - Ensure you are able to run the following command without error from the collector:

    ```
    # kinit [username@DOMAINNAME]
    ```

    - If you see the following error, change the domain name to all capital letters:

- Ensure that your WinRM settings match your ScienceLogic credential.

    - To print out current WinRM settings:

Troubleshooting WinRM Error Messages

```
# winrm get winrm/config
```

- If your ScienceLogic credential says no encryption, AllowUnencrypted should be set to True for both the Client and the Service:

```
# winrm set winrm/config/client '@{AllowUnencrypted="$true"}'
# winrm set winrm/config/service '@{AllowUnencrypted="$true"}'
```

- If you are using local type credentials, basic Authentication should be set to True for both Client and Service:

```
# winrm set winrm/config/client/Auth '@{Basic="$true"}'
# winrm set winrm/config/service/Auth '@{Basic="$true"}'
```

- If you are using AD type credentials, Kerberos Authentication should be set to True for both Client and Service:

```
# winrm set winrm/config/client/Auth '@{Kerberos="$true"}'
# winrm set winrm/config/service/Auth '@{Kerberos="$true"}'
```

- In the ScienceLogic credential, ensure the Active Directory **Hostname/IP** field contains the FQDN and the **LDAP Domain** field includes the domain.

- In the ScienceLogic credential, the value in the **LDAP Domain** field might need to be entered in all capital letters.

- Ensure your ScienceLogic credentials are correct:

  - SSH to your Data Collector and try running the following command:

```
# wmic -U 'user%password' //IP "select * from Win32_ComputerSystem"
```

NOTE: If you choose to copy and paste the above command from this document into a shell session, you might have to replace the single and double quotation marks.

- If you are using Windows Servers 2012 and above, make sure that the user you are using belongs to the group: WinRMRemoteWMIUsers__

- If you are using Windows Server 2008, 2008r2, or below, ensure that the user you are using is an administrator. This is a Windows requirement.

- If multiple domains are in use, ensure that they are mapped in the [domain_realm] section of the Kerberos krb5.conf file.

  - The [domain_realm] section provides a translation from a domain name or hostname to a Kerberos realm name.

- Ensure that the username and password are correct and that you can log on to the system.

- Ensure your credential cache is up-to-date:

  - SSH to your Data Collector and cd to the /tmp/ directory.
  - Do an 'ls' to list all the contents of the /tmp/ directory.

- If you see any files that being with "krb5cc_", delete those files.

Debugging Code 500 Errors

If you encounter a Code 500 error, perform the following troubleshooting steps to debug the error:

- In the ScienceLogic credential, increase the value in the **Timeout** field (e.g., 180000 ms.).

- Increase the timeout in the WinRM settings:

  ```
  winrm set winrm/config '@{MaxTimeoutms="30000"}'
  ```

- Increase the maximum number of concurrent operations per user:

  ```
  winrm set winrm/config/service '@{MaxConcurrentOperationsPerUser="100"}'
  ```

- Increase the maximum number of connections:

  ```
  winrm set winrm/config/service '@{MaxConnections="100"}'
  ```

- Increase the maximum number of concurrent operations:

  ```
  winrm set winrm/config/service '@{MaxConcurrentOperations="500"}'
  ```

- Ensure that the Windows device being monitored is not exceeding its resource thresholds. You can do this by opening Resource Monitor on the Windows Device and monitoring the CPU usage.

Troubleshooting PowerShell Error Messages

SL1 can generate the following error message when monitoring Windows devices using PowerShell. This error message usually indicates that an issue with WinRM is not causing the error.

Error / Message	Cause / Resolution
Get-Counter The specified object was not found on the computer.	The PowerShell object was not found on the device that is being monitored. To test this, copy the PowerShell request from the Dynamic Application and run it on the Windows device in a PowerShell shell as Administrator. If you get a similar error message, then the counter does not exist on your Windows device. This means that the user must install the necessary service on the Windows device.