



Monitoring Linux Systems with SSH

Linux Base Pack version 111, Revision 1

Table of Contents

Introduction	3
What is SSH?	4
What Does the Linux Base Pack PowerPack Monitor?	4
Installing the Linux Base Pack PowerPack	5
Upgrading the PowerPack and Removing Dynamic Applications	5
Monitoring Linux with SSH	8
Prerequisites for Monitoring Linux Devices with SSH	9
Linux Distributions Supported by the Linux Base Pack PowerPack	9
Configuring Linux Devices	10
Creating an SSH/Key Credential	11
Creating an SSH/Key Credential in the Classic SL1 User Interface	12
Creating a PowerShell Credential in the Classic SL1 User Interface	13
Configuring the Linux Device Template	15
Configuring the Linux: IC Port Performance Dynamic Application	17
Discovering Linux Devices	18
Discovering Linux Devices in the SL1 Classic User Interface	20
Configuring Dynamic Applications for Monitoring	21
Process Monitoring with the Linux Base Pack	21
Configuring Collection Frequency for Linux IC Dynamic Applications	21
Unhiding Linux File Systems	22
Configuring Linux File System Thresholds	23
Aligning the Linux: SSH Cache Worker Dynamic Application	24
Relationships Between Component Devices	25
Configuring Syslog for Linux	26
What is Syslog?	26
Configuring Syslog for Linux	27
Collection Objects	29
Collection Objects in Linux Dynamic Applications	30

Chapter

1

Introduction

Overview

This manual describes how to configure and monitor Linux systems with SL1 using the Dynamic Applications in the *Linux Base Pack PowerPack*.

NOTE: The *Linux Base Pack PowerPack* version 109 release has been removed from the ScienceLogic Support portal and replaced with the version 110 release.

SL1 supports three protocols to monitor Linux devices:



- SNMP
- SSH
- Syslogs

SNMP and Linux are used to proactively poll the device periodically to collect information, while Syslog asynchronously receives logs from the device. Syslog can be used with SNMP or SSH, but you cannot use both SNMP and SSH together.

ScienceLogic recommends using SSH along with Syslog as that provides the most comprehensive and secure monitoring.

The following sections provide an overview of Secure Shell (SSH) and the *Linux Base Pack PowerPack*.

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 of the menu options, click the Advanced menu icon (.

This chapter covers the following topics:

[What is SSH?](#) 4

What Does the Linux Base Pack PowerPack Monitor?	4
Installing the Linux Base Pack PowerPack	5

NOTE: ScienceLogic provides this documentation for the convenience of ScienceLogic customers. Some of the configuration information contained herein pertains to third-party vendor software that 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.

What is SSH?

Secure Shell (SSH) is a network protocol that enables users to securely access a command-line shell on a remote computer or server over an unsecured network. SSH provides strong encryption and authentication capabilities, making it an ideal method for securely administering commands or transferring data between a client and server.

To make SSH even more secure, you can use SSH keys instead of a simple password to log in to a server. SSH keys consist of two long strings of characters, called a public/private key pair, that are much less susceptible than passwords are to brute force attacks. The public key is placed on the server you want to access, while the private key resides on the client. When you use SSH to log in to the server from the client, the key pair is used to authenticate the session.

In SL1, some Dynamic Applications of type "Snippet" use SSH to communicate with a remote device. To use these Dynamic Applications, you must define an SSH credential. This credential specifies the hostname or IP address of the system you want to monitor, the port number used to access that system, and the private key used for authentication.

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

What Does the Linux Base Pack PowerPack Monitor?

To monitor Linux systems with SSH using SL1, you must install the *Linux Base Pack PowerPack*. This PowerPack enables you to discover, model, and collect data about Linux systems.

The *Linux Base Pack PowerPack* includes:

- Dynamic Applications that discover and collect configuration and performance data for Linux systems
- Internal collection Dynamic Applications for Linux systems
- Event Policies and corresponding alerts that are triggered when Linux systems meet certain status criteria
- Device Classes for each type of Linux system monitored
- A Run Book Action and an Automation policy to assign the proper device classes to Linux systems
- A Device Template for discovering Linux devices

Installing the Linux Base Pack PowerPack

To monitor Linux systems with SSH, you must import and install the latest version of the *Linux Base Pack PowerPack*.

TIP: By default, installing a new version of a PowerPack overwrites all content from a previous version of that PowerPack that has already been installed on the target system. You can use the **Enable Selective PowerPack Field Protection** setting in the **Behavior Settings** page (System > Settings > Behavior) to prevent new PowerPacks from overwriting local changes for some commonly customized fields. (For more information, see the **System Administration** manual.)

IMPORTANT: The minimum required MySQL version is 5.6.0.

To download and install the PowerPack:

1. Search for and download the PowerPack from the **PowerPacks** page (Product Downloads > PowerPacks & SyncPacks) at the [ScienceLogic Support Site](#).
2. In SL1, go to the **PowerPacks** page (System > Manage > PowerPacks).
3. Click the **[Actions]** button and choose *Import PowerPack*. The **Import PowerPack** dialog box appears.
4. Click **[Browse]** and navigate to the PowerPack file from step 1.
5. Select the PowerPack file and click **[Import]**. The **PowerPack Installer** modal displays a list of the PowerPack contents.
6. Click **[Install]**. The PowerPack is added to the **PowerPacks** page.

NOTE: If you exit the **PowerPack Installer** modal without installing the imported PowerPack, the imported PowerPack will not appear in the **PowerPacks** page. However, the imported PowerPack will appear in the **Imported PowerPacks** modal. This page appears when you click the **[Actions]** menu and select *Install PowerPack*.

Upgrading the PowerPack and Removing Dynamic Applications

To upgrade the *Linux Base Pack PowerPack*, perform the following steps:

NOTE: Before you upgrade, you should check the thresholds for zombie processes and load average. The load average is compared to the threshold based on the normalized data per CPU.


1. Familiarize yourself with the Known Issues for this release in the current version's [Release Notes](#).
2. If you have not done so already, upgrade your SL1 system to the minimum version or later release required for the version of the PowerPack you are upgrading to.
3. Go to the **Device Manager** page (Devices > Device Manager) and disable all Linux devices.
4. Download the latest version of the *Linux Base Pack* PowerPack from the Support Site to a local computer.
5. Go to the **PowerPack Manager** page (System > Manage > PowerPacks). Click the **[Actions]** menu and choose *Import PowerPack*. When prompted, import the *Linux Base Pack*.
6. Click the **[Install]** button. Wait for about five minutes to ensure the virtual environment is created.
7. Return to the **Device Manager** page (Devices > Device Manager) and re-enable all Linux devices.


NOTE: Interface discovery only runs nightly, therefore interfaces will not immediately appear until that process runs. If you would like to manually run nightly discovery, SSH in to your Data Collector and run the following command:

```
sudo -u s-em7-core /opt/em7/bin/python /opt/em7/backend/discover_update.py
```

After installing the PowerPack, you must delete old Dynamic Applications from previous versions. In later versions of the *Linux Base Pack*, some Dynamic Applications replace Dynamic Applications in older versions. If these old Dynamic Applications are left enabled, they can drastically reduce the number of Linux devices supported by a Data Collector.

To remove Dynamic Applications from the *Linux Base Pack* PowerPack:

1. Go to the **PowerPack Manager** page (System > Manage > PowerPacks).
2. Locate the *Linux Base Pack* PowerPack and click its wrench icon (.
3. In the **PowerPack Properties** page, in the Navbar on the left side, select **Dynamic Applications**.
4. In the **Embedded Dynamic Applications** page, you will remove Dynamic Applications depending on which version of the *Linux Base Pack* you are upgrading from:

If you are upgrading from **version 102, 103, or 104** of the *Linux Base Pack* PowerPack, click the bomb icon () for the following Dynamic Applications:

- Linux: File System Performance
 - Linux: IC Availability
 - Linux: Interface Performance
 - Linux: Network Configuration
 - Linux: Performance Cache (Deprecated)
 - Linux: TCP Services Configuration
5. The content will be removed from the PowerPack and will now appear in the bottom pane.

NOTE: Deleting the Dynamic Applications will remove all historical data from your devices. If you need to retain their historical data, then you must at a minimum disable the Dynamic Applications. However, in all cases the "Linux: Performance Cache" Dynamic Application must be deleted.

Chapter

2

Monitoring Linux with SSH

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 of the menu options, click the Advanced menu icon (⋮).

The following sections describe how to configure and discover Linux devices for monitoring by SL1 using SSH and the *Linux Base Pack PowerPack*:

This chapter covers the following topics:

<i>Prerequisites for Monitoring Linux Devices with SSH</i>	9
<i>Linux Distributions Supported by the Linux Base Pack PowerPack</i>	9
<i>Configuring Linux Devices</i>	10
<i>Creating an SSH/Key Credential</i>	11
<i>Creating a PowerShell Credential in the Classic SL1 User Interface</i>	13
<i>Configuring the Linux Device Template</i>	15
<i>Discovering Linux Devices</i>	18
<i>Configuring Dynamic Applications for Monitoring</i>	21
<i>Relationships Between Component Devices</i>	25

Prerequisites for Monitoring Linux Devices with SSH

Before you can monitor Linux devices using the *Linux Base Pack* PowerPack, you must have the following information about the devices that have already been properly configured:

- IP addresses of the devices you want to monitor
- SSH private keys for the devices you want to monitor

To monitor devices with the *Linux Base Pack* PowerPack, you must do the following:

1. [Configure your Linux Devices](#)
2. [Create the Credentials](#)
3. [Configure the Template](#)
4. [Discover the Linux Devices](#)

NOTE: The *Linux Base Pack* PowerPack currently supports 425 devices per Data Collector.

Linux Distributions Supported by the Linux Base Pack PowerPack

Distribution	Supported Versions
Ubuntu	23
	22
	20
CentOS	8
	7
Red Hat Linux Enterprise	9
	8
	7
Oracle Linux Server	9
	8
	7
Debian GNU Linux	12
	11
	10
	9

Distribution	Supported Versions
Fedora Server	39
	38
	37
	36
	35
Amazon Linux	Amazon Linux 2
	Amazon Linux
SUSE Linux Enterprise Server	15
	12
Rocky Linux	9
	8

Configuring Linux Devices

Before creating your credentials, you must add the following permission to the sudo config file (`/etc/sudoers`) so the "Linux: Hardware Configuration" Dynamic Application will run without asking for the sudo password:

```
<username> ALL=(ALL) NOPASSWD:/usr/sbin/dmidecode
```

If you cannot enable `DMIDECODE`, you must disable the "Linux: Hardware Configuration" Dynamic Application.

NOTE: If you see the "Sorry, you must have a tty to run sudo" error message in your device logs, or your "Linux: Hardware Configuration" Dynamic Application is not collecting data even when configured with the "sudo dmidecode", you will need to configure the Tty Requirement in `/etc/sudoers`, in order to collect hardware configuration information. To do so, add the following line to the sudo config file:

```
Defaults:<username> !requiretty
```

NOTE: To collect information about password expiration, run the following command on the terminal of your Linux device (does not need sudo): `chage -l $(whoami)`

If the `chage -l $(whoami)` command asks for a password, you will need to disable it by editing the `/etc/pam.d/chage` file with the following:

```
from: auth required pam_shells.so
to: auth sufficient pam_shells.so
```

NOTE: To avoid error messages, check that a home directory exists for the Linux user.

Creating an SSH/Key Credential

To configure SL1 to monitor Linux devices using SSH, you must first create an SSH/Key credential. This credential allows the Dynamic Applications in the *Linux Base Pack PowerPack* to connect with a Linux device.

NOTE: If you are on an SL1 system prior to version 11.1.0, you will not be able to duplicate the sample credential. It is recommended that you create your new credentials using [the SL1 classic user interface](#) so you do not overwrite the sample credential.

To define an SSH/Key credential:

1. Go to the **Credentials** page (Manage > Credentials).
2. Locate the sample credential you want to use, then click its **[Actions]** icon (⋮) and select **Duplicate**. A copy of the credential, called **Linux Example Credential- copy** appears.
3. Click the **[Actions]** icon (⋮) for the credential copy and select **Edit**. The **Edit Credential** modal page appears.

The screenshot shows the 'Edit Credential' modal page. The form is titled 'Edit Credential' and contains several sections. The top section has a 'Name' field with the value 'Linux Example Credential - copy' and a 'Timeout (Sec)' field with the value '5000'. Below this is a section for 'All Organizations' with a toggle switch and a dropdown menu. The main section contains fields for 'Hostname/IP*' (value: '%D'), 'Port*' (value: '22'), 'Username' (value: 'server_user_name'), and 'Password' (masked with asterisks). There is also a field for 'Private Key (PEM Format)*' (masked with asterisks). A 'Save & Test' button is located at the bottom right of this section. To the right of the main form is a 'Credential Tester' section with a 'Select Credential Test' dropdown, a 'Select Collector' dropdown (value: 'CUG_AUTOMATION | RS-cloudDCU-80: 10.2.6.80'), and an 'IP or Hostname to test*' field. A 'Test Credential' button is located below these fields. At the bottom right of the entire modal, there is a 'Save & Close' button.

4. Supply values in the following fields:

- **Name.** Type a new name for your Linux credential.
- **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).** Keep the default value.
- **Hostname/IP.** Type the time, in milliseconds, after which SL1 will stop trying to communicate with the authenticating server.
- **Port.** Type the port number associated with the data you want to retrieve.

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

- **Username.** Type the username for an SSH or user account on the device to be monitored.
- **Password.** Type the password for an SSH user account on the device to be monitored.
- **Private Key (PEM Format).** Type or paste the SSH private key that you want SL1 to use, in PEM format.

NOTE: For PEM Keys with a Passphrase, you can use the "Password" field to set the Passphrase.


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

NOTE: The Linux Base Pack PowerPack supports a range of modern public key cryptography for authentication. The full list of supported algorithms is as follows: ssh-ed25519, ecdsa-sha2-nistp256, ecdsa-sha2-nistp384, ecdsa-sha2-nistp521, rsa-sha2-512, rsa-sha2-256, ssh-rsa, ssh-dsa. Some of these types of keys do not work with the SL1 Credential Tester, but will function correctly with the PowerPack.

Creating an SSH/Key Credential in the Classic SL1 User Interface

To configure SL1 to monitor Linux devices using SSH, you must first create an SSH/Key credential. This credential allows the Dynamic Applications in the *Linux Base Pack PowerPack* to connect with a Linux device.

To create an SSH/Key credential:

1. Go to the **Credential Management** page (System > Manage > Credentials).
2. Locate the **Linux Example Credential** credential and click its wrench icon (). The **Credential Editor** modal page appears:
3. Supply values in the following fields:

- **Credential Name.** Type a new name for the credential.
- **Hostname/IP.** Keep the default value. SL1 will replace the variable with the IP address of the device that is currently using the credential.
- **Port.** Type the port number associated with the data you want to retrieve.

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

- **Timeout (ms).** Type the time, in milliseconds, after which SL1 will stop trying to communicate with the authenticating server.
- **Username.** Type the username for an SSH or user account on the device to be monitored.
- **Password.** Type the password for an SSH user account on the device to be monitored.
- **Private Key (PEM Format).** Type or paste the SSH private key that you want SL1 to use, in PEM format.

NOTE: In the classic user interface, the private key field will only accept RSA formatted / styled keys to be saved. If you wish to create SSH credentials with a key in the OpenSSH format you must do so in the default SL1 user interface.

NOTE: For PEM Keys with a Passphrase, you can use the "Password" field to set the Passphrase.

NOTE: NOTE: The Linux Base Pack PowerPack supports a range of modern public key cryptography for authentication. The full list of supported algorithms is as follows: ssh-ed25519, ecdsa-sha2-nistp256, ecdsa-sha2-nistp384, ecdsa-sha2-nistp521, rsa-sha2-512, rsa-sha2-256, ssh-rsa, ssh-dsa. Some of these types of keys do not work with the SL1 Credential Tester, but will function correctly with the PowerPack.

4. Click the **[Save As]** button, and then click **[OK]**.

Creating a PowerShell Credential in the Classic SL1 User Interface

To configure SL1 to monitor Linux devices using Windows Active Directory and GSSAPI, you must first create a PowerShell credential. This credential allows the Dynamic Applications in the *Linux Base Pack PowerPack* to connect with a Linux device using an Active Directory user.

To create a PowerShell credential:

1. Go to the **Credential Management** page (System > Manage > Credentials).
2. Locate the "Linux Kerberos - Example" credential and click its wrench icon (🔧). The **Credential Editor** modal page appears:

The screenshot shows the 'Credential Editor [86]' window. The title bar indicates 'Edit PowerShell Credential #86'. There are 'New' and 'Reset' buttons in the top right. The form is divided into two sections: 'Basic Settings' and 'Active Directory Settings'.
Basic Settings:
 - Profile Name: Linux Kerberos - Example
 - Account Type: [Active Directory]
 - Hostname/IP: %D
 - Timeout(ms): 10000
 - Username: USERNAME_GOES_HERE
 - Password: [masked]
 - Encrypted: [yes]
 - Port: 22
 - PowerShell Proxy Hostname/IP: [empty]
Active Directory Settings:
 - Active Directory Hostname/IP: AD_HOSTNAME_GOES_HERE
 - Domain: DOMAIN_GOES_HERE
 At the bottom, there are 'Save' and 'Save As' buttons.

3. Supply values in the following fields:
 - **Credential Name.** Type a new name for the credential.
 - **Hostname/IP.** Keep the default value. SL1 will replace the variable with the IP address of the device that is currently using the credential.
 - **Port.** Type the port number associated with the data you want to retrieve; it will be used to authenticate by SSH using GSSAPI option. The default TCP port for SSH servers is 22.
 - **Timeout (ms).** Type the time, in milliseconds, after which SL1 will stop trying to communicate with the authenticating server.
 - **Username.** Type the Active Directory username for an SSH on the device to be monitored.
 - **Password.** Type the Active Directory password for an SSH on the device to be monitored.
 - **Active Directory Hostname/IP.** Type the Active Directory hostname, IP, or fully qualified domain name (FQDN).
 - **Domain.** Type the Domain of the network.
4. Click the **[Save As]** button, then click **[OK]**.

Before you begin monitoring with this type of credential, it's necessary to configure the following:

- Active Directory Server with the Linux Machines included.
- DNS Server with the Linux Machines included.
- GSSAPI option enabled in the `/etc/ssh/sshd_config` file of the target Linux machine.

```
GSSAPIAuthentication yes
```

```
GSSAPICleanupCredentials yes # optional
```


NOTE: If the option `use_fully_qualified_names` is enabled in the target Linux machine, you need to type the username in the credential including the domain, for example: `user@DOMAIN.COM`

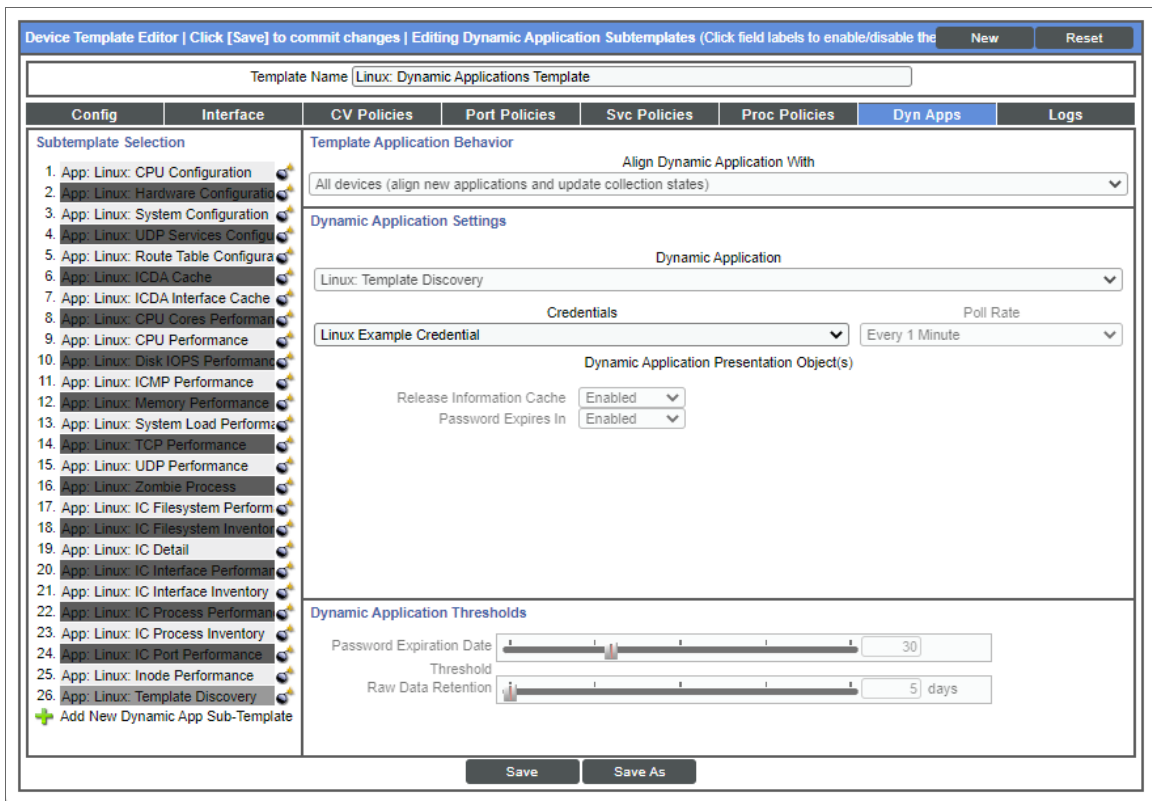
Configuring the Linux Device Template

A **device template** allows you to save a device configuration and apply it to multiple devices. You must use the **Linux: Dynamic Applications Template** device template in the discovery session to align all of the PowerPack's Dynamic Applications.

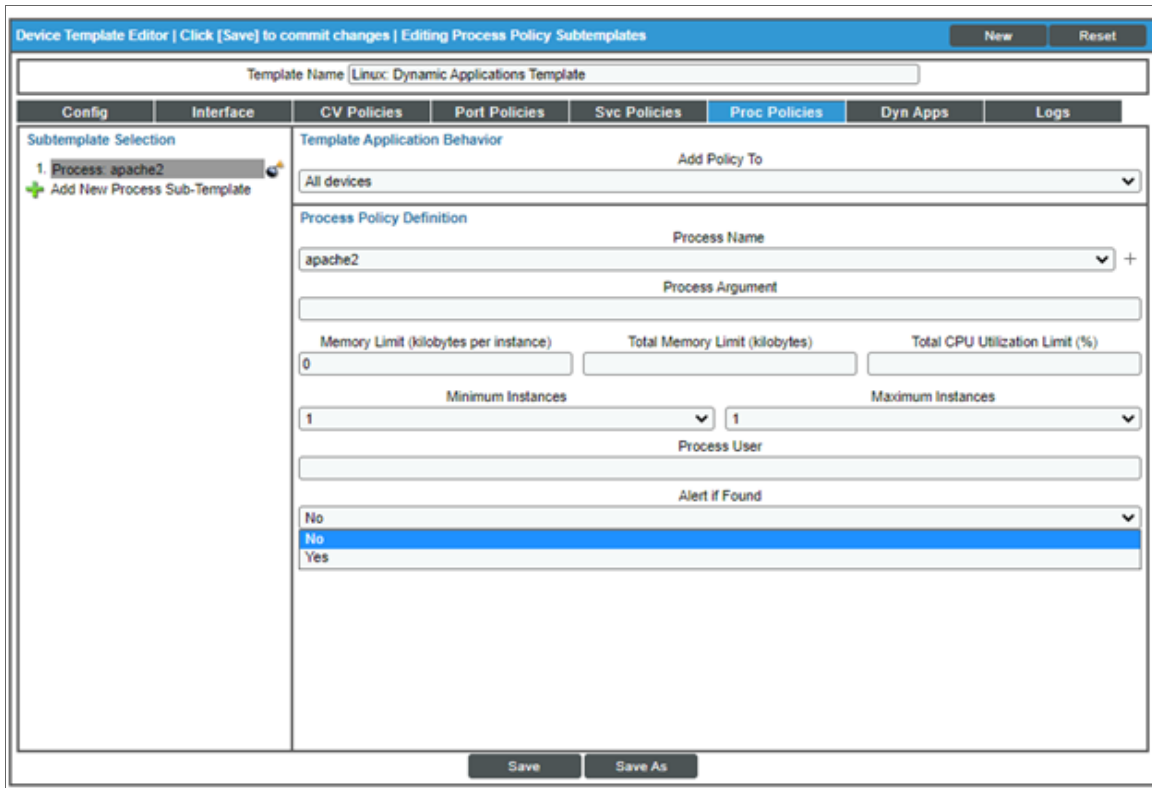
NOTE: When using the device template, ensure that only Linux devices will be discovered. Any device found during discovery will cause SL1 to apply the template to the device, resulting in Linux Dynamic Applications aligning to non-Linux devices.

To configure the Linux device template:

1. Go to the **Configuration Templates** page (Devices > Templates or Registry > Devices > Templates in the SL1 classic user interface).
2. Locate the "Linux: Dynamic Applications Template" device 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.



4. Click the "Linux: Template Discovery" Dynamic Application listed in the **Subtemplate Selection** section on the left side of the page and then click the **Credentials** field label to enable editing.
5. Select the Linux credential you created in the **Credentials** field. Repeat this step for all Dynamic Applications. All Dynamic Applications should be aligned to the credentials you created.
6. Enter a new name for the template in the **Template Name** field.
7. Click **[Save As]**.
8. Optionally, you can use the template to pre-configure Process policies and TCP/IP Port policies. To do this while configuring the template, click the **[Port Policies]** or the **[Proc Policies]** tabs and fill out the relevant fields for your policy. For more information on creating port monitoring policies and process monitoring policies with the device template, see the *Creating a Device Template* section of the **Device Groups and Device Templates** manual.



NOTE: You must rename the sample templates and click **[Save As]** to save it. If you do not rename the device template, then your device template will be overwritten the next time you upgrade the *Linux Base PackPowerPack*.

Configuring the Linux: IC Port Performance Dynamic Application

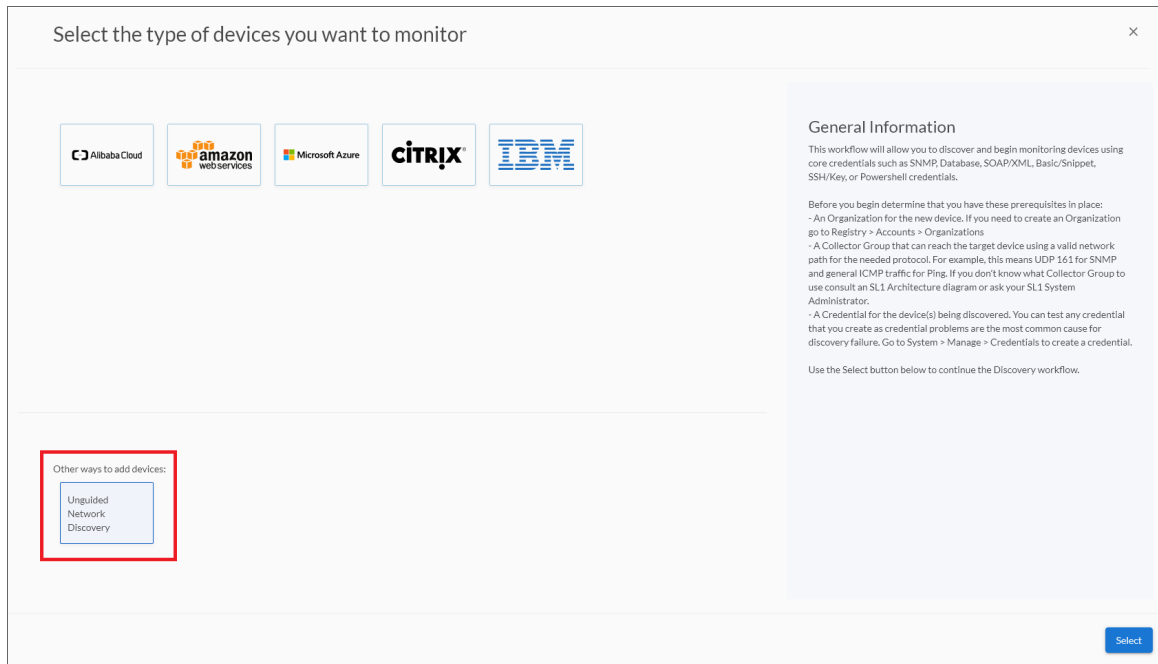
To use the "Linux: IC Port Performance" Dynamic Application, you will need to create a TCP/IP Port monitoring policy after running the discovery session. To create the TCP/IP Policy:

1. After running your discovery session, go to the **TCP/IP Port Monitoring** page (Registry > Monitors > TCP-IP Ports).
2. Click the **[Create]** button to open the **Create New TCP/IP Port Policy** page.
3. In the **Create New TCP/IP Port Policy** page, fill out the following fields:
 - **Select IP Device.** Select the Linux device with the ports you want to monitor.
 - **Port/Service.** Select the port you want to monitor from the dropdown menu.
 - Click the **[Save]** button.
4. You will see the ports monitored in the **[Performance]** tab of the **Device Summary** page.

Discovering Linux Devices

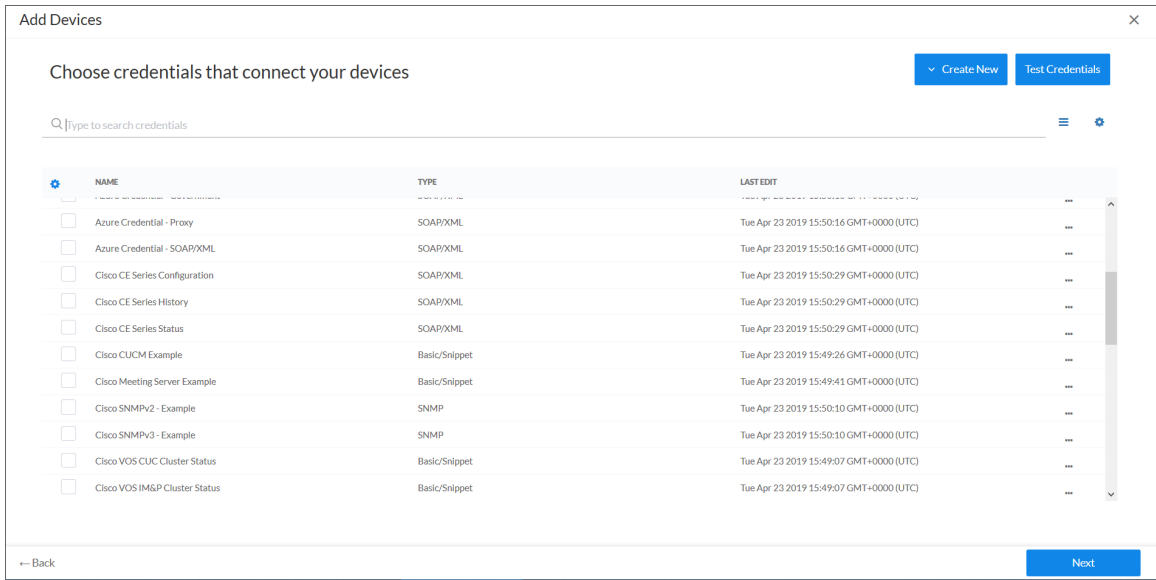
To discover Linux devices, perform the following steps:

1. On the **Devices** page (🖨️) or the **Discovery Sessions** page (Devices > Discovery Sessions), click the **[Add Devices]** button. The **Select** page appears:



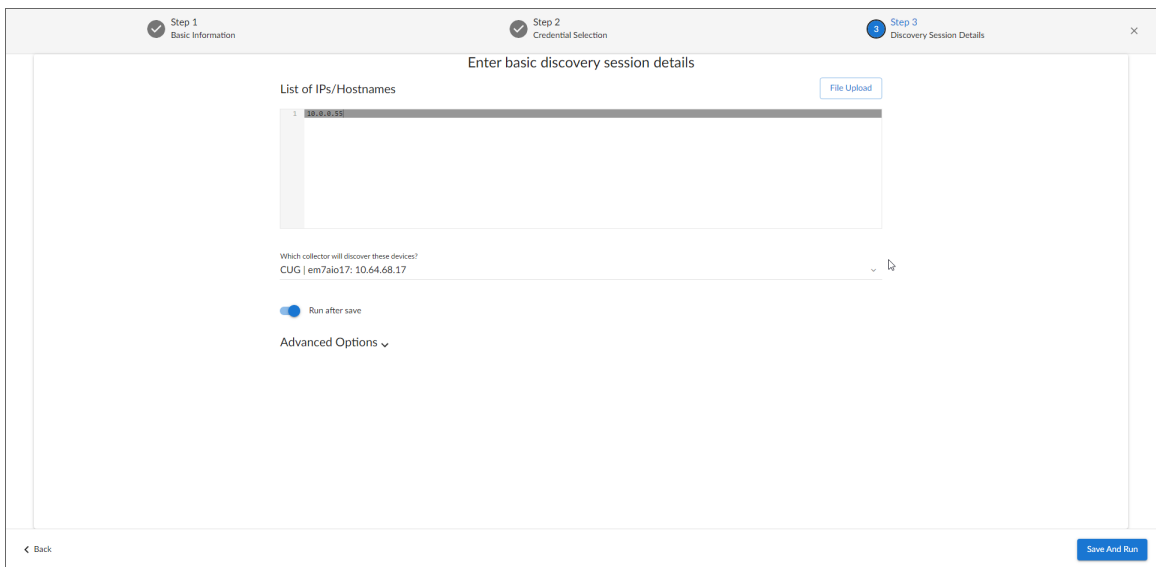
2. Click the **[Unguided Network Discovery]** button. Additional information about the requirements for discovery appears in the **General Information** pane to the right.
3. Click **[Select]**. The **Add Devices** page appears.
4. Complete the following fields:
 - **Name.** Type a unique name for this discovery session. This name is displayed in the list of discovery sessions on the **[Discovery Sessions]** tab.
 - **Description.** Optional. Type a short description of the discovery session. You can use the text in this description to search for the discovery session on the **[Discovery Sessions]** tab.
 - **Select the organization to add discovered devices to.** Select the name of the organization to which you want to add the discovered devices.

5. Click **[Next]**. The **Credentials** page of the **Add Devices** wizard appears:



6. On the **Credentials** page, locate and select the **SSH/Key credential** you created for the Linux devices.

7. Click **[Next]**. The **Discovery Session Details** page of the **Add Devices** wizard appears:



8. Complete the following fields:

- **List of IPs/Hostnames.** Type the IP addresses for the Linux devices you want to monitor.
- **Which collector will monitor these devices?** Select an existing collector to monitor the discovered devices. Required.
- **Run after save.** Select this option to run this discovery session as soon as you save the session.

In the **Advanced options** section, click the down arrow icon (▼) to complete the following fields:

- **Discover Non-SNMP**. Enable this setting.
 - **Model Devices**. Enable this setting.
 - **Select Device Template**. Select *the device template that you configured*.
9. Click **[Save and Run]** if you enabled the Run after save setting, or **[Save and Close]** to save the discovery session. The **Discovery Sessions** page (Devices > Discovery Sessions) displays the new discovery session.
 10. If you selected the **Run after save** option on this page, the discovery session runs, and the **Discovery Logs** page displays any relevant log messages. If the discovery session locates and adds any devices, the **Discovery Logs** page includes a link to the **Device Investigator** page for the discovered device.

Discovering Linux Devices in the SL1 Classic User Interface

To discover Linux devices using a discovery session, perform the following steps:

1. Go to the **Discovery Control Panel** page (System > Manage > Classic Discovery).
2. In the **Discovery Control Panel**, click the **[Create]** button.
3. The **Discovery Session Editor** page appears. On this page, define values in the following fields:

The screenshot shows the 'Discovery Session Editor | Editing Session [3]' interface. It is divided into several sections:

- Identification Information:** Name: Linux_Discovery, Description: (empty).
- IP and Credentials:** IP Address/Hostname Discovery List: 10.2.6.145, 10.2.6.148, 10.2.6.167, 10.2.6.133, 10.2.6.152, 10.2.6.147. Includes an 'Upload File' section with a 'Browse...' button.
- SNMP Credentials:** A list of SNMP credentials including 'Cisco SNMPv2 - Example', 'Cisco SNMPv3 - Example', 'Cisco. CSP SNMP Port 161 Example', 'Cisco. CSP SNMP Port 1610 Examp', 'Dell EMC. Isilon SNMPv2 Example', 'EM7 Default V2', 'EM7 Default V3 MD5', and 'EM7 Default V3 SHA'.
- Other Credentials:** A list of other credentials including 'vmmware Server Example', 'SSH/Key', 'Cisco. CSP 2100 CLI Example', 'Cisco. Dial Peer - Example', 'Linux AWS', 'Linux cloud', 'Linux Example Credential', 'Linux Local', and '[[Linux_Cred]]'.
- Detection and Scanning:** Initial Scan Level: [0. Model Device Only]. Scan Throttle: [System Default (recommended)]. Port Scan All IPs: [System Default (recommended)]. Port Scan Timeout: [System Default (recommended)]. Detection Method & Port: [Default Method] (includes UDP: 161 SNMP, TCP: 1 - tcpmux, TCP: 2 - compressnet, TCP: 3 - compressnet, TCP: 5 - rje, TCP: 7 - echo, TCP: 9 - discard, TCP: 11 - systat, TCP: 13 - daytime, TCP: 15 - netstat). Interface Inventory Timeout (ms): 600000. Maximum Allowed Interfaces: 10000. Bypass Interface Inventory: (unchecked).
- Basic Settings:** Discover Non-SNMP: (checked). Model Devices: (checked). DHCP: (unchecked). Device Model Cache TTL (h): 2. Collection Server PID: 4. [RS-AUTO-DCU-64]. Organization: [[LinuxOrg]]. Add Devices to Device Group(s): None, LayerX Appliances, Servers. Apply Device Template: [Linux: Dynamic Applications Template]. Log All: (unchecked).

Buttons at the bottom include 'Save', 'Save As', and 'Log All'.

- **IP Address Discovery List**. Type the IP addresses for the Linux devices you want to monitor, separated by a comma.

- **Other Credentials.** Select the SSH/Key credential you created for the Linux devices.
 - **Initial Scan Level.** Select *0. Model Device Only*.
 - **Discover Non-SNMP.** Select this checkbox.
 - **Model Devices.** Select this checkbox.
 - **Apply Device Template.** Select *the device template that you configured*.
4. Optionally, you can enter values in the other fields on this page. For more information about the other fields on this page, see the **Discovery & Credentials** manual.
 5. Click the **[Save]** button to save the discovery session and then close the **Discovery Session Editor** window.
 6. The discovery session you created appears at the top of the **Discovery Control Panel** page. Click its lightning-bolt icon (⚡) to run the discovery session.
 7. The **Discovery Session** window appears. When the Linux devices are discovered, click their device icons (🖨) to view the **Device Properties** pages for the Linux devices.

NOTE: The "Linux: IC Interface Inventory" Dynamic Application runs during nightly discovery. If you want to force discovery of interfaces at a time outside of nightly discovery, run the following command on the collector:

```
sudo -u s-em7-core /opt/em7/bin/python /opt/em7/backend/discover_update.py
```

Configuring Dynamic Applications for Monitoring

Process Monitoring with the Linux Base Pack

You can utilize the *Linux Base Pack PowerPack* for process monitoring in SL1. To learn more about system processes and creating system process monitoring policies, see the *Monitoring System Processes* section in the **Monitoring Device Infrastructure Health** manual.


Configuring Collection Frequency for Linux IC Dynamic Applications

The Linux IC Dynamic Applications use results from a different command from the rest of the Dynamic Applications in the PowerPack. The results of the command create a list of Filesystems mounted on the target Linux machine that is updated every two hours.


To change the collection frequency of the "Linux: IC Filesystem Inventory" Dynamic Application:

1. Go to the **Process Manager** page (System > Settings > Admin Processes or System > Settings > Processes in the SL1 classic user interface).
2. Search for the "Data Collection: Host Filesystem Inventory" process and click its wrench icon (🔧).
3. In the **Process Editor** window, use the **Frequency** dropdown field to select a new frequency.
4. Click the **[Save]** button.

To change the collection frequency of the "Linux: IC Filesystem Performance" Dynamic Application:

1. Go to the **Process Manager** page (System > Settings > Admin Processes or System > Settings > Processes in the SL1 classic user interface).
2. Search for the "Data Collection: Filesystem statistics" process and click its wrench icon ()
3. In the **Process Editor** window, use the **Frequency** dropdown field to select a new frequency.
4. Click the **[Save]** button.


To change the collection frequency of the "Linux: IC Detail" Dynamic Application:

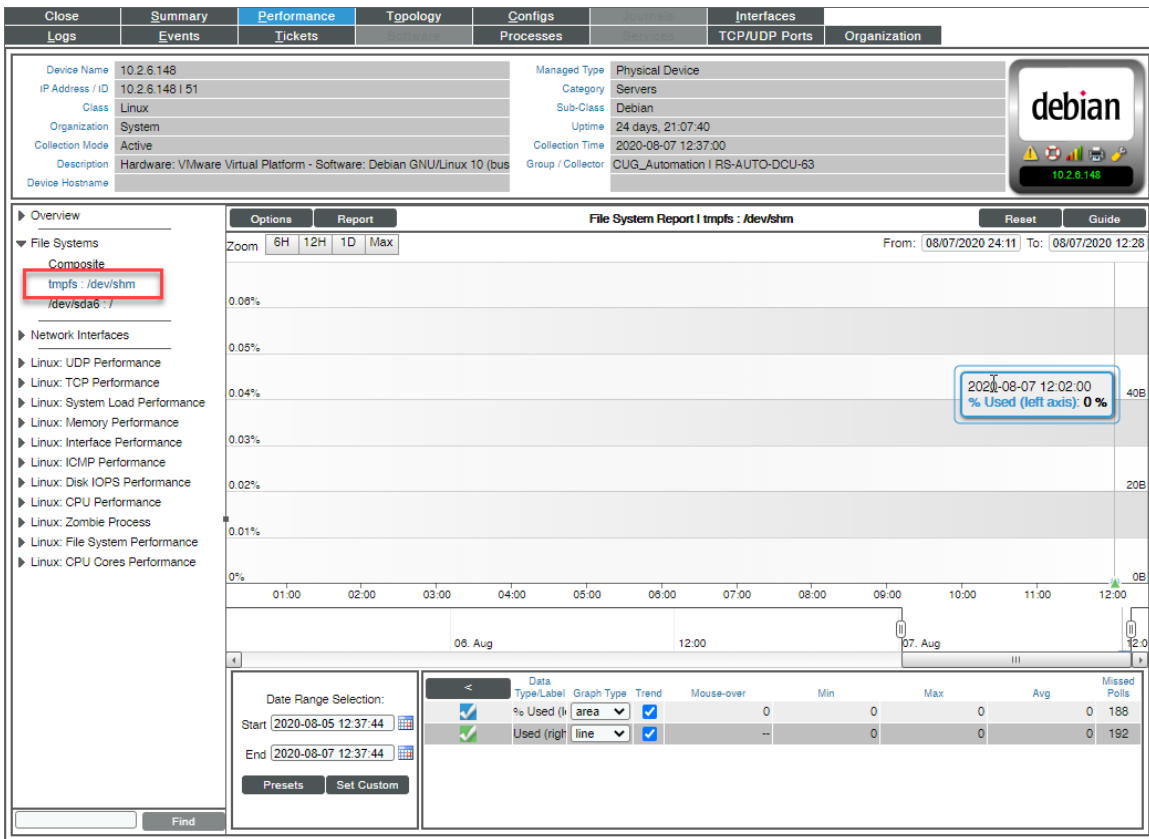
1. Go to the **Process Manager** page (System > Settings > Admin Processes or System > Settings > Processes in the SL1 classic user interface).
2. Search for the "Data Collection: SNMP Detail" process and click its wrench icon ()
3. In the **Process Editor** window, use the **Frequency** dropdown field to select a new frequency.
4. Click the **[Save]** button.

Unhiding Linux File Systems

In the **Device Hardware** page (Registry > Devices > Hardware), you can see view the size of the file system, the mount point with the name of the mounted file system, the format of the file system, and whether or not the file system is hidden.

To unhide the file system:

1. Go to the **Device Hardware** page (Devices > Hardware or Registry > Devices > Hardware in the SL1 classic user interface).
2. Find the file system you want to hide and select its checkbox.
3. In the **Select Actions** menu, select *UNHIDE File systems*.
4. Click the **[Go]** button to apply your changes.
5. Click the graph icon () next to the file system to open the **Device Summary** page.
6. Click the **[Performance]** tab.
7. You will see the unhidden file system listed in the left pane.



Configuring Linux File System Thresholds

To change the file system threshold:

1. Go to the **Device Hardware** page (Devices > Hardware or Registry > Devices > Hardware in the SL1 classic user interface).
2. Find the file system you want to hide and select its checkbox.
3. In the **Select Actions** menu, select **UNHIDE File systems**.
4. Click the **[Go]** button to apply your changes.
5. Click the wrench icon (🔧) next to the file system to open the **Device Properties** page.
6. Click the **[Thresholds]** tab.
7. In the **Device Thresholds** page, scroll down to the **File System Thresholds** section.
8. Find the threshold you want to edit and drag the sliders to adjust the threshold(s).
9. Click **[Save]** to save the threshold(s).

The screenshot displays a monitoring interface for a device. The top navigation bar includes tabs for Close, Properties, Thresholds, Collections, Monitors, and Schedule. Below this, there are sub-tabs for Logs, Toolbox, Interfaces, Relationships, Tickets, Redirects, Notes, and Attributes.

The main content area is divided into two sections:

- Device Properties:**
 - Device Name: 10.2.6.141
 - IP Address / ID: 10.2.6.141 | 5
 - Class: Linux
 - Organization: Linux
 - Collection Mode: Active
 - Description: (empty)
 - Device Hostname: (empty)
 - Managed Type: Physical Device
 - Category: Servers
 - Sub-Class: Ubuntu 16.04.2 LTS
 - Uptime: 0 days, 00:00:00
 - Collection Time: 2020-07-02 14:58:00
 - Group / Collector: CUG2 | RS-AUTO-DCU-64
- Device Thresholds:**
 - File System Thresholds:**

Path	Threshold	Current Usage
tmpfs : /dev/shm [Major]	85 %	1.0%
tmpfs : /dev/shm [Critical]	95 %	1.0%
/home/em7admin/Private	85 %	6.0%
/home/em7admin [Major]	95 %	6.0%
/home/em7admin/Private	95 %	6.0%
tmpfs : /run [Major]	85 %	12.0%
tmpfs : /run [Critical]	95 %	12.0%
/dev/sda1 : / [Major]	0.000 %	6.0%
/dev/sda1 : / [Critical]	10.000 %	6.0%
/dev/sr0 : /media/cdrom [Major]	90.000 %	100.0%
/dev/sr0 : /media/cdrom [Critical]	100.000 %	100.0%
 - Interface Inventory Thresholds:**
 - Interface Inventory Timeout: 600000 ms (Default: 600000 ms)
 - Maximum Allowed Interfaces: 10000 interfaces (Default: 10000 interfaces)

Aligning the Linux: SSH Cache Worker Dynamic Application

After updating to Linux Base Pack PowerPack version 111, you must align the "Linux: SSH Cache Worker" Dynamic Application to continue monitoring. This Dynamic Application acts as a cache producer for all the Dynamic Applications.

Devices discovered through a discovery session with the "Linux: Configuration Discovery" Dynamic Application aligned will automatically align with the "Linux: SSH Cache Worker" Dynamic Application in the next poll. However, if the Dynamic Applications were aligned using a template, you will need to set up the "Linux: SSH Cache Worker" Dynamic Application manually.

To align the "Linux: SSH Cache Worker" Dynamic Application using a template:

1. Create a new template adding the "Linux: SSH Cache Worker" Dynamic Application and credential.
 - Go to the **Device Template** (Registry>Devices>Template) and click **[Create]**. The Device Template Editor modal opens.
 - Enter a template name in the **Template Name** field.
 - On the **[Dyn Apps]** tab click **Add New Dynamic App Sub-Template** in the left **Subtemplate** menu.
 - In the **Dynamic Application** drop-down field, select "Linux: SSH Cache Worker".
 - In the **Credentials** drop-down field, select *ssh-cred*.
 - Click **[Save]**.

2. Apply the template to align the "Linux: SSH Cache Worker" Dynamic Application to multiple devices.
 - Go to the **Device Manager** (Registry>Device Manager) and select the checkbox of the devices you want to align.
 - In the **Select Action** menu at the bottom of the page, select *MODIFY by Template*. Next, click **[Go]**. The **Bulk Device Configuration** modal appears.
3. In the **Template** field, select the template you created in the previous steps and then click **[Apply]**.
4. Click **[Confirm]** to align the "Linux: SSH Cache Worker" Dynamic Application to your selected devices.

Relationships Between Component Devices



The Dynamic Applications in the *Linux Base Pack* PowerPack can automatically build relationships between Linux servers and other associated devices:

- If you discover AppDynamics applications using the Dynamic Applications in the *Cisco: AppDynamics* PowerPack, SL1 will automatically create relationships between Linux Servers and AppDynamics Nodes.
- If you discover Dynatrace environments using the Dynamic Applications in the *Dynatrace* PowerPack, SL1 will automatically create relationships between Linux Servers and Dynatrace Hosts.
- If you discover New Relic devices using the Dynamic Applications in the *New Relic* PowerPack, SL1 will automatically create relationships between Linux Servers and New Relic Servers.

Configuring Syslog for Linux

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 of the menu options, click the Advanced menu icon ().

The following sections describe how to configure syslog for Linux:

IMPORTANT: The following sections describe a general method for configuring syslog for Linux, which may not apply to your specific distribution. Please contact your Linux distribution vendor for specific instructions on how to perform syslog forwarding. For information about configuring your message collectors to accept inbound messages, see [Daily Health Tasks](#).

This chapter covers the following topics:

What is Syslog?	26
Configuring Syslog for Linux	27

What is Syslog?

Syslog is a protocol and utility for capturing and logging system information. This system information can be stored locally, remotely, or both. syslog allows a server to forward log messages over the network to SL1. SL1 then uses these messages to monitor the health of the server and trigger events (if necessary).

Because the syslog utility is mature and widely-used, there is an array of commercial and open source implementations. This chapter provides only a basic outline of how to configure syslog to send messages to SL1.

Entries in a syslog can include the following severity descriptions:

Severity	Description
0 Emergency:	System is unusable. A "panic" condition. Notify all technical staff. Affects multiple servers, applications, systems, or sites. For example, an outage caused by an earthquake.
1 Alert	Failure in primary system. Immediate action is required. Notify appropriate staff. Example would be "loss of backup ISP connection".
2 Critical	Failure in primary system. Immediate action is required before problem escalates to "alert". For example, "loss of primary ISP connection".
3 Error	Non-urgent failure. Action is required but not urgent. These messages should be relayed to appropriate support staff for resolution.
4 Warning	Indication that an error is about to occur. Action is required but not immediately. For example, "file system is 85% full".
5 Notice	Normal but significant condition. No immediate action required. Events that are unusual but are not considered error conditions. Should be examined to spot potential problems.
6 Informational	Normal operational messages. No action required. These may be harvested for reporting, measuring through-put, etc.
7 Debug	Information that is useful to developers for debugging the application; not useful during operations.

Configuring Syslog for Linux

To configure your Linux server to send syslogs to SL1, you must edit the file `/etc/syslog.conf`.

1. Before editing the `/etc/syslog.conf` file, ensure that syslog is enabled. To do this, open a shell session, log in as root, and enter the following at the command prompt:

```
service syslog status
```

2. Backup the existing `/etc/syslog.conf` file. To do this, open a shell session, log in as root, and enter the following at the command prompt:

```
cp /etc/syslog.conf /etc/syslog.orig
```

3. Use your favorite editor to edit the `/etc/syslog.conf` file and add the following line:

- If you are using an All-In-One Appliance, use the IP address of the All-In-One Appliance.
- If you are using a Distributed System and the Collector Group that will monitor your device includes a Message Collector, use the IP address of the Message Collector.

- If you are using a Distributed System and the Collector Group that will monitor your device includes a single Data Collector that performs the message collection function, use the IP address of the Data Collector.

*.err;local0.debug;daemon.notice;mail.crit @<IP_OF_SCIENCELOGIC_APPLIANCE>

NOTE: syslog includes many facilities. The facilities referenced above are merely a starting point as suggested by ScienceLogic.

4. After you edit the `syslog.conf` file, you must **restart the syslog service**. To do this, open a shell session and enter the following at the command prompt:

```
service syslog restart
```

5. To test sending syslog messages to SL1, open an shell session and enter the following at the command prompt:

```
logger -p local0.debug "Test Debug Message to SL1"
```

6. To see if the message was sent to SL1, check:

- on the Linux device, the file `/var/log/messages`
- in SL1, the device logs of the corresponding Linux device.

NOTE: By default, SL1 includes multiple event policies based on syslog messages. ScienceLogic recommends that you review these policies to ensure that they suit your business needs. To view these policies, go to Registry > Events > Event Manager. Use the sort and filter tools to view all policies of type "syslog." From the same page, you can edit these event policies or create your own event policies based on syslog messages. For more information on event policies, see the manual on **Events**.

Appendix



4

Collection Objects

Overview

This appendix defines the different collection objects in the *Linux Base Pack PowerPack*.

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 of the menu options, click the Advanced menu icon ().

This appendix covers the following topics:

This chapter covers the following topics:

Collection Objects in Linux Dynamic Applications	30
--	----

Collection Objects in Linux Dynamic Applications

The following tables list the Collection Objects included in those Dynamic Applications and the Linux commands used by each of those objects. You can use these commands to grant or restrict access to certain data types on the user account you will use to monitor your Linux devices.

The following table is a list of configuration and performance Dynamic Applications in the PowerPack:

Dynamic Application	Collection Object	Linux Command
Linux: Configuration Discovery		Determines if a device is a Linux system before discovery in SL1. If the device is not a Linux system, it will not be discovered.
Linux: CPU Configuration	All	<code>cat /proc/cpuinfo/ lscpu</code>
	Server Product Name	<code>cat /sys/devices/virtual/dmi/id/product_ name</code>
Linux: CPU Cores Performance	All	<code>cat /proc/stat</code>
Linux: CPU Performance	All	<code>cat /proc/stat</code>
Linux: Disk IOPs Performance	All	<code>cat /proc/diskstats</code>
Linux: Hardware Configuration	All	<code>sudo /usr/sbin/dmidecode -qt 1,2,3</code>
Linux: ICMP Performance	All	<code>cat /proc/net/snmp</code>
Linux: Inode Performance	All	<code>timeout 3 df -iPT</code>
Linux: Memory Performance	All	<code>cat /proc/meminfo</code>
Linux: Route Table Configuration	All	<code>ip route 2>/dev/null /sbin/ip route 2>/dev/null</code>

Dynamic Application	Collection Object	Linux Command	
Linux: System Configuration	Kernel Version	cat /proc/sys/kernel/osrelease	
	Distribution Genus	cat 2> /dev/null /etc/os-release grep PRETTY_NAME cat 2> /dev/null /etc/redhat-release cat 2> /dev/null /etc/lsb-release grep DISTRIB_DESCRIPTION cat /etc/SuSE-release	
	Host Name	cat /proc/sys/kernel/hostname	
	Distribution Release	cat /etc/os-release grep PRETTY_NAME	
	AppDynamics Host Name IP Address	hostname=\$(cat /proc/sys/kernel/hostname) && echo \$hostname "<silos:ip>	
	AppDynamics Namespace	echo "appdynamics/ns"	
	Architecture Type	uname -a	
	Compiler	cat /proc/version	
	Domain Name	cat /proc/sys/kernel/domainname && cat /proc/sys/kernel/hostname	
	Dynatrace Hostname	cat /proc/sys/kernel/hostname	
	Dynatrace Namespace	echo "dynatrace/physical/ns"	
	New Relic Hostname	cat /proc/sys/kernel/hostname	
	New Relic Namespace	echo "newrelic/server/ns"	
	Release Date	cat /proc/sys/kernel/version	
	SMP Support	cat /proc/sys/kernel/version	
	Time Zone	date "+%Z"	
	Total Physical Memory (MBytes)	cat /proc/meminfo	
	Total Swap Memory (MBytes)	cat /proc/meminfo	
	Linux: System Load Performance	All	cat /proc/loadavg
		CPU	lscpu
Linux: TCP Performance	All	cat /proc/net/snmp	

Dynamic Application	Collection Object	Linux Command
Linux: Template Discovery		Determines if a device is a Linux system before discovery in SL1. If the device is not a Linux system, it will not be discovered.
Linux: UDP Performance	All	<code>cat /proc/net/snmp</code>
Linux: UDP Services Configuration	All	<code>ss -luan 2>/dev/null /usr/sbin/ss -luan 2>/dev/null /bin/ss -luan 2>/dev/null</code>
Linux: Zombie Process	All	<code>ps aux grep Z</code>

The following table is a list of internal collection inventory and performance Dynamic Applications in the PowerPack:

Dynamic Application	Collection Object	Linux Command
Linux: IC Detail	All	Internal Collection that consumes data stored by the "Linux: ICDA Cache" Dynamic Application.
Linux: IC Filesystem Inventory	All	Internal Collection that consumes data stored by the "Linux: ICDA Cache" Dynamic Application.
Linux: IC Filesystem Performance	All	Internal Collection that consumes data stored by the "Linux: ICDA Cache" Dynamic Application.
Linux: IC Interface Inventory	All	Internal Collection that consumes data stored by the "Linux: ICDA Interface Cache" Dynamic Application.
Linux: IC Interface Performance	All	Internal Collection that consumes data stored by the "Linux: ICDA Interface Cache" Dynamic Application.
Linux: IC Port Performance	All	Internal Collection that consumes data stored by the "Linux: ICDA Cache" Dynamic Application.
Linux: IC Process Inventory	All	Internal Collection that consumes data stored by the "Linux: ICDA Interface Cache" Dynamic Application.
Linux: IC Process Performance	All	Internal Collection that consumes data stored by the "Linux: ICDA Interface Cache" Dynamic Application.
Linux: ICDA Cache	Azure Host	<code>[[\$(cat /sys/devices/virtual/dmi/id/chassis_asset_tag) == "7783-7084-3265-9085-8269-3286-77"]] && echo "Azure" echo "False"</code>
	Filesystem	<code>timeout 3 df -kPT</code>
	Hardware Config Product Name	<code>cat /sys/devices/virtual/dmi/id/product_name</code>
	Network Interfaces	<code>for x in `ls -ld /sys/class/net/* grep -v '/virtual/' rev cut -d/ -f1 rev`; do echo \$x ': ' `if [</code>

Dynamic Application	Collection Object	Linux Command
		<code>"\$x" = "lo"]; then echo "0"; else cat /sys/class/net/\$x/speed 2>/dev/null echo "0" ; fi`; done;</code>
	Network Interfaces IP Address	<code>/sbin/ip addr</code>
	Uptime	<code>cat /proc/uptime</code>
Linux: ICDA Interface Cache	Interface Stats	<code>/sbin/ip -s -s link</code>
	Process	<code>ps aux</code>
	TCP Listening Ports	<code>echo ".";ss -ltn 2</dev/null /usr/sbin/ss -ltn 2</dev/null /bin/ss -ltn 2>/dev/null</code>

© 2003 - 2024, ScienceLogic, Inc.

All rights reserved.

LIMITATION OF LIABILITY AND GENERAL DISCLAIMER

ALL INFORMATION AVAILABLE IN THIS GUIDE IS PROVIDED "AS IS," WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED. SCIENCELOGIC™ AND ITS SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT.

Although ScienceLogic™ has attempted to provide accurate information on this Site, information on this Site may contain inadvertent technical inaccuracies or typographical errors, and ScienceLogic™ assumes no responsibility for the accuracy of the information. Information may be changed or updated without notice. ScienceLogic™ may also make improvements and / or changes in the products or services described in this Site at any time without notice.

Copyrights and Trademarks

ScienceLogic, the ScienceLogic logo, and EM7 are trademarks of ScienceLogic, Inc. in the United States, other countries, or both.

Below is a list of trademarks and service marks that should be credited to ScienceLogic, Inc. The ® and ™ symbols reflect the trademark registration status in the U.S. Patent and Trademark Office and may not be appropriate for materials to be distributed outside the United States.

- ScienceLogic™
- EM7™ and em7™
- Simplify IT™
- Dynamic Application™
- Relational Infrastructure Management™

The absence of a product or service name, slogan or logo from this list does not constitute a waiver of ScienceLogic's trademark or other intellectual property rights concerning that name, slogan, or logo.

Please note that laws concerning use of trademarks or product names vary by country. Always consult a local attorney for additional guidance.

Other

If any provision of this agreement shall be unlawful, void, or for any reason unenforceable, then that provision shall be deemed severable from this agreement and shall not affect the validity and enforceability of any remaining provisions. This is the entire agreement between the parties relating to the matters contained herein.

In the U.S. and other jurisdictions, trademark owners have a duty to police the use of their marks. Therefore, if you become aware of any improper use of ScienceLogic Trademarks, including infringement or counterfeiting by third parties, report them to Science Logic's legal department immediately. Report as much detail as possible about the misuse, including the name of the party, contact information, and copies or photographs of the potential misuse to: legal@sciencelogic.com. For more information, see <https://sciencelogic.com/company/legal>.

ScienceLogic

800-SCI-LOGIC (1-800-724-5644)

International: +1-703-354-1010