



Network Connectivity PowerPack

Network Connectivity PowerPack version 102

Table of Contents

Introduction	3
What is the Network Connectivity PowerPack?	3
Installing the Network ConnectivityPowerPack	3
Network Connectivity Automation Policies	5
Standard Automation Policies	5
Standard Ping Automation Policy	9
Standard Traceroute Automation Policy	10
Standard NSLOOKUP Automation Policy	12
Standard NMAP Automation Policies	13
Run NMAP on Affected Port	13
Run NMAP on Common Port List	14
Run NMAP on Monitored Ports	15
Creating and Customizing Automation Policies	17
Prerequisites	18
Creating an Automation Policy	18
Example Automation Configuration	21
Customizing an Automation Policy	22
Removing an Automation Policy from a PowerPack	24
Customizing Network Connectivity Actions	25
Creating a Custom Action Policy with Network Connectivity Actions	26
Customizing Ping Actions	27
Custom Ping Action Parameters	27
Custom Ping Action Examples	28
Customizing Traceroute Actions	30
Custom Traceroute Action Parameters	31
Custom Traceroute Action Examples	31
Customizing NSLOOKUP Actions	32
Custom NSLOOKUP Action Parameters	32
Custom NSLOOKUP Action Examples	33
Customizing NMAP Actions	34
Custom NMAP Action Parameters	34
Custom NMAP Action Examples	35
Run Book Variables	36
Run Book Variables	36

Chapter

1

Introduction

Overview

This manual describes how to use the automation policies, automation actions, and custom action types found in the *Network Connectivity PowerPack*.

This chapter covers the following topics:

What is the Network Connectivity PowerPack?	3
Installing the Network ConnectivityPowerPack	3

What is the Network Connectivity PowerPack?

The *Network Connectivity PowerPack* enriches SL1 network connectivity events, such as availability and latency issues, by automatically running common network diagnostic commands and adding the output to the SL1 event log or an associated incident. This PowerPack includes custom action types for running ping, traceroute, nslookup, and nmap commands with parameters that you specify.

The *Network Connectivity PowerPack* does not contain or require credentials to operate. The Network Connectivity actions are executed from the SL1 All-In-One Appliance or Data Collector.

Installing the Network ConnectivityPowerPack

Before completing the steps in this manual, you must import and install the latest version of the *Network ConnectivityPowerPack*.

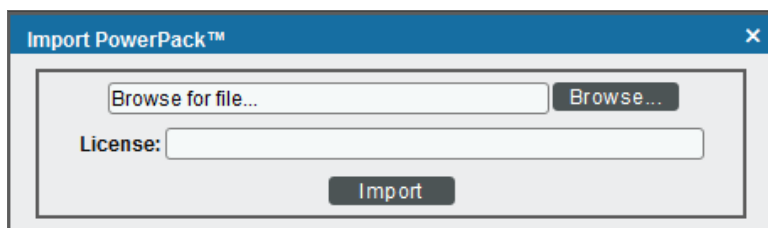
NOTE: The *Network Connectivity* PowerPack requires SL1 version 8.10.0 or later. For details on upgrading SL1, see the appropriate SL1 [Release Notes](#).

CAUTION: You must install version 101 of the Datacenter Automation Utilities PowerPack before proceeding.

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.)

To download and install a PowerPack:

1. Download the PowerPack from the [ScienceLogic Customer Portal](#).
2. Go to the **PowerPack Manager** page (System > Manage > PowerPacks).
3. In the **PowerPack Manager** page, click the **[Actions]** button, then select *Import PowerPack*.
4. The **Import PowerPack** dialog box appears:



5. Click the **[Browse]** button and navigate to the PowerPack file.
6. When the **PowerPack Installer** modal page appears, click the **[Install]** button to install the PowerPack.

NOTE: If you exit the **PowerPack Installer** modal without installing the imported PowerPack, the imported PowerPack will not appear in the **PowerPack Manager** 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*.

TIP: To use the standard automation policies, no other configuration is necessary. These automation policies run in response to network connectivity-related events that are included in SL1.

Network Connectivity Automation Policies

Overview

This chapter describes how to use the automation policies, automation actions, and custom action types found in the *Network Connectivity PowerPack*.

This chapter covers the following topics:

Standard Automation Policies	5
<i>Standard Ping Automation Policy</i>	9
<i>Standard Traceroute Automation Policy</i>	10
<i>Standard NSLOOKUP Automation Policy</i>	12
<i>Standard NMAP Automation Policies</i>	13
Run NMAP on Affected Port	13
Run NMAP on Common Port List	14
Run NMAP on Monitored Ports	15

Standard Automation Policies

The *Network Connectivity PowerPack* includes six standard automation policies, shown in the figure below. These automation policies run automatically in response to network availability events to diagnose problems. To use these standard policies, you do not have to do any additional configuration after you install the PowerPack.

Editing PowerPack™ Network Connectivity PowerPack										
Manage PowerPack™										
Embedded Run Book Policies [6]										
Automation Policy Name	ID	Policy State	Organization	Devices	Events	Actions	Edited By	Last Edited		
1. Network Connectivity: Run NMAP on Affected Port	190	Enabled	System	All	2	2	em7admin	2020-02-13 15:17:23		
2. Network Connectivity: Run NMAP on Common Ports	191	Enabled	System	All	7	2	em7admin	2020-02-13 15:17:23		
3. Network Connectivity: Run NMAP on Monitored Ports	192	Enabled	System	All	6	2	em7admin	2020-02-13 15:17:23		
4. Network Connectivity: Run Nslookup (IP)	103	Enabled	System	All	9	2	em7admin	2020-02-13 15:17:23		
5. Network Connectivity: Run Ping (IPv4)	102	Enabled	System	All	9	2	em7admin	2020-02-13 15:17:23		
6. Network Connectivity: Run Traceroute	101	Enabled	System	All	9	2	em7admin	2020-02-13 15:17:23		
Available Run Book Policies [1]										
Automation Policy Name	ID	Policy State	Organization	Devices	Events	Actions	Edited By	Last Edited		
1. Email for CPU 100	120	Enabled	System	2	5	1	em7admin	2019-11-27 09:45:34		

The following table shows the standard automation policies, their aligned events, and the automation action that runs by default in response to the events.

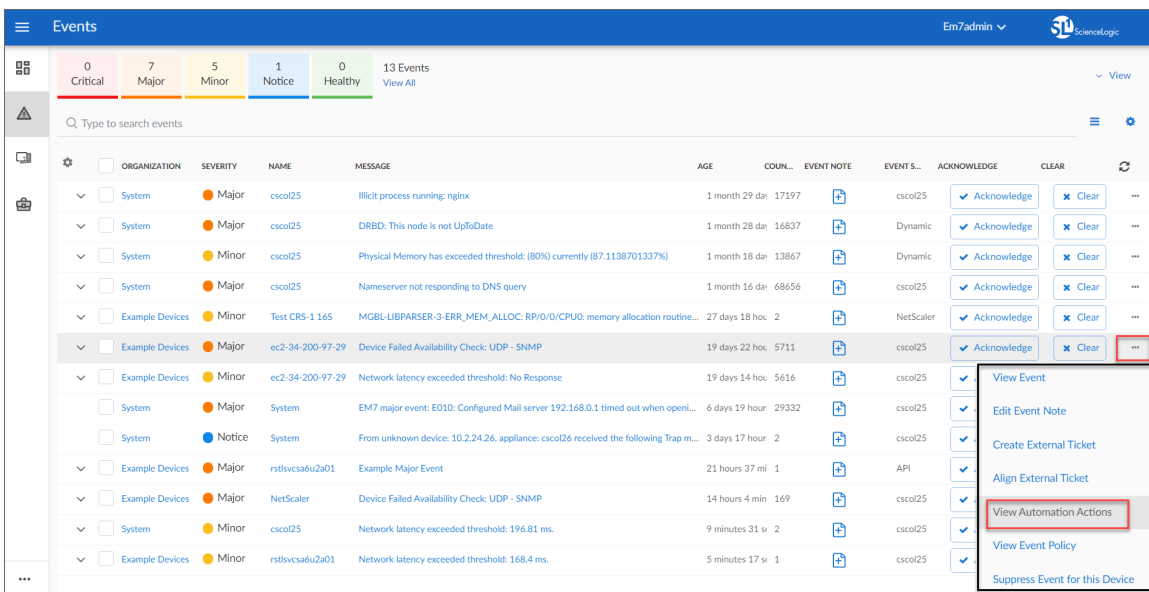
Automation Policy Name	Aligned Events	Automation Action (Default)
Network Connectivity: Run NMAP on Affected Port	<ul style="list-style-type: none"> • Poller: TCP/UDP port not responding • Poller: TCP/UDP port not responding (SMTP) 	<ul style="list-style-type: none"> • Run NMAP: Single Port from Event • Enrichment: Util: Format Command Output as HTML
Network Connectivity: Run NMAP on Common Ports	<ul style="list-style-type: none"> • Poller: Availability and Latency checks failed • Poller: Device not responding to ping (high frequency) • Poller: Availability Check Failed • Poller: Availability Flapping • Poller: TCP/UDP port not responding • Poller: TCP/UDP port not responding (SMTP) • Transactions: Round trip mail did not arrive within threshold 	<ul style="list-style-type: none"> • Run NMAP: Common Port List • Enrichment: Util: Format Command Output as HTML
Network Connectivity: Run NMAP on Monitored Ports	<ul style="list-style-type: none"> • Poller: Availability and Latency checks failed • Poller: Device not responding to ping (high frequency) • Poller: Availability Check Failed 	<ul style="list-style-type: none"> • Run NMAP: Monitored Ports • Enrichment: Util: Format Command Output as HTML

Automation Policy Name	Aligned Events	Automation Action (Default)
	<ul style="list-style-type: none"> • Poller: Availability Flapping • Poller: TCP/UDP port not responding • Poller: TCP/UDP port not responding (SMTP) 	
Network Connectivity: Run Nslookup (IPv4)	<ul style="list-style-type: none"> • Poller: Availability and Latency checks failed • Poller: Availability Check Failed • Poller: Availability Flapping • Poller: Device not responding to ping (high frequency) • Poller: DNS hostname resolution time above threshold • Poller: Failed to resolve hostname • Poller: TCP/UDP port not responding • Poller: TCP/UDP port not responding (SMTP) • Transactions: Round trip mail did not arrive within threshold 	<ul style="list-style-type: none"> • Run Nslookup: Default Options • Enrichment: Util: Format Command Output as HTML
Network Connectivity: Run Ping (IPv4)	<ul style="list-style-type: none"> • Poller: Availability and Latency checks failed • Poller: Availability Check Failed • Poller: Availability Flapping • Poller: Device not responding to ping (high frequency) • Poller: Network Latency Exceeded Threshold • Poller: TCP connection time above threshold • Poller: TCP/UDP port not responding • Poller: TCP/UDP port not responding (SMTP) • Transactions: Round trip mail did not arrive within threshold 	<ul style="list-style-type: none"> • Run Ping: Default Options • Enrichment: Util: Format Command Output as HTML
Network Connectivity: Run Traceroute (IPv4)	<ul style="list-style-type: none"> • Poller: Availability and Latency checks failed • Poller: Availability Check Failed 	<ul style="list-style-type: none"> • Run Traceroute: Default Options

Automation Policy Name	Aligned Events	Automation Action (Default)
	<ul style="list-style-type: none"> • Poller: Availability Flapping • Poller: Device not responding to ping (high frequency) • Poller: Network Latency Exceeded Threshold • Poller: TCP connection time above threshold • Poller: TCP/UDP port not responding • Poller: TCP/UDP port not responding (SMTP) • Transactions: Round trip mail did not arrive within threshold 	<ul style="list-style-type: none"> • Enrichment: Util: Format Command Output as HTML

For every device that has an IP address, SL1 monitors availability every five minutes. If you have enabled Critical Ping for a device and enabled the event "Poller: Device not responding to ping (high frequency)", you can monitor availability at a higher frequency than five minutes. The automation policies included in this PowerPack respond to events from Critical Ping, as well.

The following figure shows some network availability events on the **Events** page:



To see the automation actions triggered by an event, click the **[Actions]** button (≡) and select *View Automation Actions*. The **Event Actions Log** page appears. Notice the highlighted Traceroute and Nslookup information in the following figure. The log indicates that the following actions ran successfully:

- Run Nslookup: Default Options and Enrichment: Util: Format Command Output as HTML
- Run Traceroute: Default Options and Enrichment: Util: Format Command Output as HTML

The screenshot displays the 'Event Actions Log' for event [51606]. It contains five entries:

- 2020-02-10 19:48:11:** Automation Policy Network Connectivity: Run Ping (IPv4) action Run Ping: Default Options ran Successfully. Message CustomActionType (379) executed without incident. Result: {command_list_out: [{"ping -c 5 10.2.24.30", "PING 10.2.24.30 (10.2.24.30) 56(84) bytes of data: \nFrom 10.2.24.26 icmp_seq=1 Destination Host Unreachable\nFrom 10.2.24.26 icmp_seq=2 Destination Host Unreachable\nFrom 10.2.24.26 icmp_seq=3 Destination Host Unreachable\nFrom 10.2.24.26 icmp_seq=4 Destination Host Unreachable\nFrom 10.2.24.26 icmp_seq=5 Destination Host Unreachable\n\n-- 10.2.24.30 ping statistics --\n5 packets transmitted, 0 received, +5 errors, 100% packet loss, time 3998ms\npipe 3\n", None]}}
- 2020-02-10 19:48:11:** Automation Policy Network Connectivity: Run Nslookup (IPv4) action Enrichment: Util: Format Command Output as HTML ran Successfully. Message Snippet (365) executed without incident. Result: Enrichment Command Output. Command: nslookup 10.2.24.30. Output: *** server can't find 30.24.2.10.in-addr.arpa: NXDOMAIN
- 2020-02-10 19:48:11:** Automation Policy Network Connectivity: Run Traceroute (IPv4) action Enrichment: Util: Format Command Output as HTML ran Successfully. Message Snippet (365) executed without incident. Result: Enrichment Command Output. Command: traceroute 10.2.24.30. Output: traceroute to 10.2.24.30 (10.2.24.30), 30 hops max, 60 byte packets\n1 csc0126 (10.2.24.26) 2847.197 ms !H 2847.157 ms !H 2847.141 ms !H
- 2020-02-10 19:47:56:** Automation Policy Network Connectivity: Run Traceroute (IPv4) action Run Traceroute: Default Options ran Successfully. Message CustomActionType (378) executed without incident. Result: {command_list_out: [{"traceroute 10.2.24.30", "traceroute to 10.2.24.30 (10.2.24.30), 30 hops max, 60 byte packets\n1 csc0126 (10.2.24.26) 2847.197 ms !H 2847.157 ms !H 2847.141 ms !H\n", None]}}
- 2020-02-10 19:47:56:** Automation Policy Network Connectivity: Run Nslookup (IPv4) action Run Nslookup: Default Options ran Successfully. Message CustomActionType (380) executed without incident. Result: {command_list_out: [{"nslookup 10.2.24.30", "*** server can't find 30.24.2.10.in-addr.arpa: NXDOMAIN\n\n", None]}}

TIP: Although you can edit the automation actions described in this section, best practice is to "Save As" to create a new, renamed automation action, instead of customizing the standard automation policies.

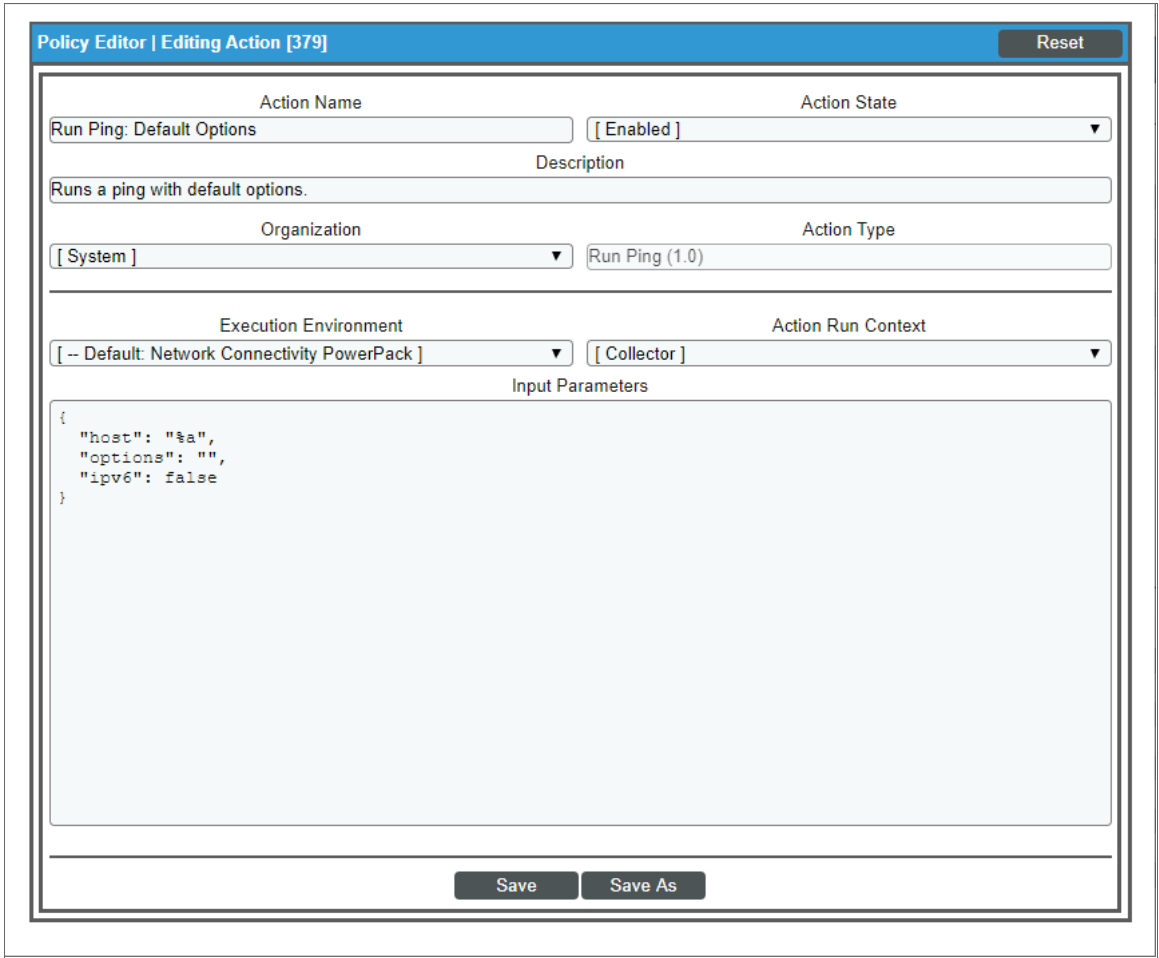
Standard Ping Automation Policy

The "Network Connectivity: Run Ping (IPv4)" automation policy is triggered by the following events:

- Poller: Availability and Latency checks failed
- Poller: Availability Check Failed
- Poller: Availability Flapping
- Poller: Device not responding to ping (high frequency)
- Poller: Network Latency Exceeded Threshold
- Poller: TCP connection time above threshold
- Poller: TCP/UDP port not responding
- Poller: TCP/UDP port not responding (SMTP)
- Transactions: Round trip mail did not arrive within threshold

Default Behavior. When these events occur, the automation policy "Network Connectivity: Run Ping (IPv4)" executes the action "Run Ping: Default Options" and formats the output with "Enrichment: Util: Format Command Output as HTML". This action runs a standard ping (IPv4) command automatically. The output of the command is formatted for display in the SL1 **Events** page, or in an incident ticket on an external system.

The following figure shows the details of the ping action:



Options. In some cases, you may want to modify the action that is run in response to the triggering events. For example, if you are monitoring an IPv6 network, you can select one of the Ping6 actions. The following ping actions are available in this PowerPack:

- Run Ping: Default Options
- Run Ping6: Default Options

For information about customizing automation policies, see [Customizing an Automation Policy](#).

Standard Traceroute Automation Policy

The "Network Connectivity: Run Traceroute (IPv4)" automation policy is triggered by the following events:

- Poller: Availability and Latency checks failed
- Poller: Availability Check Failed
- Poller: Availability Flapping
- Poller: Device not responding to ping (high frequency)
- Poller: Network Latency Exceeded Threshold
- Poller: TCP connection time above threshold
- Poller: TCP/UDP port not responding
- Poller: TCP/UDP port not responding (SMTP)
- Transactions: Round trip mail did not arrive within threshold

Default Behavior. When these events occur, the automation policy "Network Connectivity: Run Traceroute (IPv4)" executes the action "Run Traceroute: Default Options". This action runs a standard traceroute (IPv4) command automatically. The output of the command is formatted for display in the SL1 **Events** page, or in an incident ticket on an external system.

The following figure shows the details of the traceroute action:

The screenshot shows the 'Policy Editor | Editing Action [378]' window. The action is named 'Run Traceroute: Default Options' and is currently 'Enabled'. The description is 'Runs an IPv4 traceroute with default options.' The organization is set to '[System]' and the action type is 'Run Traceroute (1.0)'. The execution environment is '[-- Default: Network Connectivity PowerPack]' and the action run context is '[Collector]'. The input parameters are defined as a JSON object: { "host": "%a", "options": "", "packet_length": 0 }. The interface includes 'Save' and 'Save As' buttons at the bottom.

Field	Value
Action Name	Run Traceroute: Default Options
Action State	[Enabled]
Description	Runs an IPv4 traceroute with default options.
Organization	[System]
Action Type	Run Traceroute (1.0)
Execution Environment	[-- Default: Network Connectivity PowerPack]
Action Run Context	[Collector]
Input Parameters	<pre>{ "host": "%a", "options": "", "packet_length": 0 }</pre>

Options. In some cases, you may want to modify the action that is run in response to the triggering events. For example, if you are monitoring an IPv6 network, you can select one of the IPv6 traceroute actions. The following traceroute automation actions are available in this PowerPack:

- Run Traceroute: Default Options
- Run IPv6 Traceroute: Default Options

For information about customizing automation policies, see [Customizing an Automation Policy](#).

Standard NSLOOKUP Automation Policy

The "Network Connectivity: Run Nslookup (IPv4)" automation policy is triggered by the following events:

- Poller: Availability and Latency checks failed
- Poller: Availability Check Failed
- Poller: Availability Flapping
- Poller: Device not responding to ping (high frequency)
- Poller: DNS hostname resolution time above threshold
- Poller: Failed to resolve hostname
- Poller: TCP/UDP port not responding
- Poller: TCP/UDP port not responding (SMTP)
- Transactions: Round trip mail did not arrive within threshold

Default Behavior. When these events occur, the automation policy "Network Connectivity: Run Nslookup (IPv4)" executes the action "Run Nslookup: Default Options" and formats the output with "Enrichment: Util: Format Command Output as HTML". This action runs a standard NSLOOKUP (IPv4) command automatically. The output of the command is formatted for display in the SL1 **Events** page, or in an incident ticket on an external system.

The screenshot shows the 'Policy Editor | Editing Action [380]' window. At the top right is a 'Reset' button. The main area contains several fields:

- Action Name:** Run Nslookup: Default Options
- Action State:** [Enabled]
- Description:** Runs an nslookup with default options.
- Organization:** [System]
- Action Type:** Run Nslookup (1.0)
- Execution Environment:** [-- Default: Network Connectivity PowerPack]
- Action Run Context:** [Collector]
- Input Parameters:** A text area containing a JSON object:


```
{
  "host": "%a",
  "nameserver": "",
  "options": ""
}
```

At the bottom of the window are 'Save' and 'Save As' buttons.

Options. In some cases, you may want to modify the action that is run in response to the triggering events. For example, you can run NSLOOKUP with plaintext output.

For information about customizing automation policies, see [Customizing an Automation Policy](#).

Standard NMAP Automation Policies

Three NMAP automation policies are included with this PowerPack. Each policy is described in more detail in this section.

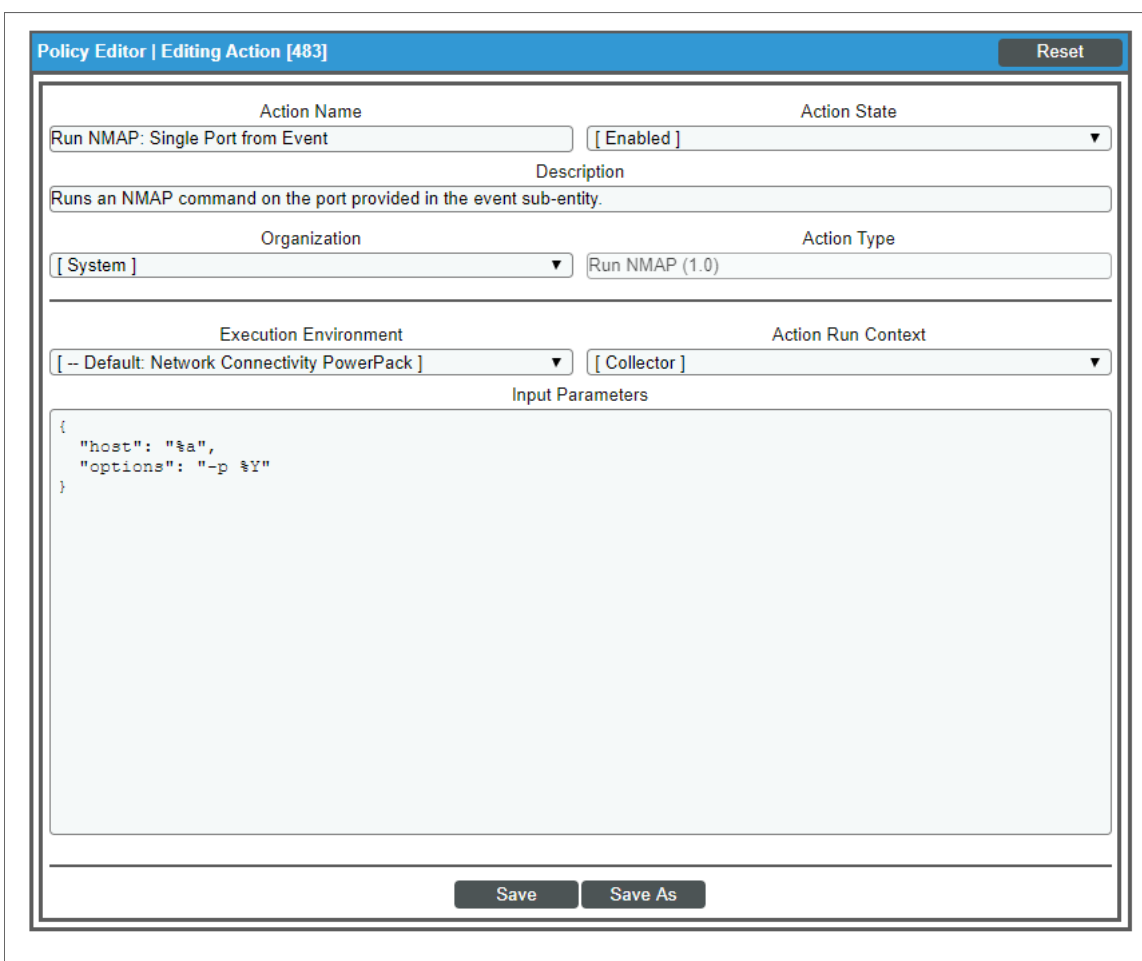
Run NMAP on Affected Port

The "Network Connectivity: Run NMAP on Affected Port" automation policy is triggered by the following events:

- Poller: TCP/UDP port not responding
- Poller: TCP/UDP port not responding (SMTP)

Default Behavior. When these events occur, the automation policy "Network Connectivity: Run NMAP on Affected Port" executes the action "Run NMAP: Single Port from Event" and formats the output with "Enrichment: Util: Format Command Output as HTML". This action runs a standard NMAP command on the port provided in the event. The output of the command is formatted for display in the SL1 **Events** page, or in an incident ticket on an external system.

The following figure shows the details of the NMAP action:



The screenshot shows the "Policy Editor | Editing Action [483]" window. It features a "Reset" button in the top right corner. The main area contains several fields and dropdown menus:

- Action Name:** Run NMAP: Single Port from Event
- Action State:** [Enabled]
- Description:** Runs an NMAP command on the port provided in the event sub-entity.
- Organization:** [System]
- Action Type:** Run NMAP (1.0)
- Execution Environment:** [-- Default: Network Connectivity PowerPack]
- Action Run Context:** [Collector]
- Input Parameters:** A text area containing a JSON object:

```
{
  "host": "%a",
  "options": "-p %Y"
}
```

At the bottom of the window, there are "Save" and "Save As" buttons.

For information about customizing automation policies, see [Customizing an Automation Policy](#).

Run NMAP on Common Port List

The "Network Connectivity: Run NMAP on Common Port List" automation policy is triggered by the following events:

- Poller: Availability and Latency checks failed
- Poller: Device not responding to ping (high frequency)
- Poller: Availability Check Failed

- Poller: Availability Flapping
- Poller: TCP/UDP port not responding
- Poller: TCP/UDP port not responding (SMTP)
- Transactions: Round trip mail did not arrive within threshold

Default Behavior. When these events occur, the automation policy "Network Connectivity: Run NMAP on Common Port List" executes the action "Run NMAP: Common Port List" and formats the output with "Enrichment: Util: Format Command Output as HTML". This action runs a standard NMAP command on ports 21, 22, 25, 53, 80, 443, 5985, and 5986. The output of the command is formatted for display in the SL1 **Events** page, or in an incident ticket on an external system.

The following figure shows the details of the NMAP action:

The screenshot shows the 'Policy Editor | Editing Action [484]' window. The action is named 'Run NMAP: Common Port List' and is currently 'Enabled'. The description is 'Runs an NMAP command using a list of common ports.' The organization is set to '[System]' and the action type is 'Run NMAP (1.0)'. The execution environment is '-- Default: Network Connectivity PowerPack' and the action run context is '[Database]'. The input parameters are defined in a JSON format:

```
{
  "host": "%a",
  "options": "-p 21,22,25,53,80,443,5985,5986"
}
```

Buttons for 'Save' and 'Save As' are visible at the bottom of the configuration area.

For information about customizing automation policies, see [Customizing an Automation Policy](#).

Run NMAP on Monitored Ports

The "Network Connectivity: Run NMAP on Monitored Ports" automation policy is triggered by the following events:

- Poller: Availability and Latency checks failed
- Poller: Device not responding to ping (high frequency)
- Poller: Availability Check Failed
- Poller: Availability Flapping
- Poller: TCP/UDP port not responding
- Poller: TCP/UDP port not responding (SMTP)

Default Behavior. When these events occur, the automation policy "Network Connectivity: Run NMAP on Monitored Ports" executes the action "Run NMAP: Monitored Ports" and formats the output with "Enrichment: Util: Format Command Output as HTML". This action runs a standard NMAP command on any ports that are currently monitored with a port monitoring policy on the triggering device. The output of the command is formatted for display in the SLI **Events** page, or in an incident ticket on an external system.

The following figure shows the details of the NMAP action:

The screenshot shows the 'Policy Editor | Editing Action [485]' window. The action is named 'Run NMAP: Monitored Ports' and is currently 'Enabled'. The description is 'Runs an NMAP command on the ports that are currently monitored on the device.' The organization is set to '[System]' and the action type is 'Run NMAP (1.0)'. The execution environment is '-- Default: Network Connectivity PowerPack' and the action run context is '[Collector]'. The input parameters are defined as a JSON object:

```
{
  "host": "%a",
  "options": "-p %_monitored_ports_%"}

```

 At the bottom of the editor are 'Save' and 'Save As' buttons.

For information about customizing automation policies, see [Customizing an Automation Policy](#).

Creating and Customizing Automation Policies

Overview

This chapter describes how to create automation policies using the automation actions in the *Network Connectivity PowerPack*.

This chapter covers the following topics:

<i>Prerequisites</i>	18
<i>Creating an Automation Policy</i>	18
<i>Example Automation Configuration</i>	21
<i>Customizing an Automation Policy</i>	22
<i>Removing an Automation Policy from a PowerPack</i>	24

Prerequisites

Before you create an automation policy using the automation actions in the *Network Connectivity* PowerPack, you must determine:

- Which commands (Ping, Traceroute, NSLOOKUP, or NMAP) you want to run on a device when an event occurs. There are eight automation actions in the PowerPack that run these commands with different options. You can also create your own automation actions using the custom action types supplied in the PowerPack.
- What event criteria you want to use to determine when the automation actions will trigger, or the set of rules that an event must match before the automation is executed. This can include matching only specific event policies, event severity, associated devices, and so on. For a description of all the options that are available in Automation Policies, see the *Run Book Automation* manual.

Creating an Automation Policy

To create an automation policy that uses the automation actions in the *Network Connectivity* PowerPack, perform the following steps:

1. Go to the **Automation Policy Manager** page (Registry > Run Book > Automation).

2. Click **[Create]**. The **Automation Policy Editor** page appears.

The screenshot shows the 'Automation Policy Editor | Creating New Automation Policy' interface. At the top right is a 'Reset' button. The main configuration area includes:

- Policy Name:** A text input field.
- Policy Type:** A dropdown menu with '[Active Events]' selected.
- Policy State:** A dropdown menu with '[Enabled]' selected.
- Policy Priority:** A dropdown menu with '[Default]' selected.
- Organization:** A dropdown menu with 'Example Devices' selected.
- Criteria Logic:** A series of dropdown menus: '[Severity >=]', '[Minor,]', '[and 5 minutes has elapsed]', '[since the first occurrence,]', '[and event is NOT cleared]', and 'and all times are valid'.
- Match Logic:** A dropdown menu with '[Text search]' selected.
- Match Syntax:** A text input field.
- Repeat Time:** A dropdown menu with '[Only once]' selected.
- Align With:** A dropdown menu with '[Devices]' selected.
- Include events for entities other than devices (organizations, assets, etc.)
- Trigger on Child Rollup

Below these settings are four panels for selecting items to be included in the policy:

- Available Devices:** A list of devices including 'Example Devices', 'Cisco Systems: CRS-1 16S: Test CRS-1 16S', 'Citrix: NetScaler: NetScaler', 'Ping: ICMP: ec2-34-200-97-29', 'Virtual Device: Domain Name: Test Device', 'Virtual Device: Domain Name: Test Device 2', 'Linux Devices', and 'Linux: CentOS Linux 7 (Core): 10.2.24.31'.
- Aligned Devices:** A list containing '(All devices)'. Navigation arrows are present between the available and aligned lists.
- Available Events:** A list of event IDs and descriptions such as '[5678] Critical: 3PAR Trap: Critical Alert', '[5649] Critical: 3PAR: Disk Utilization Exceeded Critical Threshold', '[3569] Critical: AKCP: AC Voltage sensor detects no current', '[3578] Critical: AKCP: DC Voltage sensor High Critical', '[3579] Critical: AKCP: DC Voltage sensor Low Critical', '[3568] Critical: AKCP: Dry Contact Sensor Low Critical', '[3574] Critical: AKCP: Smoke Detector Alert!', and '[3572] Critical: AKCP: Water Sensor has detected water'.
- Aligned Events:** A list containing '(All events)'. Navigation arrows are present between the available and aligned lists.
- Available Actions:** A list of actions including 'SNMP Trap [1]: EM7 Event Trap', 'SNMP Trap [1]: RBA Base Pack: Send Trap', 'SNMP Trap [1]: SL1 Event Trap', 'Create Ticket [2]: RBA Base Pack: Create Ticket', 'Snippet [5]: API VeloCloud initial disable', 'Snippet [5]: Automation Utilities: Calculate Memory Size for Each A', 'Snippet [5]: AWS: Disable Instance By Tag', and 'Snippet [5]: AWS: Discover from EC2 IP'.
- Aligned Actions:** An empty list with up and down navigation arrows.

A 'Save' button is located at the bottom center of the form.

3. Complete the following required fields:

- **Policy Name.** Enter a name for the automation policy.
- **Policy Type.** Select whether the automation policy will match events that are active, match when events are cleared, or run on a scheduled basis. Typically, you would select *Active Events* in this field.
- **Policy State.** Specifies whether the policy will be evaluated against the events in the system. If you want this policy to begin matching events immediately, select *Enabled*.
- **Policy Priority.** Specifies whether the policy is high-priority or default priority. These options determine how the policy is queued.

- **Organization**. Select one or more organizations to associate with the automation policy. The automation policy will execute only for devices in the selected organizations (that also match the other criteria in the policy). To configure a policy to execute for all organizations, select *System*.
- **Aligned Actions**. This field includes the actions from the *Network Connectivity* PowerPack. You should see Run Ping, Run Traceroute, Run Nslookup, and Run NMAP actions in this field.

To add an action to the **Aligned Actions** field, select the action in the **Available Actions** field and click the right arrow (>>). To re-order the actions in the **Aligned Actions** field, select an action and use the up arrow or down arrow buttons to change that action's position in the sequence. Select an output format action from the *Datacenter Automation Utilities* PowerPack.

CAUTION: Remember that you must include an output format action (from the *Datacenter Automation Utilities* PowerPack) for this action to produce output.

4. Optionally, supply values in the other fields on this page to refine when the automation will trigger.
5. Click **[Save]**.

NOTE: You can also modify one of the automation policies included with this PowerPack. Best practice is to use the **[Save As]** option to create a new, renamed automation policy, instead of customizing the standard automation policies.

If you modify one of the included automation policies and save it with the original name, the customizations in that policy will be overwritten when you upgrade the PowerPack unless you remove the association between the automation policy and the PowerPack before upgrading.

Example Automation Configuration

The following is an example of an automation policy that uses the automation actions in the *Network Connectivity PowerPack*:

The screenshot shows the 'Automation Policy Editor' for 'Editing Automation Policy [293]'. The interface includes a 'Reset' button in the top right. The main configuration area is divided into several sections:

- Policy Name:** Network Connectivity: Run Ping (IPv4)
- Policy Type:** [Active Events]
- Policy State:** [Enabled]
- Policy Priority:** [Default]
- Organization:** [System]

The **Criteria Logic** section contains a list of conditions:

- [Severity >=] [Minor,]
- [and no time has elapsed]
- [since the first occurrence,]
- [and event is NOT cleared]
- [and all times are valid]

The **Match Logic** section is set to [Text search].

The **Match Syntax** section is empty.

The **Repeat Time** is set to [Only once] and **Align With** is set to [Devices].

There is an unchecked checkbox for 'Include events for entities other than devices (organizations, assets, etc.)' and a checked checkbox for 'Trigger on Child Rollup'.

The **Available Devices** list includes: Example Devices, Cisco Systems: CRS-1 16S: Test CRS-1 16S, and Citrix: NetScaler: NetScaler. The **Aligned Devices** list contains (All devices).

The **Available Events** list includes: [5678] Critical: 3PAR Trap: Critical Alert, [5649] Critical: 3PAR: Disk Utilization Exceeded Critical Thresl, and [3569] Critical: AKCP: AC Voltage sensor detects no current. The **Aligned Events** list includes: [1934] Critical: Poller: Availability and Latency checks failed, [4071] Critical: Poller: Device not responding to ping (high freq, [1932] Major: Poller: Availability Check Failed, and [4011] Major: Poller: Availability Flapping.

The **Available Actions** list includes: SNMP Trap [1]: EM7 Event Trap, SNMP Trap [1]: RBA Base Pack: Send Trap, and SNMP Trap [1]: SL1 Event Trap. The **Aligned Actions** list includes: 1. Run Ping [113]: Run Ping: Default Options and 2. Snippet [5]: Datacenter Automation: Format Output as.

At the bottom, there are 'Save' and 'Save As' buttons.

The policy uses the following settings:


- **Policy Name.** The policy is named "Network Connectivity: My Run Traceroute (IPv6)".
- **Policy Type.** The policy runs when an event is in an active state. *Active Events* is selected in this field.
- **Policy State.** *Enabled* is selected in this field.
- **Organization.** The policy executes for all orgnaizations, so *System* is selected in this field.
- **Criteria Logic.** The policy is configured to execute immediately when an event matches these criteria: "Severity >= Minor, and no time has elapsed since the first occurrence, and event is NOT cleared, and all times are valid".
- **Aligned Devices.** The policy is configured to trigger for all devices in the system.
- **Aligned Events.** The policy is configured to trigger only when the following events are triggered:

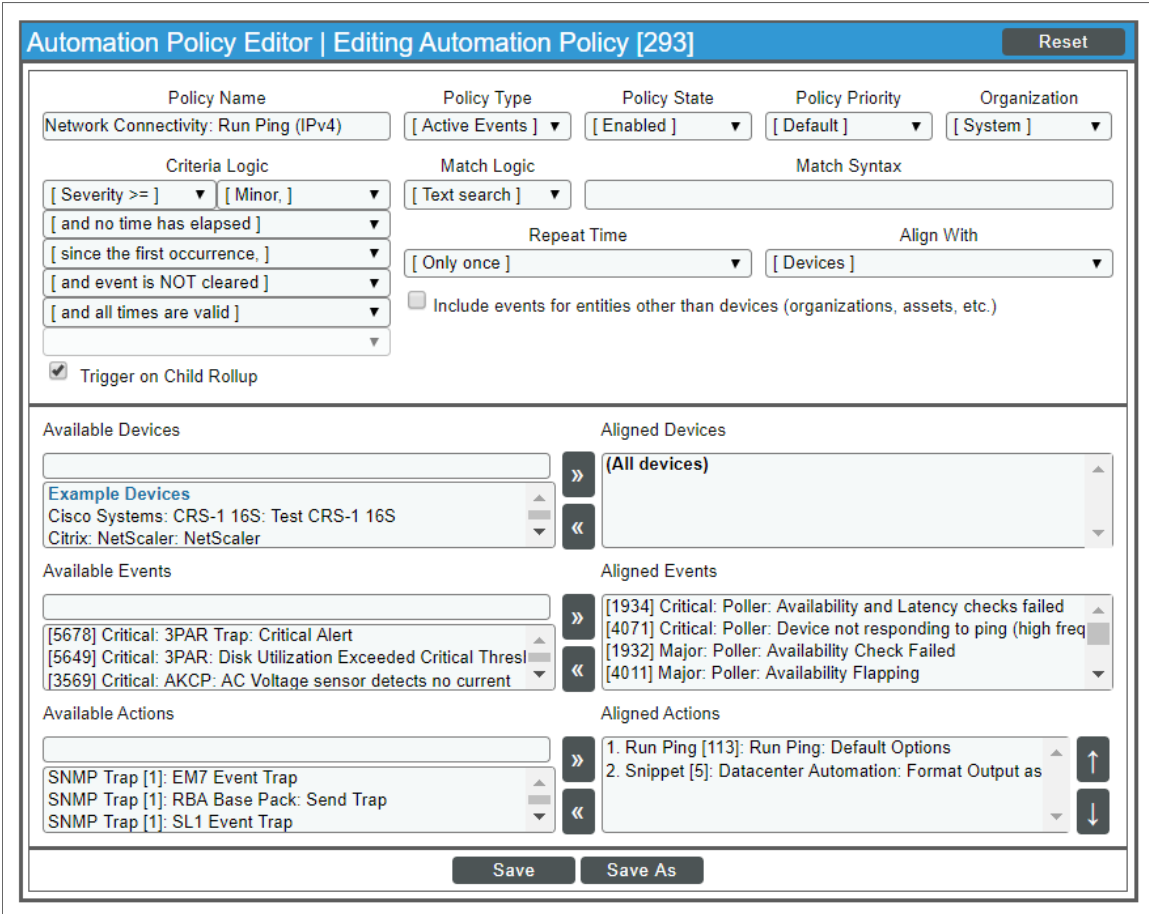
- Critical: Poller: Availability and Latency checks failed
 - Critical: Poller: Device not responding to ping (high frequency)
 - Major: Poller: Availability Check Failed
 - Major: Poller: Availability Flapping
 - Major: Poller: TCP/UDP port not responding (SMTP)
 - Major: Transactions: Round trip mail did not arrive within threshold
 - Minor: Poller: Network Latency Exceeded Threshold
 - Minor: Poller: TCP connections time above threshold
- **Aligned Actions.** The automation includes the following actions. The formatting action allows you to view the output of traceroute in the Automation Log, accessed through the SL1 Event Console:
 - Run Traceroute): Run IPv6 Traceroute: Default options
 - Enrichment: Util: Format Command Output as HTML

Customizing an Automation Policy

To customize an automation policy:

1. Go to the **Automation Policy Manager** page (Registry > Run Book > Automation).

2. Search for the *Network Connectivity* automation policy you want to edit and click the wrench icon () for that policy . The **Automation Policy Editor** page appears:



The screenshot shows the 'Automation Policy Editor' interface for editing a policy named 'Network Connectivity: Run Ping (IPv4)'. The interface includes several configuration sections:

- Policy Information:** Policy Name (Network Connectivity: Run Ping (IPv4)), Policy Type ([Active Events]), Policy State ([Enabled]), Policy Priority ([Default]), and Organization ([System]).
- Criteria Logic:** A series of dropdown menus for defining event criteria, such as [Severity >=], [Minor,], [and no time has elapsed], [since the first occurrence,], [and event is NOT cleared], and [and all times are valid].
- Match Logic:** Match Logic ([Text search]) and Match Syntax (empty field).
- Repeat Time and Align With:** Repeat Time ([Only once]) and Align With ([Devices]).
- Include events for entities other than devices (organizations, assets, etc.):** An unchecked checkbox.
- Trigger on Child Rollup:** A checked checkbox.
- Available Devices:** A list of devices including 'Cisco Systems: CRS-1 16S: Test CRS-1 16S' and 'Citrix: NetScaler: NetScaler'.
- Aligned Devices:** A list containing '(All devices)'.
- Available Events:** A list of events including '[5678] Critical: 3PAR Trap: Critical Alert', '[5649] Critical: 3PAR: Disk Utilization Exceeded Critical Thresl', and '[3569] Critical: AKCP: AC Voltage sensor detects no current'.
- Aligned Events:** A list of events including '[1934] Critical: Poller: Availability and Latency checks failed', '[4071] Critical: Poller: Device not responding to ping (high freq', '[1932] Major: Poller: Availability Check Failed', and '[4011] Major: Poller: Availability Flapping'.
- Available Actions:** A list of actions including 'SNMP Trap [1]: EM7 Event Trap', 'SNMP Trap [1]: RBA Base Pack: Send Trap', and 'SNMP Trap [1]: SL1 Event Trap'.
- Aligned Actions:** A list of actions including '1. Run Ping [113]: Run Ping: Default Options' and '2. Snippet [5]: Datacenter Automation: Format Output as'.

At the bottom of the editor, there are 'Save' and 'Save As' buttons.

3. Complete the following fields as needed:

- **Policy Name.** Type a new name for the automation policy to avoid overwriting the default policy.
- **Policy Type.** Select whether the automation policy will match events that are active, match when events are cleared, or run on a scheduled basis. Typically, you would select *Active Events* in this field.
- **Policy State.** Specifies whether the policy will be evaluated against the events in the system. If you want this policy to begin matching events immediately, select *Enabled*.
- **Policy Priority.** Specifies whether the policy is high-priority or default priority. These options determine how the policy is queued.

- **Aligned Actions.** This field includes the actions from the Network Connectivity PowerPack. You should see Run Ping, Run Traceroute, Run Nslookup, and Run NMAP actions in this field.

To add an action to the **Aligned Actions** field, select the action in the **Available Actions** field and click the right arrow (>>). To re-order the actions in the **Aligned Actions** field, select an action and use the up arrow or down arrow buttons to change that action's position in the sequence. Select an output format action from the *Datacenter Automation Utilities* PowerPack.



CAUTION: Remember that you must include an output format action (from the *Datacenter Automation Utilities* PowerPack) for this action to produce output.

- **Organization.** Select the organization that will use this policy.
4. Optionally, supply values in the other fields on the **Automation Policy Editor** page to refine when the automation will trigger.
 5. Click **[Save]**.

Removing an Automation Policy from a PowerPack

After you have customized a policy from a *Network Connectivity* PowerPack, you might want to remove that policy from that PowerPack to prevent your changes from being overwritten if you update the PowerPack later. If you have the license key with author's privileges for a PowerPack or if you have owner/administrator privileges with your license key, you can remove content from a PowerPack.

To remove content from a PowerPack:

1. Go to the **PowerPack Manager** page (System > Manage > PowerPacks).
2. Find the *Network Connectivity* PowerPack. Click its wrench icon ()
3. In the **PowerPack Properties** page, in the navigation bar on the left side, click **Run Book Policies**.
4. In the **Embedded Run Book Policies** pane, locate the policy you updated, and click the bomb icon () for that policy. The policy will be removed from the PowerPack and will now appear in the bottom pane.

Customizing Network Connectivity Actions

Overview

This manual describes how to customize the three action types embedded in the Network Connectivity PowerPack to create automation actions to meet your organization's specific requirements.

For more information about creating automation policies using custom action types, see [Creating and Customizing Automation Policies](#).

This chapter covers the following topics:

<i>Creating a Custom Action Policy with Network Connectivity Actions</i>	26
<i>Customizing Ping Actions</i>	27
<i>Custom Ping Action Parameters</i>	27
<i>Custom Ping Action Examples</i>	28
<i>Customizing Traceroute Actions</i>	30
<i>Custom Traceroute Action Parameters</i>	31
<i>Custom Traceroute Action Examples</i>	31
<i>Customizing NSLOOKUP Actions</i>	32
<i>Custom NSLOOKUP Action Parameters</i>	32
<i>Custom NSLOOKUP Action Examples</i>	33
<i>Customizing NMAP Actions</i>	34
<i>Custom NMAP Action Parameters</i>	34
<i>Custom NMAP Action Examples</i>	35

Creating a Custom Action Policy with Network Connectivity Actions

You can use one of the Action Types included with the Network Connectivity PowerPack to create custom actions that you can then use to build custom automation policies.

To create an action policy:

1. Navigate to the **Action Policy Manager** page (Registry > Run Book > Actions).
2. In the **Action Policy Manager** page, click the **[Create]** button.
3. The **Action Policy Editor** modal appears.

The screenshot shows the 'Action Editor' modal window. The title bar reads 'Action Editor' with a close button. The main header is 'Policy Editor | Creating New Action' with a 'Reset' button. The form contains the following fields and options:

- Action Name:** An empty text input field.
- Action State:** A dropdown menu currently set to '[Enabled]'.
- Description:** An empty text input field.
- Organization:** A dropdown menu currently set to '[System]'.
- Action Type:** A dropdown menu with a list of options including 'Send an Email Notification', 'Send an Email Notification', 'Send an SNMP Trap', 'Create a New Ticket', 'Send an SNMP Set', 'Run a Snippet', 'Execute an SQL Query', 'Update an Existing Ticket', 'Send an AWS SNS message', 'Execute Commands via SSH (1.0)', 'Execute Remote PowerShell Request (1.0)', 'Get VMware Diagnostic Logs (1.0)', 'Run NMAP (1.0)', 'Run Nslookup (0.9)', 'Run Nslookup (1.0)', 'Run Ping (0.9)', 'Run Ping (1.0)', 'Run Traceroute (0.9)', 'Run Traceroute (1.0)', 'ServiceNow: Create, Update, Clear Incident (1.0)', and 'Update PowerPack Automation Policies (1.0)'. 'Run Ping (1.0)' is currently selected.
- Email Subject:** A text input field containing the placeholder '%S Event: %M'.
- Email:** A text area containing a list of variables: 'Severity: %S', 'First Occurred: %D', 'Last Occurred: %d', 'Occurrences: %c', 'Source: %Z', 'Organization: %O', and 'Device: %X'.
- Available Emails:** A list of email addresses with a scroll bar, including 'em7admin: admin@sciencelogic.com', 'JColtrane: jude.evans-mccarthy@sciencelogic.com', 'JEvans: jude.evans-mccarthy@sciencelogic.com', 'kgibson: kgibson@sciencelogic.com', 'mjasper: mjasper@sciencelogic.com', '[Armstrong, Louis]: jude.evans-mccarthy@sciencelogic.com', '[Cole, J]: shannon.meehan@sciencelogic.com', and '[Davis, Miles]: jude.evans-mccarthy@sciencelogic.com'.

A 'Save' button is located at the bottom center of the modal.

4. In the **Action Policy Editor** page, supply a value in each field.
 - **Action Name.** Specify the name for the action policy.
 - **Action State.** Specifies whether the policy can be executed by an automation policy (enabled) or cannot be executed (disabled).
 - **Description.** Allows you to enter a detailed description of the action.
 - **Organization.** Organization to associate with the action policy.
 - **Action Type.** Type of action that will be executed. Your choices are:

- Run Ping
- Run Traceroute
- Run Nslookup
- Run NMAP
- **Execution Environment.** Select from the list of available Execution Environments. The default execution environment is *System*.
- **Action Run Context.** Select *Database* or *Collector* as the context in which the action policy will run.
- **Input Parameters.** A JSON structure that specifies each input parameter. Each parameter definition includes its name, data type, and whether the input is optional or required for this Custom Action Type.

NOTE: Input parameters must be defined as a JSON structure, even if only one parameter is defined.

6. Click **[Save]**. If you are modifying an existing action policy, click **[Save As]**. Supply a new value in the **Action Name** field, and save the current action policy, including any edits, as a new policy.

Customizing Ping Actions

The Network Connectivity PowerPack includes two automation actions that execute a Ping or Ping6 command. You can specify the host and the options in a JSON structure that you enter in the **Input Parameters** field in the **Action Policy Editor** modal.

The following automation actions that use the "Run Ping" action type are included in the Network Connectivity PowerPack.

Action Name	Description	host	options	ipv6
Run Ping: Default Options	Runs a ping with default options	Default is %a (IP address of current device)	Default is None (empty string)	false
Run Ping6: Default Options	Runs a ping6 with default options	Default is %a (IP address of current device)	Default is None (empty string)	true

TIP: For more information about substitution variables, see [Appendix A](#).

Custom Ping Action Parameters

The Ping actions accepts the following parameters in JSON:

Parameter	Input type	Description
host	string	The hostname or IP address to include in the ping command. You can also use the substitution variable "%a" to specify the IP address of the current device.
options	string	The options string to include in the command. Escape characters are not supported. You can include any of the options supported by the ping command-line utility in this field. If you do not include the "-c" or "-w" options in this field, the ping command will automatically include the option "-c 5", meaning that Ping will send five ECHO_REQUEST packets.
ipv6	boolean	(optional) If the ipv6 option is true, the ping6 command will be executed. If the ipv6 option is false, the ping command will be executed.

NOTE: The pipe (|) and semi-colon (;) characters are not permitted as input to the "host" and "options" parameters.

Using Substitution Values. The host and options inputs can contain substitution values that match the keys in EM7_VALUES. For example, to run a ping against the IP address of the device that triggered the event, you can specify "%a" in the "host" parameter.

TIP: For more information about substitution variables, see [Appendix A](#).

Custom Ping Action Examples

IPv4. If the options parameter contains either "-c" or "-w" as a sub-string, and the ipv6 parameter is false or not supplied, the ping command string is built in the following format:

```
ping [options input] [host input]
```

For example, for the following settings:

- **host.** 192.168.1.1
- **options.** -c 10

The equivalent ping command string would be: `ping -c 10 192.168.1.1`

The equivalent JSON structure would be:

```
{
  "host": "192.168.1.1"
  "options": "-c 10"
  "ipv6": false
}
```

IPv6. If the options parameter contains either "-c" or "-w" as a sub-string and the ipv6 parameter is true, a ping command string is built in the following format:

```
ping6 [options input] [host input]
```

For example, for the following settings:

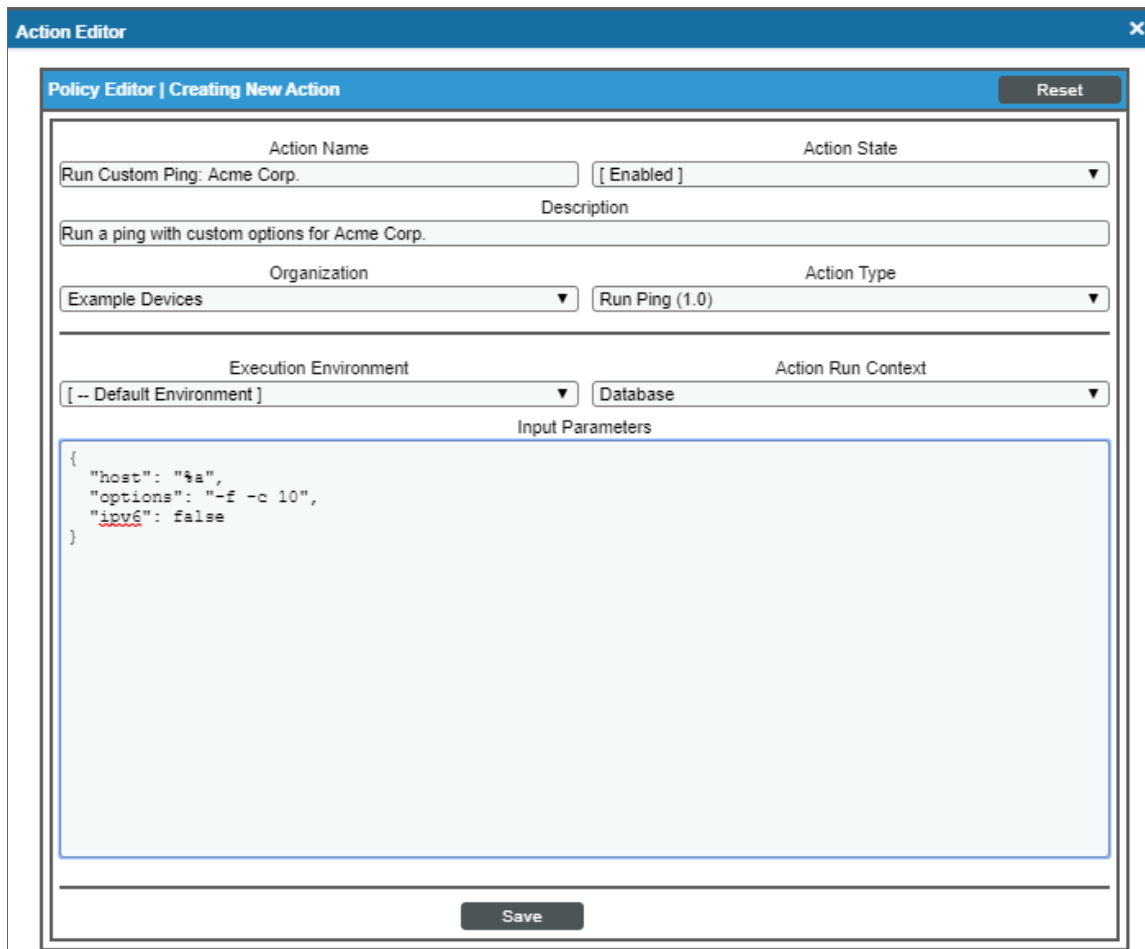
- **host.** 192.168.1.1
- **options.** -c 10

The equivalent ping command string would be: `ping6 -c 10 192.168.1.1.`

The equivalent JSON structure would be:

```
{  
  "host": "192.168.1.1"  
  "options": "-c 10"  
  "ipv6": true  
}
```

The following figure shows a custom ping action for a fictitious company. This custom action is designed to ping IPv4 addresses 10 times without fragmenting the ICMP packets. The action will use the IP address of the current device as the IP address argument.



For a description of all options that are available in Automation Policies, see the *Run Book Automation* manual.

Customizing Traceroute Actions

The Network Connectivity PowerPack includes two automation actions that execute a traceroute command. You can specify the host and the options in a JSON structure (name:value pairs) that you enter in the **Input Parameters** field in the **Action Policy Editor** modal.

The following automation actions that use the "Run Traceroute" custom action type are included in the *Network ConnectivityPowerPack*.

Action Name	Description	host	options	packet_length
Run Traceroute: Default Options	Runs an IPv4 traceroute with default options	Default value is %a (IP address of the current device)	Default value is None (empty string)	Default value is 0
Run IPv6 Traceroute: Default Options	Runs an IPv6 traceroute with all other options as default	Default value is %a (IP address of the current device)	Default value is -6	Default value is 0

TIP: For more information about substitution variables, see [Appendix A](#).

Custom Traceroute Action Parameters

The custom Traceroute action type accepts the following parameters:

Parameter	Input type	Description
host	string	The hostname or IP address to include in the traceroute command. You can also use the substitution variable "%a" to specify the IP address of the current device.
options	string	The options string to include in the command. You can include any of the options supported by the traceroute command-line utility, except for "-T" and "-I", in this field.
packet_length	integer	The packet length to include in the traceroute command. To use the default packet length, use "0".

NOTE: The pipe (|) and semi-colon (;) characters are not permitted as input to the "host" and "options" parameters.

Using Substitution Values. The host and options inputs can contain substitution values that match the keys in EM7_VALUES. For example, to run a traceroute against the IP address of the device that triggered the event, you can specify "%a" in the "host" parameter.

TIP: For more information about substitution variables, see [Appendix A](#).

Custom Traceroute Action Examples

For the following settings, the equivalent traceroute command string would be: `traceroute -T 192.168.1.1`

- **host.** 192.168.1.1
- **options.** -T
- **packet_length.** 0

The equivalent JSON structure would be:

```
{
  "host": "192.168.1.1"
  "options": "-t"
  "packet_length": 0
}
```

For the following settings, the equivalent traceroute command string would be: `traceroute 192.168.1.2 100`

- **host.** 192.168.1.2
- **options.** An empty string
- **packet_length.** 100

The equivalent JSON structure would be:

```
{
  "host": "192.168.1.2"
  "options": ""
  "packet_length": 100
}
```

Customizing NSLOOKUP Actions

The Network Connectivity PowerPack includes an automation action that executes an NSLOOKUP command. You can specify the host and the options in a JSON structure (name:value pairs) that you enter in the **Input Parameters** field in the **Action Policy Editor** modal

The following automation actions that use the Run Nslookup custom action type are included in the *Network ConnectivityPowerPack*.

Action Name	Description	host	options	nameserver
Run Nslookup: Default Options	Runs an nslookup with default options	Default value is %a (IP address of the current device)	Default value is None (empty string)	Default value is None (empty string)

TIP: For more information about substitution variables, see [Appendix A](#).

Custom NSLOOKUP Action Parameters

The custom NSLOOKUP action type accepts the following parameters:

Parameter	Input type	Description
host	string	The hostname or IP address to include in the NSLOOKUP command. You can also use the substitution variable "%a" to specify the IP address of the current device.
nameserver	string	The IP address or hostname of the nameserver to include in the NSLOOKUP command
options	string	The options string to include in the command. You can include any of the options supported by the NSLOOKUP command-line utility in this field.

NOTE: The pipe (|) and semi-colon (;) characters are not permitted as input parameters.

Using Substitution Values. The host and options inputs can contain substitution values that match the keys in EM7_VALUES. For example, to run a traceroute against the IP address of the device that triggered the event, you can specify "%a" in the "host" parameter.

TIP: For more information about substitution variables, see [Appendix A](#).

Custom NSLOOKUP Action Examples

For example, for the following settings, the equivalent NSLOOKUP command string would be:

```
nslookup -timeout=10 192.168.1.1
```

- **host.** 192.168.1.1
- **options.** -timeout=10
- **nameserver.** An empty string

The equivalent JSON structure would be:

```
{
  "host": "192.168.1.1"
  "nameserver": ""
  "options": "-timeout=10"
}
```

For the following settings, the equivalent NSLOOKUP command string would be:

```
nslookup 192.168.1.2 10.64.148.32
```

- **host.** 192.168.1.2
- **options.** An empty string
- **nameserver.** 10.64.148.32

The equivalent JSON structure would be:

```
{
  "host": "192.168.1.2"
  "nameserver": "10.64.148.32"
  "options": ""
}
```

Customizing NMAP Actions

The Network Connectivity PowerPack includes three automation actions that execute an NMAP command. You can specify the host and the options in a JSON structure that you enter in the **Input Parameters** field in the **Action Policy Editor** modal.

The following automation actions that use the "Run NMAP" action type are included in the Network Connectivity PowerPack.

Action Name	Description	host	options
Run NMAP: Common Port List	Runs an NMAP command using a list of common ports.	Default is %a (IP address of current device)	Default ports are 21, 22, 25, 53, 80, 443, 5985, and 5986.
Run NMAP: Monitored Ports	Runs an NMAP command on the ports that are currently monitored on the device.	Default is %a (IP address of current device)	Default is %_monitored_ports_%
Run NMAP: Single Port from Event	Runs an NMAP command on the port provided in the event sub-entity.	Default is %a (IP address of current device)	Default is %Y

TIP: For more information about substitution variables, see [Appendix A](#).

Custom NMAP Action Parameters

Custom NMAP action types accept the following parameters:

Parameter	Input type	Description
host	string	The hostname or IP address to include in the NMAP command. You can use the substitution variable "%a" to specify the IP address of the current device.
options	string	The options string to include in the command. See the parameters for specific NMAP actions earlier in this section.

NOTE: The pipe (|) and semi-colon (;) characters are not permitted as input to the "host" and "options" parameters.

Using Substitution Values. The host and options inputs can contain substitution values that match the keys in EM7_VALUES.

The special %_monitored_ports_% substitution variable is supported for the "Run NMAP" action type. This variable replaces a comma-separated list of ports from the monitoring policies aligned to the triggering device.

TIP: For more information about substitution variables, see [Appendix A](#).

Custom NMAP Action Examples

For example, for the following settings, the equivalent NMAP command string would be:

```
nmap -p 22 192.168.1.1
```

- **host.** 192.168.1.1
- **options.** -p 22

The equivalent JSON structure would be:

```
{  
  "host": "192.168.1.1"  
  "options": "-p 22"  
}
```

Suppose you want to scan a range of ports. In this example, we're scanning the ports from 1 to 100. For the following settings, the equivalent NMAP command string would be:

```
nmap -p 1-100 192.168.1.1
```

- **host.** 192.168.1.1
- **options.** -p 1-100

The equivalent JSON structure would be:



```
{  
  "host": "192.168.1.2"  
  "options": "-p 1-100"  
}
```

Run Book Variables

Overview

This appendix defines the different variables you can use when creating an action policy.

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:

Run Book Variables	36
--	----

Run Book Variables

You can include variables when creating an action policy. These variables are listed in the table below.

- In an action policy of type **Send an Email Notification**, you can include one or more of these variables in the fields **Email Subject** and **Email Body**.
- In an action policy of type **Send an SNMP Trap**, you can include one or more of these variables in the **Trap OID** field, **Varbind OID** field, and the **Varbind Value** field.
- In an action policy of type **Create a New Ticket**, you can include one or more of these variables in the **Description** field or the **Note** field of the related Ticket Template.
- In an action policy of type **Send an SNMP Set**, you can include one or more of these variables in the **SNMP OID** field and the **SNMP Value** field.
- In an action policy of type **Run A Snippet**, you can access variables from the global dictionary **EM7_VALUES**.

- In a policy of type **Execute an SQL Query**, you can include one or more of these variables in the **SQL Query** field.

Variable	Source	Description
%A	Account	Username
%N	Action	Automation action name
%g	Asset	Asset serial
%h	Asset	Device ID associated with the asset
%i (lowercase "eye")	Asset	Asset Location
%k	Asset	Asset Room
%K	Asset	Asset Floor
%P	Asset	Asset plate
%p	Asset	Asset panel
%q	Asset	Asset zone
%Q	Asset	Asset punch
%U	Asset	Asset rack
%u	Asset	Asset shelf
%v	Asset	Asset tag
%w	Asset	Asset model
%W	Asset	Asset make
%m	Automation	Automation policy note
%n	Automation	Automation policy name
%F	Dynamic Alert	Alert ID for a Dynamic Application Alert
%l (uppercase "eye")	Dynamic Alert	For events with a source of "dynamic", this variable contains the index value from SNMP. For events with a source of "syslog" or "trap", this variable contains the value that matches the Identifier Pattern field in the event definition.
%T	Dynamic Alert	Value returned by the Threshold function in a Dynamic Application Alert.

Variable	Source	Description
%V	Dynamic Alert	Value returned by the Result function in a Dynamic Application Alert.
%a	Entity	IP address
_%category_id	Entity	Device category ID associated with the entity in the event.
_%category_name	Entity	Device category name associated with the entity in the event.
_%class_id	Entity	Device class ID associated with the entity in the event.
_%class_name	Entity	Device class name associated with the entity in the event.
_%parent_id	Entity	For component devices, the device ID of the parent device.
_%parent_name	Entity	For component devices, the name of the parent device.
_%root_id	Entity	For component devices, the device ID of the root device.
_%root_name	Entity	For component devices, the name of the root device.
%1 (one)	Event	Entity type. Possible values are: <ul style="list-style-type: none"> • 0. Organization • 1. Device • 2. Asset • 4. IP Network • 5. Interface • 6. Vendor • 7. Account • 8. Virtual Interface • 9. Device Group • 10. IT Service • 11. Ticket

Variable	Source	Description
%2	Event	<p>Sub-entity type.</p> <p>Possible values for organizations are:</p> <ul style="list-style-type: none"> • 9. News feed <p>Possible values for devices are:</p> <ul style="list-style-type: none"> • 1. CPU • 2. Disk • 3. File System • 4. Memory • 5. Swap • 6. Component • 7. Interface • 9. Process • 10. Port • 11. Service • 12. Content • 13. Email
%4	Event	Text string of the user name that cleared the event.
%5	Event	Timestamp of when event was deleted.
%6	Event	Timestamp for event becoming active.
%7	Event	<p>Event severity (1-5), for compatibility with previous versions of SL1. 1=critical, 2=major, 3=minor, 4=notify, 5=healthy.</p> <div style="border: 1px solid black; padding: 5px; margin-top: 10px;"> <p>NOTE: When referring to an event, %7 represents severity (for previous versions of SL1). When referring to a ticket, %7 represents the subject line of an email used to create a ticket.</p> </div>
%c	Event	Event counter
%d	Event	Timestamp of last event occurrence.
%D	Event	Timestamp of first event occurrence.
%e	Event	Event ID

Variable	Source	Description
%H	Event	URL link to event
%M	Event	Event message
%s	Event	severity (0 - 4). 0=healthy, 1=notify, 2=minor, 3=major, 4=critical.
%S	Event	Severity (HEALTHY - CRITICAL)
_%user_note	Event	Current note about the event that is displayed on the Events page.
%x	Event	Entity ID
%X	Event	Entity name
%y	Event	Sub-entity ID
%Y	Event	Sub-entity name
%Z	Event	Event source (Syslog - Group)
%z	Event	Event source (1 - 8)
_%ext_ticket_ref	Event	For events associated with an external Ticket ID, this variable contains the external Ticket ID.
%3	Event Policy	Event policy ID
%E	Event Policy	External ID from event policy
%f	Event Policy	Specifies whether event is stateful, that is, has an associated event that will clear the current event. 1 (one)=stateful; 0 (zero)=not stateful.
%G	Event Policy	Event Category
%R	Event Policy	Event policy cause/action text
_%event_policy_name	Event Policy	Name of the event policy that triggered the event.
%B	Organization	Organization billing ID
%b	Organization	Impacted organization
%C	Organization	Organization CRM ID
%o (lowercase "oh")	Organization	Organization ID
%O (uppercase "oh")	Organization	Organization name

Variable	Source	Description
%r	System	Unique ID / name for the current SL1 system
%7	Ticket	<p>Subject of email used to create a ticket. If you specify this variable in a ticket template, SL1 will use the subject line of the email in the ticket description or note text when SL1 creates the ticket.</p> <div style="border: 1px solid black; padding: 5px; margin-top: 10px;"> <p>NOTE: When referring to a ticket, %7 represents the subject line of an Email used to create a ticket. When referring to an event, %7 represents severity (for previous versions of SL1).</p> </div>
%t	Ticket	Ticket ID

© 2003 - 2020, 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



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

International: +1-703-354-1010