# Microsoft Hyper-V Automations PowerPack

Version 102

# Table of Contents

# Chapter 1

# Microsoft Hyper-V Automations

## Overview

This manual describes how to use the run book automation policies, run book actions, and custom action types found in the "Microsoft Hyper-V Automations" PowerPack. Installation of the "Windows PowerShell Automations" PowerPack is required before using the "Microsoft Hyper-V Automations" PowerPack.

**IMPORTANT:** Version 102 and later of the "Microsoft Hyper-V Automations" PowerPack includes updates for compatibility with Python 3. As a result, this version of the PowerPack runs only in a Python 3 execution environment.

This chapter covers the following topics:

# What is the Microsoft Hyper-V Automations PowerPack?

The "Microsoft Hyper-V Automations" PowerPack includes:

- A set of run book actions that run diagnostic commands on Hyper-V systems via PowerShell
- A set of run book automation policies that tie events from monitoring PowerPacks to the automation actions
- A dynamic device group for Hyper-V devices that is used to scope the automation policies

The run book actions in this PowerPack are executed on the Skylar One All-In-One Appliance or Data Collector.

In addition to using the standard content, you can use the content in the "Microsoft Hyper-V Automations" PowerPack to create your own automation policies that include the pre-defined actions that run different sets of diagnostic commands.

The "Microsoft Hyper-V Automations" PowerPack uses the supplied "Execute PowerShell Request" custom action type included with the "Windows PowerShell Automations" PowerPack.

# Prerequisites

Before installing the "Microsoft Hyper-V Automations" PowerPack, you must perform the following actions:

- Install version 202 or later of the "Datacenter Automation Utilities" PowerPack.
- Install the "Microsoft: Hyper-V Server" PowerPack version 100 or later and configure it to monitor your Hyper-V devices.
- Install version 105 or later of the "Windows PowerShell Automations" PowerPack .

# Upgrading from a Previous Version of This PowerPack

If you are upgrading from a previous version of this PowerPack, you will need to execute the following SQL query on your Skylar One system to ensure that any run book actions on the Skylar One system from the previous versions of this PowerPack can receive updates after upgrading to this release.

To upgrade from a previous release:

1. In Skylar One, go to the Database Tool page (System > Tools > DB Tool).
2. Execute the following queries:

```
UPDATE master.policies_actions SET action_guid
="DA83E56F17CD08430B91F64C5AAA345D" WHERE action_guid IN
("7EEE6230C4050096564969938AD9F07E")
```

```
UPDATE master.policies_actions SET action_guid
="1C5EA01D9D944FEE3C7C2E196C5D162F" WHERE action_guid IN
("86324AFDAE1081C3A13ECEC93F6197FA");
```

# Installing the PowerPack

Before completing the steps in this manual, you must import and install the latest version of the "Microsoft Hyper-V Automations" 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 section on *Global Settings*.

**NOTE:** For details on upgrading Skylar One, see the relevant *Skylar One Platform Release Notes*.

To download and install the PowerPack:

1. Search for and download the PowerPack from the **PowerPacks** page on the ScienceLogic Support Site (Skylar One > PowerPacks).
2. In Skylar One, 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*.

## Creating a Credential for Hyper-V Automation

The "Microsoft Hyper-V Automations" PowerPack uses the same credential that you created for the "Windows PowerShell Automations" PowerPack. Refer to the *Creating a Credential* section in the ***PowerShell Automations*** manual for more information.

> **NOTE:** If you have the "Microsoft: Windows Server" PowerPack installed and configured, you may skip this section.

For more information about configuring credentials in Skylar One, see the ***Discovery and Credentials*** manual.

# Chapter 2

# Configuring Microsoft Hyper-V Automations

## Overview

This chapter describes how to create automation policies using the run book actions in Microsoft Automation PowerPacks.

This chapter covers the following topics:

# Microsoft Hyper-V Automation Policies

The "Microsoft Hyper-V Automations" PowerPack includes the following run book automation policies:

- Hyper-V: CPU & Memory Diagnostic Commands
- Hyper-V: Disk & Storage Diagnostic Commands
- Hyper-V: Guests Below Threshold Diagnostic Commands
- Hyper-V: Guests Below Threshold Diagnostic Commands

Each policy triggers three run book actions that collect diagnostic data within a PowerShell session, and an action that formats the output in HTML. All of the run book actions use the same custom action type, "Execute PowerShell Request", which is supplied in the "Windows PowerShell Automations" PowerPack.

All of the standard automation policies are tied to included ScienceLogic Skylar One events generated by the Dynamic Applications from the "Microsoft: Hyper-V" Server PowerPack.

Several of the run book actions use the substitution character feature of the “Execute PowerShell Request” custom action type. If an event variable is included in a command (such as "%Y" for the sub-entity name), the custom action type automatically replaces that variable with the value from the triggering event.

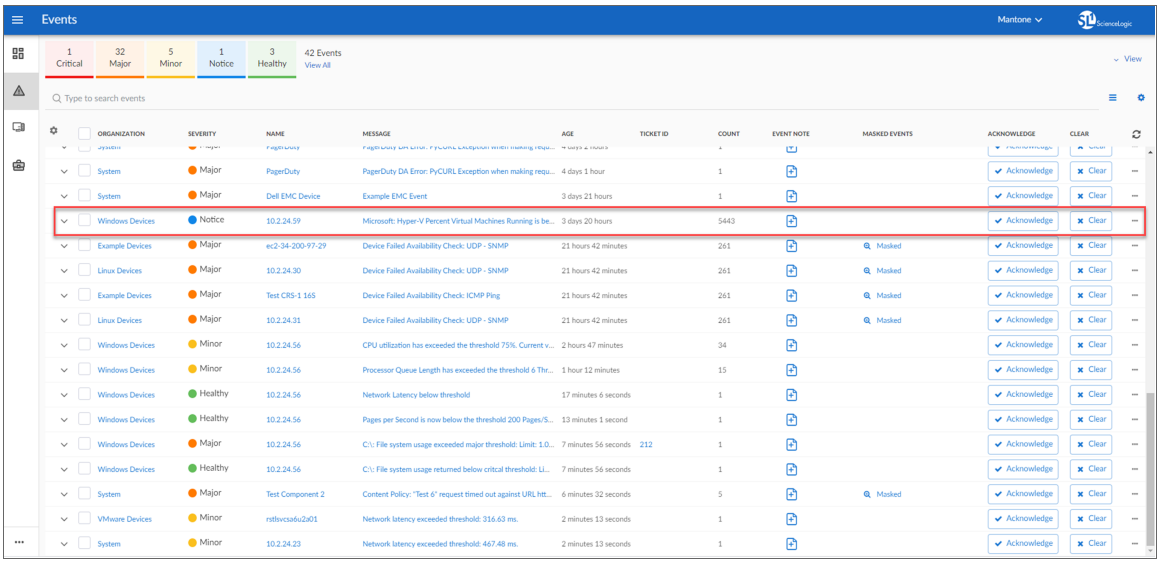
The following table shows the standard automation policies, their aligned events, and the run book actions that run in response to the events.

**NOTE:** The aligned events are included as part of the "Microsoft: Hyper-V Server" PowerPack and are not installed with the Skylar One platform. You must install the "Microsoft: Hyper-V Server" PowerPack to obtain these events.

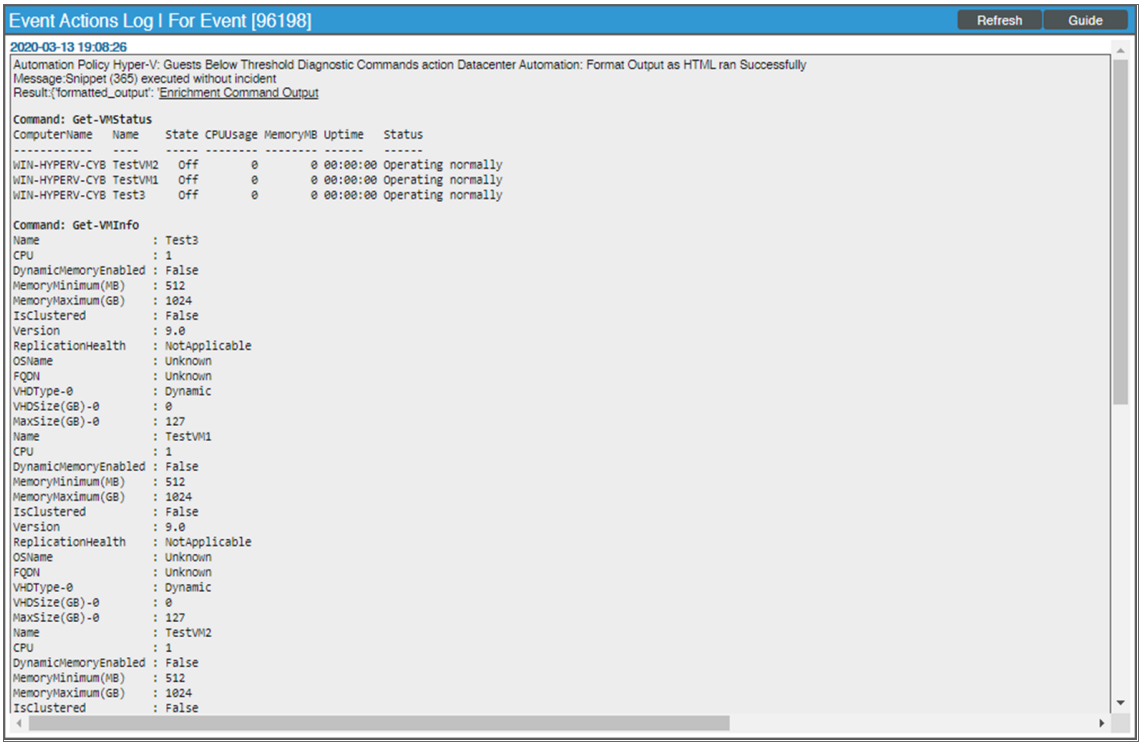
| Automation Policy Name | Aligned Events | Run Book Actions |
|---|---|---|
| Hyper-V: CPU & Memory Diagnostic Commands | • Microsoft: Windows CPU Utilization has exceeded the threshold<br>• Microsoft: Windows Processor Queue Length exceeded the threshold<br>• Microsoft: Windows Available Memory below threshold<br>• Microsoft: Windows Pages per Second has exceeded threshold<br>• Microsoft: Windows Paging File has exceeded threshold | • Automation Utilities: Calculate Memory Size for Each action<br>• Hyper-V Guest Status Diagnostic Commands<br>• Hyper-V Log Collection<br>• Datacenter Automation: Format Output as HTML |
| Hyper-V: Disk & Storage Diagnostic Commands | • Microsoft: Windows Disk Transfer Time (Physical Disk) exceeded threshold<br>• Microsoft: Windows % Disk Time (Logical Disk) exceeded threshold | • Automation Utilities: Calculate Memory Size for Each action |

| Automation Policy Name | Aligned Events | Run Book Actions |
|---|---|---|
| | • Microsoft: Windows % Disk Time (Physical Disk) exceeded threshold<br>• Microsoft: Windows Current Disk Queue Length (Physical Disk) exceeded threshold<br>• Poller: File system usage exceeded (major) threshold<br>• Poller: File system usage exceeded (critical) threshold | • Hyper-V Guest Replication Diagnostic Command<br>• Hyper-V Guest Status Diagnostic Commands<br>• Hyper-V Guest Storage Diagnostic Commands<br>• Datacenter Automation: Format Output as HTML |
| Hyper-V: Guests Below Threshold Diagnostic Commands | • Microsoft: Hyper-V Percent VMs Running below threshold | • Automation Utilities: Calculate Memory Size for Each Action<br>• Hyper-V Guest Replication Diagnostic Command<br>• Hyper-V Guest Status Diagnostic Commands<br>• Hyper-V Guest Storage and Replication Diagnostic Commands<br>• Hyper-V Log Collection<br>• Datacenter Automation: Format Output as HTML |
| Hyper-V: Run Time Capacity Diagnostic Commands | • Microsoft: Hyper-V Percent Total Run Time has exceeded major threshold<br>• Microsoft: Hyper-V Percent Total Run Time has exceeded minor threshold | • Automation Utilities: Calculate Memory Size for Each Action<br>• Hyper-V Guest Status Diagnostic Commands<br>• Hyper-V Allocation Diagnostic Commands<br>• Hyper-V Log Collection<br>• Datacenter Automation: Format Output as HTML |

The following figure shows a memory event with a classification of "Major" appears on the **Events** page. Click the **[Actions]** button ( ... ) for an event, and select *View Automation Actions* to see the run book actions triggered by the events.




The results shown for this event, in the **Event Actions Log**, include the automation policy that ran (shown at the top of the following figure), along with the run book actions (commands) that ran. Results for each command are also displayed. The following figure shows an example of this HTML output.




To learn more about which commands are executed by default for a given run book action, see *Customizing Run Book Actions*.

TIP: Although you can edit the automation policies described in this section, it is a best practice to use "Save As" to create a new run book action, rather than to customize the standard automation policies.

## Prerequisites

Before you create an automation policy using the run book actions in this PowerPack, you must determine:

- Which set of commands you want to run on a monitored device when an event occurs. There are ten run book actions in the PowerPack that run the "Execute PowerShell Request" action type with different commands. You can also create your own run book actions using the custom action type supplied in the PowerPack.
- What event criteria you want to use to determine when the run book 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 run book actions in this PowerPack:

1. Go to the **Automation Policy Manager** page (Registry > Run Book > Automation).
2. Click **[Create]**. The **Automation Policy Editor** page appears.
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* without specifying individual devices to align to.

- ***Aligned Action***s. This field includes the actions from the PowerPack. 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.

> **NOTE:** You must have at least two ***Aligned Actions***: one that runs the run book action and one that provides the output format. The actions providing the output formats are contained in the "Datacenter Automation Utilities" PowerPack, which is a prerequisite for running automations in this PowerPack.

> **NOTE:** If you are selecting multiple collection actions that use the "Execute PowerShell Request" action type, you may want to include the “Calculate Memory Size for Each Action” automation action, found in the "Datacenter Automation Utilities" PowerPack, in your automation policy.

4. To align the policy with the *Windows Automation* device group, which is supplied in the PowerPack, do the following:
   a. In the ***Align With*** drop-down menu, select "Device Groups".
   b. In the ***Available Device Groups*** field, select, the "Windows Automation" device group, and click the right arrow (>>).
5. Optionally, supply values in the other fields on this page to refine when the automation will trigger.
6. Click **[Save]** or **[Save As]**. 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. The best practice is to use **[Save As]** option to create a new, renamed automation policy, instead of customizing the standard automation policies.

## Removing an Automation Policy from a PowerPack

After you have customized a policy from a 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 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.

Chapter

3

# Configuring Microsoft Hyper-V Run Book Actions

## Overview

This manual describes how to customize the run book actions embedded in the "Microsoft Hyper-V Automations" PowerPack to create run book actions to meet your organization's specific requirements.

For more information about creating automation policies using custom action types, see *Configuring Microsoft Hyper-V Automations*.

This chapter covers the following topics:

# Creating a Custom Action Policy

You can use the "Execute PowerShell Request" action type included with the "Windows PowerShell Automations" PowerPack to create custom run book actions that you can then use to build custom automation policies.

To create a custom action policy using the "Execute PowerShell Request (2.0)" action type:

1. Navigate to the **Action Policy Manager** page (Registry > Run Book > Actions).
2. In the **Action Policy Manager** page, click the **[Create]** button. The **Action Policy Editor** modal appears.
3. 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. Select the "Execute PowerShell Request (2.0)" action type.
    - ***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. For more information about the available input parameters, see the table in *Creating a Windows PowerShell Run Book Action*. Input parameters must be defined as a JSON structure.
4. 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 Run Book Actions

The "Microsoft Hyper-V Automations" PowerPack includes run book actions that execute the "Execute PowerShell Request" action type to request diagnostic information or remediate an issue. You can specify the commands and the options in the ***Input Parameters*** field in the **Action Policy Editor** modal.

The following run book actions that use the "Execute PowerShell Request" action type are included in the "Microsoft Hyper-V Automations" PowerPack. For more information about input parameter fields, see the table in *Creating a New Microsoft Hyper-V Run Book Action*.

| Action Name | Description |
|---|---|
| Hyper-V Allocation Diagnostic Command | Determines the current resource allocation health of the Hyper-V server or cluster. |

| Action Name | Description |
|---|---|
| Hyper-V Guest Replication Diagnostic Command | Runs a diagnostic command related to Hyper-V guest replication |
| Hyper-V Guest Status Diagnostic Commands | Runs diagnostic commands to collect Hyper-V guest status and configuration information. |
| Hyper-V Guest Storage Diagnostic Commands | Runs diagnostic commands related to Hyper-V Guest storage and replication. |
| Hyper-V Log Collection | Collects the most recent 25 log entries from the Hyper-V logs. |

**TIP:** For more information about substitution variables, see *Appendix A: Run Book Variables*.

## Creating a Windows Hyper-V Run Book Action

You can create a new run book action that runs remote PowerShell requests using the "Execute PowerShell Request" custom action type. To do this, select "Execute PowerShell Request" in the ***Action Type*** drop-down list when you create a new run book action. You can also use the existing run book actions in the PowerPack as a template by using the **[Save As]** option.

The Windows PowerShell run book actions accept the following parameters in JSON:

| Parameter | Input type | Description |
|---|---|---|
| commands | string | Specifies a single command or a list of commands, in JSON format, to execute. You can use substitution variables in the commands. |
| request_key | string | **(Optional field)**<br>**Default value**: empty<br>The unique key for each instance (row) returned by the request. This unique key must be a property name, and the request must include that property (column) and return values from that property name (column).<br>**Example:** Suppose you want to get the ID, number of cores, name, and maximum clock speed of every CPU installed on a Windows system, run the following command, where "DeviceID" is the request key.<br>`Get-WmiObject -Class Win32_Processor -Property DeviceID, NumberOfCores, Name, MaxClockSpeed \| Format-List DeviceID, NumberOfCores, Name, MaxClockSpeed` |
| credential_id | integer | **Default value**: 0<br>Specifies the **credential_id** to use for the connection. |

| Parameter | Input type | Description |
| --- | --- | --- |
| | | • If set to 0 (false), the custom action type will dynamically determine the credential.<br>• If set to an ID number, it maps to the credential ID specified. You can find credential IDs by going to System > Manage > Credentials. |

**Using Substitution Values.** The commands input can contain substitution values that match the keys in EM7_VALUES.

**TIP:** For more information about substitution variables, see *Appendix A: Run Book Variables*.

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

# Appendix

# A

# 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 Skylar One 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:

# 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 |
| %a | Entity | IP address |
| %F | Dynamic Alert | Alert ID for a Dynamic Application Alert |
| %g | Asset | Asset serial |
| %h | Asset | Device ID associated with the asset |
| %I (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. |
| %i (lowercase "eye") | Asset | Asset Location |
| %K | Asset | Asset Floor |
| %k | Asset | Asset Room |
| %L | Dynamic Alert | Value returned by the label variable in a Dynamic Application Alert. |
| %m | Automation | Automation policy note |
| %N | Action | Automation action name |
| %n | Automation | Automation policy name |
| %P | Asset | Asset plate |
| %p | Asset | Asset panel |
| %Q | Asset | Asset punch |
| %q | Asset | Asset zone |

| Variable | Source | Description |
|---|---|---|
| %T | Dynamic Alert | Value returned by the Threshold function in a Dynamic Application Alert. |
| %U | Asset | Asset rack |
| %u | Asset | Asset shelf |
| %v | Asset | Asset tag |
| %W | Asset | Asset make |
| %w | Asset | Asset model |
| %V | Dynamic Alert | Value returned by the Result function in a Dynamic Application Alert. |
| %_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 description 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. |
| %_service_ investigator_url | Entity | The URL of the Business Service Investigator page for the event that triggered the automation (for run book actions that run against events aligned with business services). |
| %1 (one) | Event | Entity type. Possible values are:<br>• 0. Organization<br>• 1. Device<br>• 2. Asset<br>• 4. IP Network<br>• 5. Interface<br>• 6. Vendor<br>• 7. Account<br>• 8. Virtual Interface<br>• 9. Device Group<br>• 10. IT Service<br>• 11. Ticket |
| %2 | Event | Sub-entity type.<br>Possible values for organizations are:<br>• 9. News feed |

| Variable | Source | Description |
|---|---|---|
| | | Possible values for devices are:<br>• 1. CPU<br>• 2. Disk<br>• 3. File System<br>• 4. Memory<br>• 5. Swap<br>• 6. Component<br>• 7. Interface<br>• 9. Process<br>• 10. Port<br>• 11. Service<br>• 12. Content<br>• 13. Email |
| %4 | Event | Text string of the user name that cleared the event. |
| %5 | Event | Date/time when event was deleted. |
| %6 | Event | Date/time when event became active. |
| %7 | Event | Event severity (1-5), for compatibility with previous versions of Skylar One. 1=critical, 2=major, 3=minor, 4=notify, 5=healthy.<br><br>**NOTE**: When referring to an event, %7 represents severity (for previous versions of Skylar One). When referring to a ticket, %7 represents the subject line of an email used to create a ticket. |
| %c | Event | Event counter |
| %d | Event | Date/time when last event occurred. |
| %D | Event | Date/time of first event occurrence. |
| %e | Event | Event ID |
| %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 |

| Variable | Source | Description |
|---|---|---|
| %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 | External 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 |
| %r | System | Unique ID / name for the current Skylar One system |
| %7 | Ticket | Subject of email used to create a ticket. If you specify this variable in a ticket template, Skylar One will use the subject line of the email in the ticket description or note text when Skylar One creates the ticket.<br><br>**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 Skylar One). |
| %t | Ticket | Ticket ID |
| %J | Ticket | Description field from the Skylar One ticket. |

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

International: +1-703-354-1010