Automation Actions

RBA Base Pack: Apply Device Template

The RBA Base Pack: Apply Device Template action applies a device template to the device associated with the event triggering the automation. The device template to be applied is determined at execution time by searching for a device template using the following variables in the snippet code:

- filter_field: This specifies the device template attribute that will be used to match. This can be any field that the /api/device_template REST API resource supports for searching. The commonly used fields are the device template GUID (template_guid) and the device template name (template_name).
- filter_match: This field defines the REST API matching option that will be used. This will typically be set to "eq", where the match_value must be equal to the device template attribute.
- match_value: This field defines the value to search for, e.g. the template GUID or the template name of the template you want to use.

The automation determines the device template by performing a GET to the following REST API URI:

/api/device_template?limit=1&hide_filterinfo=1&filter.{filter_field}.{filter_match}={match_value}

The default values in the automation specify a GUID match for the example device template that is included in the PowerPack (RBA Base Pack Example Template).

If the automation action successfully applies a template, an alert is generated for the triggering device indicating the template was applied.

RBA Base Pack: Apply Device Template to Group

The RBA Base Pack: Apply Device Template to Group action uses an action type of the same name. The action applies a device template to the device group specified in the action parameters. The action type takes the following parameters:

- dev_template_field: This specifies the device template attribute that will be used to match. This can be any field that the /api/device_template REST API resource supports for searching. The commonly used fields are the device template GUID (template_guid) and the device template name (template_name).
- dev_template_match_type: This field defines the REST API matching option that will be used to look up a device template. This will typically be set to "eq", where the match_value must be equal to the device template attribute.
- dev_template_match: This field defines the value to search for when looking up a device template, e.g. the template GUID or the template name of the template you want to use.

- dev_group_field: This specifies the device template attribute that will be used to match. This can be any field that the /api/device_group REST API resource supports for searching. The commonly used field is the device group name (name).
- dev_group_match_type: This field defines the REST API matching option that will be used to look up a device group. This will typically be set to "eq", where the match_value must be equal to the device template attribute.
- dev_group_match: This field defines the value to search for when looking up a device group, e.g. the group name.

The automation determines the device template by performing a GET to the following REST API URI:

/api/device_template?limit=1&hide_filterinfo=1&filter.{dev_template_field}.{dev_template_match_type}={dev_template_match}

The automation determines the device group by performing a GET to the following REST API URI:

/api/device_group?limit=1&hide_filterinfo=1&filter.{dev_group_field}.{dev_group_match_type}={dev_group_match}

Note that when you schedule an automation for a device group, the automation is executed for every device in that device group. If you want to schedule this automation action, you should align it to only a single device, not a device group, to avoid the action repeating multiple times.

The automation will fail if the matching device group does not have Config Policies/Bulk Edit enabled in the Visibility field.

RBA Base Pack: Disable Device

The RBA Base Pack: Disable Device automation action updates the device record of the device associated with the triggering event using the REST API. The collection state of the device is changed to disabled.

An alert is generated for the triggering device indicating the result of the API request.

RBA Base Pack: Get Child Component Devices via GQL

The RBA Base Pack: Get Child Component Devices via GQL automation action requests the device ID, name, and state of the component device children (i.e. direct descendants in a component tree) of the device associated with the triggering event using the GQL API. The result of the request is formatted for display in the Event Action Log.

RBA Base Pack: Get PowerFlow Status

The RBA Base Pack: Get PowerFlow Status automation action gets the status of a PowerFlow system by connecting to the REST API. The URL, username, and password for the PowerFlow system comes from the aligned credential. The PowerPack includes a sample credential that is aligned to this action. The sample credential uses the %D substitution, i.e. the URL is dynamically generated using the IP address of the device associated with the event that triggered the automation.

RBA Base Pack: Modify File System State

The RBA Base Pack: Modify File System State automation action updates the "Hide" setting on one or more file systems discovered for the device associated with the event that triggered the automation. The file system(s) to be updated and the setting to apply are configured using the following variables in the snippet code:

- regex_match: A regular expression string. This regex will be matched against the Description (i.e. mount point) field for all file systems on the device. For all matching file systems, the Hidden setting is updated to the hide_state value.
- hide_state: Must be either 0 (no) or 1 (yes), which specifies the hide state of the filesystem. Hidden filesystems (yes) are not actively monitored by SL1.

RBA Base Pack: Modify Interface Polling by Rule

The RBA Base Pack: Modify Interface Polling by Rule automation action updates the monitoring settings on one or more network interfaces discovered for the device associated with the event that triggered the automation.

The action iterates through the provided interface_rules list and applies the specified settings to the matching interfaces. See the comments in the snippet code for instructions on how to configure interface rules.

RBA Base Pack: SNS Notification

The RBA Base Pack: SNS Notification automation action is an example of how to send an AWS SNS notification in an automation. This action includes body text that demonstrates how to use event variables. To use this action, you will need to configure an AWS credential and align it to the action.

Automation Policies

The following automation policies are included in the PowerPack. If you make changes as described in this section, it is recommended that you also follow the steps in the Removing Content from PowerPacks section:

- **RBA Base Pack: Apply Device Template**: This automation runs the RBA Base Pack: Apply Device Template action when the Device Record Created or Component Device Record Created events are triggered for a device, i.e. the template is applied to every new device in the system. This automation is disabled by default. To use this automation:
    - Update the RBA Base Pack: Apply Device Template action to use the appropriate template or update the example template to meet your needs.
    - Update the RBA Base Pack: Apply Device Template automation policy to enable it and adjust other settings to meet your needs.
- **RBA Base Pack: Disable All New Component Device**: This automation runs the RBA Base Pack: Disable Device action when the Component Device Record Created event is triggered for a device, i.e. all new component devices will be disabled. This automation is disabled by default. To use this automation:
    - Update the RBA Base Pack: Apply Device Template automation policy to enable it and adjust other settings to meet your needs. For example, you might create a device group that dynamically includes devices and scope the automation policy to only apply to devices in that device group.
- **RBA Base Pack: Disable New VMware VMs**: This automation runs the RBA Base Pack: Disable Device action when the Component Device Record Created event is triggered for devices in the VMware Virtual Machine device group, i.e. all new VMware Virtual Machine component devices will be disabled. This automation is disabled by default. To use this automation:
    - Update the RBA Base Pack: Disable New VMware VMs automation policy to enable it and adjust other settings to meet your needs.
- **RBA Base Pack: Modify Interface Collections at Device Discovery**: This automation runs the RBA Base Pack: Modify Interface Polling by Rule action when the Device Record Created event is triggered for a device, i.e. the interface rules in that action are applied to every new non-component device in the system. The automation is enabled by default. To use this automation:
    - Update the RBA Base Pack: Modify Interface Polling by Rule action to specify appropriate interface polling rules.
    - Update the RBA Base Pack: Modify Interface Collections at Device Discovery automation policy to adjust the settings to meet your needs.

- **RBA Base Pack: Update File System State**: This automation runs the RBA Base Pack: Modify File System State action when the Completed File System Inventory event is triggered for a device, i.e. the file system rule in the action is applied every time file system inventory completes. Both the event policy and automation are disabled by default. To use this automation:
    - Update the RBA Base Pack: Modify File System State action to specify an appropriate file system state rule.
    - Update the Poller: Completed File System Inventory event policy to enable the policy.
    - Update the RBA Base Pack: Update File System State automation policy to enable the policy and adjust the other settings to meet your needs.

Removing Content from PowerPacks

If you modify content in a PowerPack, e.g. enable an event policy or automation policy in the RBA Base Pack, it is recommended that you remove the content from the PowerPack so that the change is not reverted when the PowerPack is updated. To remove content from a PowerPack:

1. Go to the System > Manage > PowerPacks page.
2. Click the wrench icon for the PowerPack.
3. In the Editing PowerPack window, select the content type in the left NavBar. For example, if you are removing an automation policy, select "Run Book Policies".
4. In the upper pane, click the remove icon for the content. The content moves to the lower pane.

Other PowerPack Content

The following content is included in the RBA Base Pack to enable functionality in specific automations. Refer to the description of the relevant automation policy or action for how to use this content:

- The Poller: Completed File System Inventory event policy is used by the RBA Base Pack: Update File System State automation policy.
- The RBA Base Pack Example Template device template is used by the RBA Base Pack: Apply Device Template automation action.
- The RBA Base Pack Example Group device group is used by the RBA Base Pack: Apply Device Template to Group automation action.
- The VMware Virtual Machines device group is used by the RBA Base Pack: Disable New VMware VMs automation policy.
- The RBA Base Pack: PowerFlow Status credential is used by the RBA Base Pack: Get PowerFlow Status automation action.

Using the Automation Actions as Examples

The following table indicates which automation actions in the RBA Base Pack can be used as examples of how to do specific functions in a snippet.

Function	Automation Action
Use an aligned credential	RBA Base Pack: Get PowerFlow Status
Connect to an external API	RBA Base Pack: Get PowerFlow Status
Query the SL1 GQL API	RBA Base Pack: Get Child Component Devices via GQL
Debug logging	RBA Base Pack: Get Child Component Devices via GQL
Format Output to be read in the SL1 user interface	RBA Base Pack: Get Child Component Devices via GQL
Perform a GET request to the SL1 REST API	RBA Base Pack: Modify Interface Polling by Rule
Perform a POST request to the SL1 REST API	RBA Base Pack: Disable Device
Generate an alert for a device (appears in the device logs and matches API event policies)	RBA Base Pack: Disable Device
Use the device ID and device name from the EM7_VALUES dictionary	RBA Base Pack: Disable Device
Process the inputs and outputs for an Action Type to re-use code for multiple automations	RBA Base Pack: Apply Device Template to Group (both the automation action and the action type)