

Managing Newly Discovered VMs Using Run Book Automation

Challenges:

- As a service provider:
 - I only want to actively monitor the devices my end customer is paying for
 - I want to ensure the VMs in my shared environment get aligned with the correct organization, so that they appear in my end customer's dashboard
- As an enterprise:
 - I do not want to monitor dev. and test VMs that may generate superfluous events

Solution:

Use Run Book Automation to configure new VMs to meet my business needs.

- Example rules to implement with RBA:
 - Move all VMs on specific host to the “Development” org
 - Disable collection on new VMs with “QA” in the name
 - Assign customer org from lookup in external database

Disabling Collection For New VMs

Use “device record created” events as trigger for RBA
Events added in 7.3.2, disabled by default

	 Component Device Record Created	Internal	Disabled	Yes	Notice
	 Device Record Created	Internal	Disabled	Yes	Notice

 LAB-HYPERV	Device record created (Class: Microsoft I Hyper-V Hypervisor)	
---	---	---

Simple
automation
policy

Runs action:
“Apply template
to device”

when a new
physical or
component
device is created

Automation Policy Editor | Editing Automation Policy [12] Reset

Policy Name	Policy Type	Policy State	Organization
Manage New Device By Template	[Active]	[Enabled]	[System]

Criteria Logic	Match Logic	Match Syntax
[Severity >=]	[Text search]	
[Healthy,]		
[and no time has elapsed,]		
[and event is NOT cleared]		

Repeat Time	Align With
[Only once]	[Devices]

☐ Include events for entities other than devices (organizations, assets, etc.)

Available Devices

Boom Vang

Apple Computer, Inc.: Macintosh: RC-Mac

Rackspace: Cloud Services: Test 3

Virtual Device: Content Verification: _EX_Dev3

Xen Cluster: Xen Cluster: Dracon

Aligned Devices

(All devices)

Available Events

Critical: .Test Logger

Critical: AKCP: AC Voltage sensor detects no current

Critical: AKCP: DC Voltage sensor High Critical

Critical: AKCP: DC Voltage sensor Low Critical

Critical: AKCP: Drv Contact Sensor Low Critical

Aligned Events

Notice: Component Device Record Created

Notice: Device Record Created

Available Actions

Send Email: email to rchart

SNMP Trap: EM7 Event Trap

Snippet: Apply template to device

Snippet: EM7 Ping Snippet

Snippet: VMware: vCloud Extract Target

Aligned Actions

1. Snippet: Apply template to device

Save Save As

Action: Apply Template to Device

- Uses new device details to determine template to apply
 - Apply template based on device name, class, etc.
 - Optionally perform lookup in external business rules database
- Template determines any changes to device, such as:
 - Organization
 - Whether collection is enabled
 - Specific monitors to apply
- Template is applied using API call

Device Templates Used In Automation

Configuration Templates Templates Found [6]						Create	Reset	Guide
	Template Name ▾	ID	Created By	Created On	Last Edited By	Edited On	☒	
1.	PRBA: Data Center Infrastructure	8	rchart	2013-06-16 21:19:52	rchart	2013-06-17 00:36:36	☐	
2.	PRBA: Exhibition Center	6	rchart	2013-06-16 20:43:21	rchart	2013-06-17 00:36:38	☐	
3.	PRBA: QA	7	rchart	2013-06-16 21:19:29	rchart	2013-06-17 00:36:40	☐	
4.	UCS Template	3	em7admin	2013-03-04 17:03:24	em7admin	2013-05-29 22:12:26	☐	
5.	VMware vSphere Template	1	em7admin	2013-03-06 21:44:24	em7admin	2013-06-04 08:22:39	☐	
6.	VMware: vCloud Template	4	em7admin	2013-02-05 19:25:10	em7admin	2013-05-30 01:14:12	☐	

```
template_filter_by_name = 'api/device_template?limit=1&hide_filterinfo=1&filter.template_name='

device_templates = {
    '_EX_': 'RBA: Exhibition Center',
    '_QA_': 'RBA: QA',
    '_DC_': 'RBA: Data Center Infrastructure'
}

device_name = EM7_VALUES['%X']
device_id = EM7_VALUES['%x']

for match_val in device_templates:
    if device_name.find(match_val) != NO_MATCH:

        # Lookup template URI with this name
        template_filter = template_filter_by_name + device_templates[match_val]

        r = apiCall(hostname, username, password, template_filter, timeout=timeout_seconds)
        jtemplate = r['result'].json()[0]

        r = apiCall(hostname, username, password, 'api/device/%s' % device_id, 'POST', jtemplate['URI'],
                  {'content-type': 'application/em7-resource-uri'}, timeout_seconds) ..
```


Example Template

Sets organization to QA and disable collection

Template Name	
RBA: _QA_	
Config	Interface CV Policies Port Policies Svc Policies Proc Policies Dyn Apps
Access & Monitoring	
Device Organization	QA
SNMP Read	.Test App
Availability Protocol	TCP
Latency Protocol	TCP
Avail+Latency Alert	Disabled
Collection	Disabled
Coll. Type	Standard
Critical Ping	Disabled
Event Mask	Disabled
SNMP Write	None
Avail Port	ICMP
Latency Port	ICMP
Collector Grp	CUG
Device Preferences	
Auto-Clear Events	Scan All IPs
Accept All Logs	Dynamic Discovery
Daily Port Scans	Preserve Hostname
Auto-Update	Disable Asset Update

New option on automation policy editor

Typical mode:

- Uncheck for event escalations, ticketing
- Check for business automation

Automation Policy Editor | Creating New Automation Policy Reset

Policy Name	Policy Type	Policy State	Organization
	[Active]	[Enabled]	MI6

Criteria Logic	Match Logic	Match Syntax
[Severity >=]	[Text search]	
[and 5 minutes has elapsed,]		
[and event is NOT cleared]		

Repeat Time	Align With
[Only once]	[Devices]

☒ Trigger on Child Rollup ☐ Include events for entities other than devices (organizations, assets, etc.)

Available Devices	Aligned Devices
MI6 .Content: Cluster: Cluster1 .Content: Datacenter: Datacenter1 .Content: HostSystem: Host1 .Content: HostSystem: Host2	(All devices)

Available Events	Aligned Events
Critical: AKCP: AC Voltage sensor detects no current Critical: AKCP: DC Voltage sensor High Critical Critical: AKCP: DC Voltage sensor Low Critical Critical: AKCP: Dry Contact Sensor Low Critical Critical: AKCP: Smoke Detector Alert!	(All events)

Available Actions	Aligned Actions
Send Email: 14301 (Disabled) SNMP Trap: EM7 Event Trap Snippet: EM7 Ping Snippet Snippet: Windows Restart Service	

Save