



## Red Hat Insights 2020-04

# Monitoring and Reacting to Configuration Changes Using Policies

How to create policies to detect system configuration changes and get notified by email



## Red Hat Insights 2020-04 Monitoring and Reacting to Configuration Changes Using Policies

---

How to create policies to detect system configuration changes and get notified by email

Red Hat Customer Content Services

## Legal Notice

Copyright © 2020 Red Hat, Inc.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at

<http://creativecommons.org/licenses/by-sa/3.0/>

. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, the Red Hat logo, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux<sup>®</sup> is the registered trademark of Linus Torvalds in the United States and other countries.

Java<sup>®</sup> is a registered trademark of Oracle and/or its affiliates.

XFS<sup>®</sup> is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL<sup>®</sup> is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js<sup>®</sup> is an official trademark of Joyent. Red Hat is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack<sup>®</sup> Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

## Abstract

This document provides an overview of the Policies service and demonstrates how to create a policy to detect system configuration changes and get notified by email. Providing Feedback: If you have a suggestion to improve this document or find an error, submit a Bugzilla report at <http://bugzilla.redhat.com> against Cloud Software Services (cloud.redhat.com) for the Policies component.

---

## Table of Contents

<b>CHAPTER 1. OVERVIEW</b> .....	<b>3</b>
<b>CHAPTER 2. USER PREFERENCES</b> .....	<b>4</b>
<b>CHAPTER 3. CREATING A POLICY TO DETECT CONFIGURATION CHANGES AND GET NOTIFIED BY EMAIL</b>	<b>5</b>
3.1. CREATING A POLICY TO ENSURE PUBLIC CLOUD PROVIDERS ARE NOT OVER PROVISIONED	5
3.2. CREATING A POLICY TO DETECT IF SYSTEMS ARE RUNNING AN OUTDATED VERSION OF RHEL (OLDER THAN RHEL 8.1) AND GET NOTIFIED BY EMAIL	6
3.3. CREATING A POLICY TO DETECT A VULNERABLE PACKAGE VERSION BASED ON RECENT CVE AND GET NOTIFIED BY EMAIL	6
<b>CHAPTER 4. REVIEWING AND MANAGING POLICIES</b> .....	<b>8</b>
<b>CHAPTER 5. APPENDIX</b> .....	<b>9</b>
5.1. AVAILABLE FACTS AND THEIR FUNCTIONS	9
5.2. AVAILABLE OPERATORS IN CONDITIONS	10



## CHAPTER 1. OVERVIEW

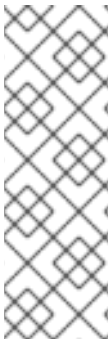
Policies evaluate system configurations in your environment and are processed on reception of uploading an insights-client payload to cloud.redhat.com. If condition(s) in your policy are met, defined action(s) are triggered. Policies are applied to all systems registered in the Insights inventory. Users can create and manage policies using the user interface or via API.

Use policies to assist operational management with simple tasks such as:


- Raising an alert when some conditions are met on system configuration.
- Emailing a team when security packages are out of date on a system.

Using policies to monitor configuration changes in your environment and notifying by email requires:

1. Setting user email preferences (if not already set).
2. Creating a policy to detect configuration changes and selecting email as the trigger action.



### NOTE

- Use the Role Based Access Control (RBAC) capability in <https://cloud.redhat.com> (Settings  > User access) to control user access for Policies.
- See [Role Based Access Control for Red Hat Insights and cloud management services for Red Hat Enterprise Linux](#) for more information about this feature and example use cases.

## CHAPTER 2. USER PREFERENCES

Update your information and set email preferences for cloud.redhat.com services in user preferences.

1. Click the user menu located on the upper-right side, then go to **User preferences → Email preferences**.
2. For Policies, you can subscribe to **Instant notification** emails for each system with triggered policies and/or **Daily digest** (summary) of all systems with triggered policies depending on your email notification preference. You can also select your preference for other <https://cloud.redhat.com> emails you want to receive on this page.



### NOTE

Subscribing to instant notification can result in receiving a lot of emails on large inventories, that is, one email per system checking in.

3. Click **Submit**.



## CHAPTER 3. CREATING A POLICY TO DETECT CONFIGURATION CHANGES AND GET NOTIFIED BY EMAIL

The following workflow examples demonstrate how to create a policy to detect system configuration changes and get notified by email.



### NOTE

If you see a warning message about email alerts not opted in when creating your policy, set your preferences to receive email from your policies as described in the [Chapter 2, User preferences](#) section.

### 3.1. CREATING A POLICY TO ENSURE PUBLIC CLOUD PROVIDERS ARE NOT OVER PROVISIONED

1. In the cloud.redhat.com platform, click [Policies](#) under Red Hat Insights.
2. Click **Create policy**.
3. On the Create Policy page, click **From scratch** or **As a copy of existing Policy** as required. Note that the **As a copy of existing Policy** option will prompt you to select a policy from the list of existing policies to use as a starting point.

**Add Policy** ✕

Policies are processed on reception of system profile messages. If condition(s) are met, defined action(s) are triggered.

1 Create Policy

2 Policy Details

3 Conditions

4 Trigger actions

5 Review and activate

#### Create Policy

Define a new policy:

From scratch

As a copy of existing Policy

▼ Name      Filter by name      🔍

Name	Trigger actions
<input type="radio"/> Test policy that triggers every single time	✉
<input checked="" type="radio"/> Ensure public cloud providers are not over provisioned	🔗
<input type="radio"/> Ensure all systems are updated to later RHEL 8.1 release	🔗 ✉

Next      Cancel

4. Click **Next**.
5. Enter a **Name** and **Description** for the policy.
6. Click **Next**.
7. Enter **Condition**. In this case, enter: `facts.cloud_provider in [alibaba, aws, azure, google] and (facts.number_of_cpus >= 8 or facts.number_of_sockets >=2)`. This condition will detect if an instance running on the said public cloud providers are running with CPU hardware higher than the allowed limit.

8. Click **Validate condition**, then click **Next**.
9. On the Trigger actions page, click **Add trigger actions** and select **Email**.
10. Click **Next**.
11. On the Review and activate page, click the toggle switch to activate the policy and review its details.
12. Click **Finish**.

Your new policy is created. When the policy is evaluated on a system check-in, and if the condition in the policy is met, an email will be sent to all users on the account with access to Policies according to their email preferences.

### 3.2. CREATING A POLICY TO DETECT IF SYSTEMS ARE RUNNING AN OUTDATED VERSION OF RHEL (OLDER THAN RHEL 8.1) AND GET NOTIFIED BY EMAIL

1. In the cloud.redhat.com platform, click [Policies](#) under Red Hat Insights.
2. Click **Create policy**.
3. On the Create Policy page, click **From scratch** or **As a copy of existing Policy** as required. Note that the **As a copy of existing Policy** option will prompt you to select a policy from the list of existing policies to use as a starting point.
4. Click **Next**.
5. Enter a **Name** and **Description** for the policy.
6. Click **Next**.
7. Enter **Condition**. In this case, enter: **facts.os\_release < 8.1** This condition will detect if systems still run an outdated version of our operating system based on RHEL 8.1.
8. Click **Validate condition**, then click **Next**.
9. On the Trigger actions page, click **Add trigger actions** and select **Email**.
10. Click **Next**.
11. On the Review and activate page, click the toggle switch to activate the policy and review its details.
12. Click **Finish**.

Your new policy is created. When the policy is evaluated on a system check-in, and if the condition in the policy is met, an email will be sent to all users on the account with access to Policies according to their email preferences.

### 3.3. CREATING A POLICY TO DETECT A VULNERABLE PACKAGE VERSION BASED ON RECENT CVE AND GET NOTIFIED BY EMAIL

1. In the cloud.redhat.com platform, click [Policies](#) under Red Hat Insights.


2. Click **Create policy**.
3. On the Create Policy page, click **From scratch** or **As a copy of existing Policy** as required. Note that the **As a copy of existing Policy** option will prompt you to select a policy from the list of existing policies to use as a starting point.
4. Click **Next**.
5. Enter a **Name** and **Description** for the policy.
6. Click **Next**.
7. Enter **Condition**. In this case, enter: **facts.installed\_packages contains [openssh-4.5]**. This condition will detect if systems still run a vulnerable version of an **openssh** package based on recent CVE.
8. Click **Validate condition**, then click **Next**.
9. On the Trigger actions page, click **Add trigger actions** and select **Email**.
10. Click **Next**.
11. On the Review and activate page, click the toggle switch to activate the policy and review its details.
12. Click **Finish**.

Your new policy is created. When the policy is evaluated on a system check-in, and if the condition in the policy is met, an email will be sent to all users on the account with access to Policies according to their email preferences.


## CHAPTER 4. REVIEWING AND MANAGING POLICIES

You can review and manage all created policies (enabled and disabled) by clicking [Policies](#) on the left-side menu in Red Hat Insights.

The screenshot shows the Red Hat Insights interface. On the left is a dark sidebar with a navigation menu including Dashboard, Advisor, Vulnerability, Compliance, Patch, Drift, Policies (highlighted), Inventory, Remediations, Subscription Watch, and Documentation. The main area is titled 'Policies' and features a yellow warning banner: 'Enable email alerts. One or more of your policies have an email alert. To receive these emails, opt in to email alerts. Open email preferences'. Below this is a search bar with 'Name' selected, a 'Filter by name' input, and a 'Create policy' button. A table lists three policies with columns for Name, Trigger actions, and Last evaluated. A context menu is open over the table, listing 'Disable policy', 'Edit', 'Duplicate', and 'Delete'.

You can filter the list of policies by name and by active state. You can click the options menu  next to a policy to perform the following operations:

- Enable and disable
- Edit
- Duplicate
- Delete

Additionally, you can perform the following operations in bulk by selecting multiple policies from the list of policies and clicking the options menu  located next to the **Create policy** button at the top:

- Delete policies
- Enable policies
- Disable policies



### NOTE

If you see a warning message about email alerts not opted in, set your preferences to receive email from your policies as described in the [Chapter 2, User preferences](#) section.

## CHAPTER 5. APPENDIX

## 5.1. AVAILABLE FACTS AND THEIR FUNCTIONS

Table 5.1. System Facts and Their Functions

Fact Name	Description	Example Value
<b>arch</b>	System architecture	<b>x86_64</b>
<b>bios_release_date</b>	BIOS release date; typically <b>MM/DD/YYYY</b>	01/01/2011
<b>bios_vendor</b>	BIOS vendor name	LENOVO
<b>bios_version</b>	BIOS version	1.17.0
<b>cloud_provider</b>	Cloud vendor. Values are <b>google</b> , <b>azure</b> , <b>aws</b> , <b>alibaba</b> , or empty	<b>google</b>
<b>cores_per_socket</b>	Number of CPU cores per socket	2
<b>cpu_flags</b>	Category with a list of CPU flags. Each name is the CPU flag (ex: <b>vmx</b> ), and the value is always <b>enabled</b> .	<b>vmx</b> , with a value of <b>enabled</b> .
<b>enabled_services</b>	Category with a list of enabled services. Each name in the category is the service name (ex: <b>crond</b> ), and the value is always <b>enabled</b> .	<b>crond</b> , with a value of <b>enabled</b> .
<b>fqdn</b>	System Fully Qualified Domain Name	<i>system1.example.com</i>
<b>infrastructure_type</b>	System infrastructure; common values are <b>virtual</b> or <b>physical</b>	<b>virtual</b>
<b>infrastructure_vendor</b>	Infrastructure vendor; common values are <b>kvm</b> , <b>vmware</b> , <b>baremetal</b> , etc.	<b>kvm</b>
<b>installed_packages</b>	List of installed RPM packages. This is a category.	<b>bash</b> , with a value of <b>4.2.46-33.el7.x86_64</b> .
<b>installed_services</b>	Category with a list of installed services. Each name in the category is the service name (ex: <b>crond</b> ), and the value is always <b>installed</b> .	<b>crond</b> , with a value of <b>installed</b> .
<b>kernel_modules</b>	List of kernel modules. Each name in the category is the kernel module (ex: <b>nfs</b> ), and the value is <b>enabled</b> .	<b>nfs</b> , with a value of <b>enabled</b> .

Fact Name	Description	Example Value
<b>last_boot_time</b>	The boot time in <b>YYYY-MM-DDTHH:MM:SS</b> format. Informational only; we do not compare boot times across systems.	<b>2019-09-18T16:54:56</b>
<b>network_interfaces</b>	List of facts related to network interfaces.	
	There are six facts for each interface: <b>ipv6_addresses</b> , <b>ipv4_addresses</b> , <b>mac_address</b> , <b>mtu</b> , <b>state</b> and <b>type</b> . The two address fields are comma-separated lists of IP addresses. The <b>state</b> field is either <b>UP</b> or <b>DOWN</b> . The <b>type</b> field is the interface type (ex: <b>ether</b> , <b>loopback</b> , <b>bridge</b> , etc.).	
	Each interface (ex: <b>lo</b> , <b>em1</b> , etc) is prefixed to the fact name. For example, em1's mac address would be the fact named <b>em1.mac_address</b> .	
	Most network interface facts are compared to ensure they are equal across systems. However, <b>ipv4_addresses</b> , <b>ipv6_addresses</b> , and <b>mac_address</b> are checked to ensure they are different across systems. A subexception for <b>lo</b> should always have the same IP and mac address on all systems.	
<b>number_of_cpus</b>	Total number of CPUs	<b>1</b>
<b>number_of_sockets</b>	Total number of sockets	<b>1</b>
<b>os_kernel_version</b>	Kernel version	<b>4.18.0</b>
<b>os_release</b>	Kernel release	<b>8.1</b>
<b>running_processes</b>	List of running processes. The fact name is the name of the process, and the value is the instance count.	<b>crond</b> , with a value of <b>1</b> .
<b>satellite_managed</b>	Boolean field that indicates is a system is registered to a Satellite server.	<b>FALSE</b>
<b>system_memory</b>	Total system memory in human-readable form	<b>3.45 GiB</b>

## 5.2. AVAILABLE OPERATORS IN CONDITIONS

Available Operators	Value
Logical Operators	AND
	OR
Boolean Operators	EQUAL
	NOTEQUAL
Numeric Compare Operators	GT
	GTE
	LT
	LTE
String Compare Operator	CONTAINS
Array Operators	IN
	CONTAINS
Parser Operators	OR
	AND
	NOT
	EQUAL
	NOTEQUAL
	CONTAINS
	NEG