



## Red Hat Insights 1-latest

# Comparing System Configurations and Baselines in Red Hat Insights Inventory

Using drift service comparisons and baselines for system analysis



# Red Hat Insights 1-latest Comparing System Configurations and Baselines in Red Hat Insights Inventory

---

Using drift service comparisons and baselines for system analysis

## Legal Notice

Copyright © 2024 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

Compare system configurations of a system over time, or to other systems and baselines to identify discrepancies in your environment, and perform drift analysis. Red Hat is committed to replacing problematic language in our code, documentation, and web properties. We are beginning with these four terms: master, slave, blacklist, and whitelist. Because of the enormity of this endeavor, these changes will be implemented gradually over several upcoming releases. For more details, see our CTO Chris Wright's message.

## Table of Contents

<b>CHAPTER 1. USING COMPARISONS IN YOUR INSIGHTS FOR RED HAT ENTERPRISE LINUX INVENTORY TO PERFORM DRIFT ANALYSIS</b> .....	<b>4</b>
1.1. USER ACCESS FOR DRIFT SERVICE USERS	4
1.1.1. Drift analysis administrator role	4
1.1.2. Drift viewer role	5
<b>CHAPTER 2. ACCESSING THE DRIFT SERVICE</b> .....	<b>6</b>
<b>CHAPTER 3. USING SYSTEM FACTS TO COMPARE SYSTEMS IN YOUR INSIGHTS INVENTORY</b> .....	<b>8</b>
3.1. CREATING AND ADDING TO COMPARISONS	8
<b>CHAPTER 4. REVIEWING AND MANAGING BASELINES</b> .....	<b>10</b>
4.1. CREATING NEW BASELINES	10
4.1.1. Creating an entirely new baseline	10
4.1.2. Copying an existing baseline to create a new baseline	11
4.1.3. Copying an existing system to create a new baseline	12
4.1.4. Copying a historical system profile to create a new baseline	12
4.2. EDITING OR DELETING BASELINES	13
4.2.1. Editing a baseline	13
4.2.2. Deleting a baseline	14
4.3. COMPARING SYSTEM CONFIGURATIONS TO BASELINES	14
4.4. EXPORTING BASELINES	15
4.4.1. Exporting individual baselines, facts, and categories	15
4.4.2. Exporting all system baselines	15
<b>CHAPTER 5. USING SYSTEM FACTS IN COMPARISONS</b> .....	<b>16</b>
5.1. THE ROLE OF SYSTEM FACTS IN COMPARISONS	16
5.2. HOW FACTS ARE USED IN COMPARISONS	16
5.3. FILTERING SYSTEM FACTS USING THE USER INTERFACE (UI)	16
5.3.1. Selecting systems to compare	16
5.3.2. Filtering by comparison state	17
5.3.3. Filtering by fact name	17
5.3.4. Filtering by fact category	18
5.3.5. Using the UI to sort system facts	18
5.3.6. Sorting facts by comparison state	18
5.3.7. Examples and screenshots	18
5.4. FILTERING SYSTEM FACTS USING URL PARAMETERS	19
5.4.1. Sorting by editing the URL	19
5.5. SORTING SYSTEM FACTS	21
5.6. USING OBFUSCATED VALUES IN COMPARISONS	22
5.7. UNDERSTANDING MULTI-VALUE FACTS	23
5.8. USING MULTI-FACT FILTERING	24
5.9. AVAILABLE FACTS AND THEIR FUNCTIONS	24
<b>CHAPTER 6. COMPARING SYSTEMS AND SYSTEM PROFILES</b> .....	<b>28</b>
6.1. COMPARING SYSTEMS TO A REFERENCE POINT	28
6.1.1. Comparing systems to a single point of reference	28
6.2. USING THE HISTORY OF SYSTEM PROFILES	28
6.3. EXPORTING SYSTEM COMPARISON OUTPUT	28
6.3.1. Exporting system comparison output to a CSV or JSON file	29
<b>CHAPTER 7. ENABLING NOTIFICATIONS AND INTEGRATIONS FOR DRIFT EVENTS</b> .....	<b>30</b>
7.1. ENABLING NOTIFICATIONS AND INTEGRATIONS	30

PROVIDING FEEDBACK ON RED HAT DOCUMENTATION ..... 31



# CHAPTER 1. USING COMPARISONS IN YOUR INSIGHTS FOR RED HAT ENTERPRISE LINUX INVENTORY TO PERFORM DRIFT ANALYSIS

The drift service enables you to compare the system configuration of one system to other systems in your Insights for Red Hat Enterprise Linux inventory. You can also compare system configurations over time to track and analyze changes. In addition, you can set a reference point and compare it to the states of all your systems. You can use any of these comparisons to troubleshoot systems at various points in time.

When comparing configurations across different systems, you can filter profile facts and system tags to highlight configurations that match, or to see where they differ. You can also see which systems have missing or incorrect information.

You can also define and manage baselines. You can use defined baselines in your Insights for Red Hat Enterprise Linux inventory to compare baselines and system configurations. In addition, you can generate a comma-separated values (CSV), or JavaScript Object Notation (JSON) output of the systems and baselines you are comparing.

## 1.1. USER ACCESS FOR DRIFT SERVICE USERS

Before you can access certain features in the Red Hat Insights for Red Hat Enterprise Linux application, you must have the correct permissions, which are granted in [Red Hat Hybrid Cloud Console > the Settings icon \(⚙\) > Identity & Access Management > User Access > Groups](#) . An Organization Administrator or **User Access administrator** must add you as a member to a User Access group with the required roles.

By default, User Access on the Red Hat Hybrid Cloud Console has preconfigured a **Drift analysis administrator** role (all access) and a **Drift viewer** role (read-only access). If your organization determines that the predefined roles provide insufficient access, a **User Access administrator** can configure a custom role to provide the specific permissions that your users require.

The following sections in this chapter describe each of the predefined roles for drift service users.



### IMPORTANT

Changes to User Access must be performed by an Organization Administrator on your Red Hat account, or by an account user who is a member of a User Access group with the **User Access administrator** role.

#### Additional resources

- [User Access Configuration Guide for Role-based Access Control \(RBAC\)](#)

### 1.1.1. Drift analysis administrator role

The **Drift analysis administrator** role is a predefined role in the **Default admin access group**. In its default configuration, members of a group with the **Drift analysis administrator** role can perform any available operation on drift resources.



**NOTE**

If the **Default access group** has previously been modified, these changes will not be applied to it. If you configured groups before October 2022, any user who is not an organization administrator will have the **Drift analysis administrator** role replaced with the **Drift viewer** role.

**1.1.2. Drift viewer role**

The **Drift viewer** role is a predefined role in the **Default admin access group**. All Insights for Red Hat Enterprise Linux users on your account are members of the **Default access group**. The **Drift viewer** role provides read only access. If your organization determines that the default configuration of the **Drift viewer** role is inadequate, a **User Access administrator** can create a custom role with the specific permissions required.

**NOTE**

If the **Default access group** has previously been modified, these changes will not be applied to it. If you configured groups before October 2022, any user who is not an organization administrator will have the **Drift analysis administrator** role replaced with the **Drift viewer** role.

## CHAPTER 2. ACCESSING THE DRIFT SERVICE

The drift service is part of Red Hat Insights for Red Hat Enterprise Linux. Access this service using [Red Hat Hybrid Cloud Console](#).

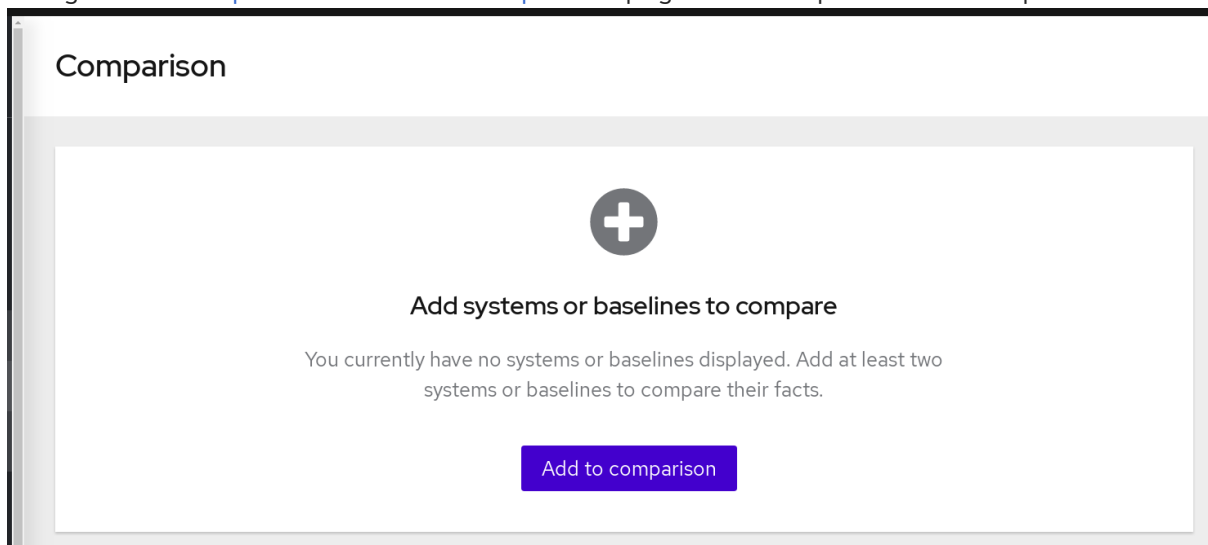
Baselines are standard configurations that a group of systems must maintain. Configurations are sets of name:value pairs. You can either create the pairs or copy them from an existing system configuration.

### Prerequisites

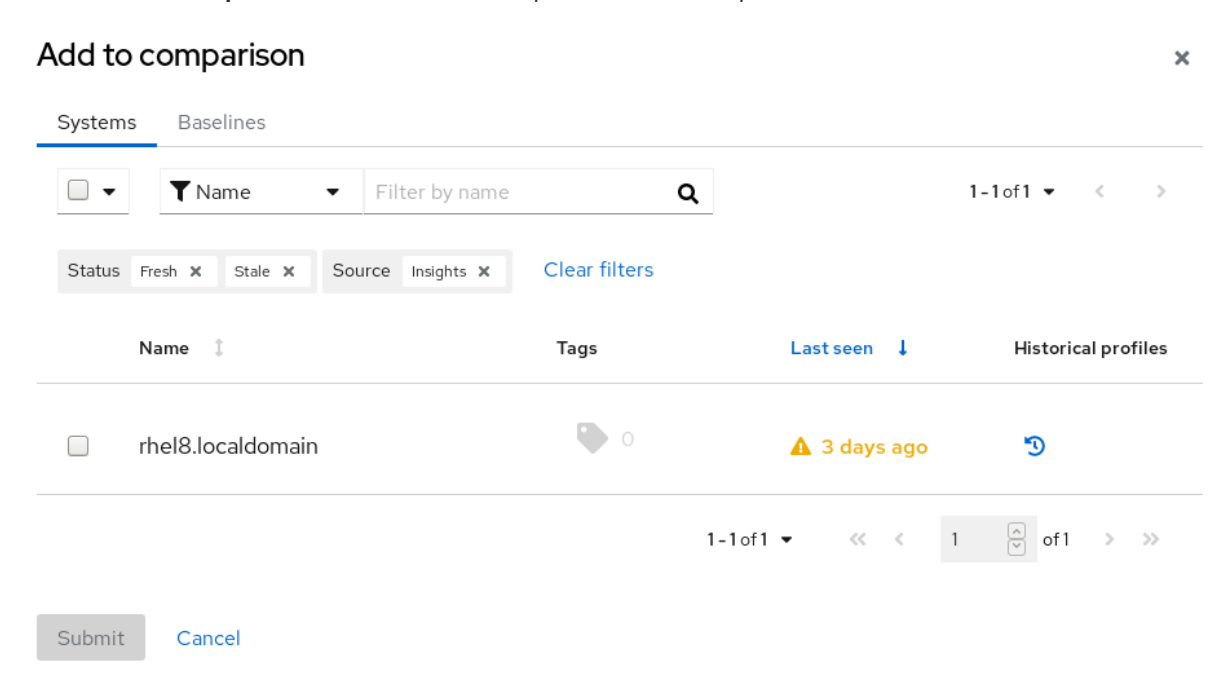
- The Insights client must be installed and running on the systems before those systems can upload to the drift service.

### Procedure

1. Navigate to the [Operations > Drift > Comparison](#) page. The Comparison screen opens.

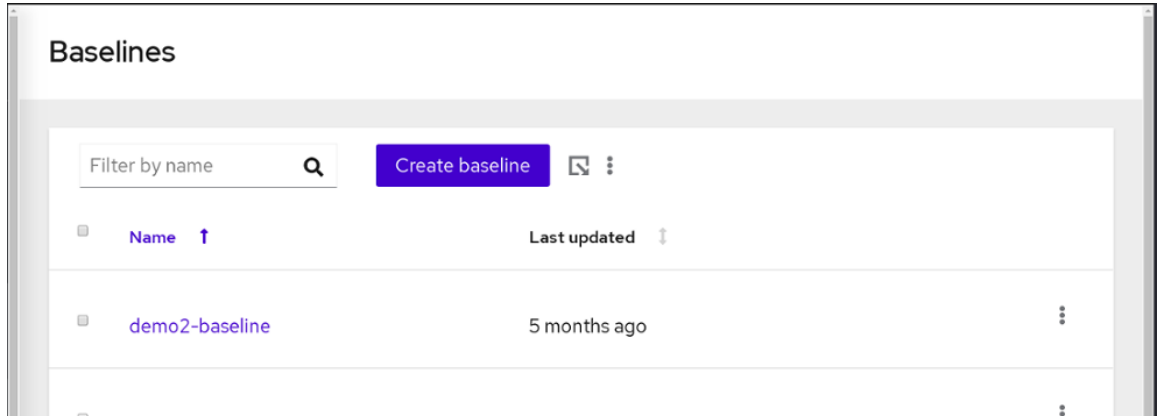


2. Click **Add to comparison**. The Add to comparison screen opens.



- On the **Systems** tab, add the systems you want to compare. This screen also lists any baselines that already exist in your Insights for Red Hat Enterprise Linux inventory.

- On the **Baselines** tab, create baselines in **name:value** form, or copy and paste them from an existing baseline. This screen also lists any baselines that already exist in your Insights for Red Hat Enterprise Linux inventory.



3. Click **Submit** to add your systems or baselines.

## CHAPTER 3. USING SYSTEM FACTS TO COMPARE SYSTEMS IN YOUR INSIGHTS INVENTORY

Use system facts to create useful information about your system, or to provide comparisons between systems. Analyzing the facts about components in your Insights for Red Hat Enterprise Linux inventory gives you a view of your system, or multiple systems, at a particular time. You can also create comparisons of your systems over time to learn about and analyze changes to systems.

### 3.1. CREATING AND ADDING TO COMPARISONS

Complete the following steps to create or add comparisons in your Insights for Red Hat Enterprise Linux inventory.

#### Prerequisites

- Insights client must be installed and operational on the systems.

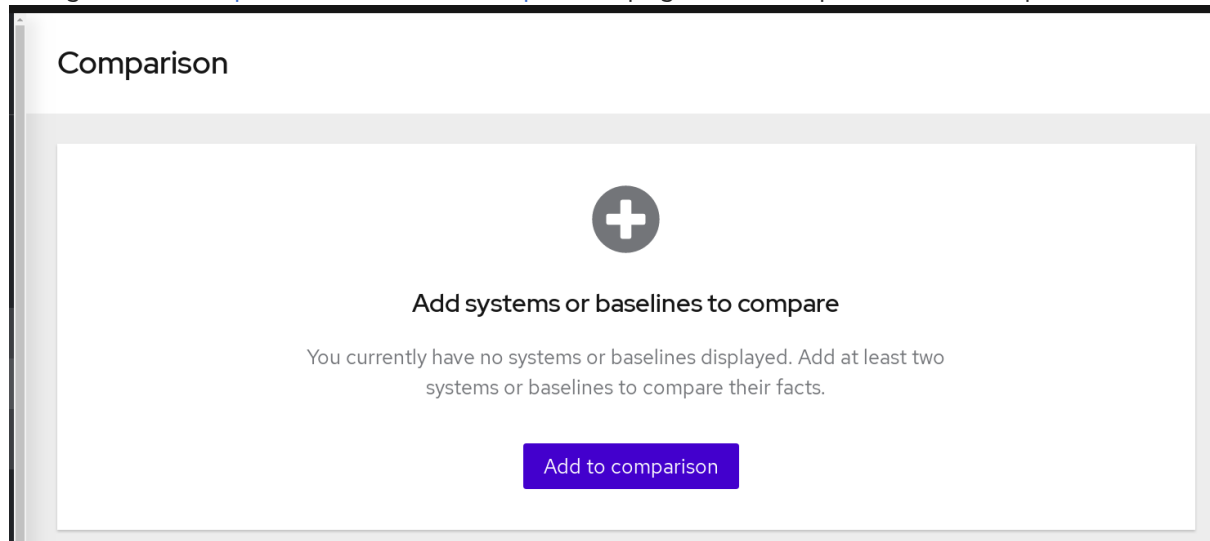


#### NOTE

For steps on how to install the Insights client and register your systems to Insights for Red Hat Enterprise Linux, see [Red Hat Insights for Red Hat Enterprise Linux, Get Started instructions](#).

#### Procedure

1. Navigate to the [Operations > Drift > Comparison](#) page. The Comparison screen opens.



2. Click **Add to comparison**. The Add to comparison screen opens.

Add to comparison ✕

Systems Baselines

---

Name Filter by name

1-1 of 1 < >

Status
Fresh ✕
Stale ✕

Source
Insights ✕

Clear filters

	Name <span style="font-size: small;">↑</span>	Tags	Last seen <span style="font-size: small;">↓</span>	Historical profiles
<input type="checkbox"/>	rhel8.localdomain	0	⚠ 3 days ago	🔄

1-1 of 1
<<
<
1
>
>>
of 1

Submit
Cancel

- On the **Systems** tab, you can add any systems you want to compare. This screen also lists any baselines that already exist in your Insights for Red Hat Enterprise Linux inventory.
- On the **Baselines** tab, you can create baselines. This screen also lists any baselines that already exist in your Insights for Red Hat Enterprise Linux inventory.

### Baselines

Create baseline
⌵
⋮

	Name <span style="font-size: small;">↑</span>	Last updated <span style="font-size: small;">↑</span>	
<input type="checkbox"/>	demo2-baseline	5 months ago	⋮

3. Click **Submit** to add your systems or baselines for comparison.



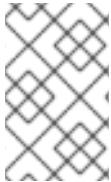
#### NOTE

- At any time, you can add more systems by clicking the **Add to comparison** button near the top of the Comparison screen.
- Similarly, you can remove a particular system under comparison by clicking the **X** symbol on the upper-right corner of the individual system name.
- You can remove all systems under comparison by clicking the options menu (⋮) located at the top.
- Click **Clear all comparisons** to start again.

## CHAPTER 4. REVIEWING AND MANAGING BASELINES

A baseline is a standard configuration that a group of systems must maintain. A configuration is a set of name:value pairs that you can either create or copy from an existing system configuration.

You can manage system profile baseline definitions for your organization. You can edit baselines by changing the values of facts, or you can delete the facts. You can use the drift user interface to create, copy, edit, delete, and export baselines.



### NOTE

You can also use the baselines API to create and manage baselines. See the Red Hat Insights for Red Hat Enterprise Linux API documentation for information about how to use the REST API to query and edit baselines.

## 4.1. CREATING NEW BASELINES

You can use several methods to create a baseline:

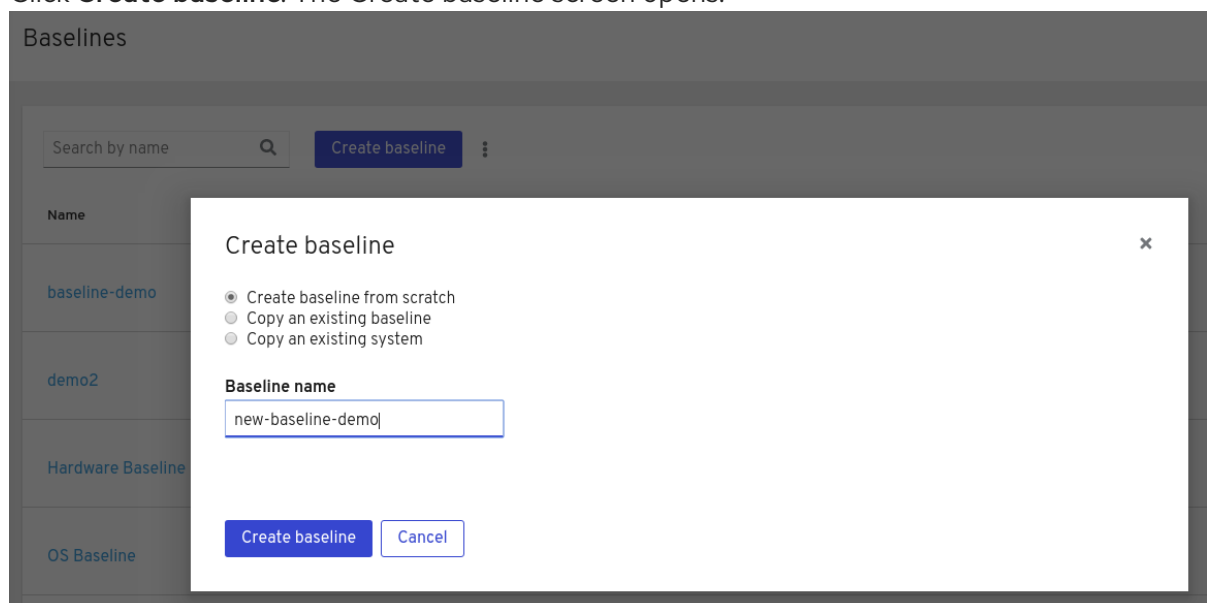
- Create an entirely new baseline
- Copy an existing baseline
- Copy an existing system
- Copy a historical system profile

### 4.1.1. Creating an entirely new baseline


Create a baseline from scratch.

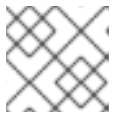
#### Procedure

1. Navigate to the [Operations > Drift > Baselines](#) page. The Baselines screen opens.
2. Click **Create baseline**. The Create baseline screen opens.



3. Select **Create baseline from scratch**.

4. In the **Baseline name** field, enter a name for the baseline you want to create.
5. Click **Create baseline**. The Edit screen for the new baseline opens.
6. Click **Add fact or category**. The **Add fact** screen for the new baseline opens.
7. If this is a category (parent fact) under which you will add subfact(s), select **This is a category** and only enter the **Category name**. Click **Save**.
8. Otherwise, enter the **Fact name** and **Value**, then click **Save**.
9. To add sub facts under a category, click the Options menu  next to a category, and then select **Add sub fact**. You can also use the More Options menu to edit or delete a category.

**NOTE**


Any changes performed are saved automatically.

10. Once you have completed adding all facts and sub facts and their values, navigate to [Operations > Drift > Baselines](#) to view and manage the baseline you have created.

#### 4.1.2. Copying an existing baseline to create a new baseline

You can copy an existing baseline to create a new baseline.

##### Procedure

1. Navigate to the [Operations > Drift > Baselines](#) page. The Baselines screen opens.
2. Click **Create baseline**. The Create baseline screen opens.
3. Select **Copy an existing baseline or historical profile**
4. In the **Baseline name** field, enter a name for the new baseline.
5. From the list of existing baselines, choose the baseline you want to copy and click **Create baseline**. The screen that opens has the newly created baseline with all facts populated.
6. Use the More Options menu  next to a fact to edit or delete that fact.
7. To add a new fact or category, click **Add fact or category**. You can also add a value to a category.

Once you have finished adding or editing facts and their values, navigate to the [Operations > Drift > Baselines](#) page to can view and manage the baseline you have created.

**NOTE**


Based on typical expectations about fact behavior, the drift service alerts users to facts that might require attention. It does not flag facts that differ from each other if the differences are expected, or if they are unimportant. It flags only important differences or similarities that might cause problems.

You can then address these exceptions to typical fact behavior.

### 4.1.3. Copying an existing system to create a new baseline

You can copy an existing system to create a new baseline.

#### Procedure

1. Navigate to the [Operations > Drift > Baselines](#) page. The Baselines screen opens.
2. Click **Create baseline**. The Create baseline screen opens.
3. Select **Copy an existing system**
4. In the Baseline name field, enter a name for the new baseline.
5. From the existing list of systems, select the system with the baseline that you want to copy and click **Create baseline**. The screen that opens has the newly created baseline with all facts populated.
6. To delete existing facts, use the **More options** menu . This menu is located to the right of the blue **Add fact or category** button.
7. To add a new fact or category, click **Add fact or category**. You can also add a sub fact to a category.

**NOTE**

Based on typical expectations about fact behavior, the drift service alerts users to facts that might require attention. It does not flag facts that differ from each other if the differences are expected, or if they are unimportant. It flags only important differences or similarities that might cause problems. You can then address these exceptions to typical fact behavior.

Once you have completed adding or editing facts, sub facts and their values, you can view and manage the baseline you have created by navigating to **drift → Baselines**.



### 4.1.4. Copying a historical system profile to create a new baseline

You can copy a historical system profile to create a new baseline.

#### Procedure

1. Navigate to the [Operations > Drift > Baselines](#) page. The Baselines screen opens.
2. Click **Create baseline**. The Create baseline screen opens.



3. Select **Copy an existing system or historical profile**
4. In the Baseline name field, enter a name for the new baseline.
5. From the existing list of systems, select the system that has the baseline you want to copy.
6. In the Historical profile column, click the Historical profile icon  .
7. From the drop-down menu, select the profile you want to copy, based on the date and timestamp. After you make the selection, drift displays the timestamp of the selected profile in the **Name** column. If you change your mind and want to select a different historical profile, click the **X** next to the profile timestamp, and select a different historical profile.
8. When you are satisfied that you have the particular profile you want to copy, click **Create baseline**. A screen opens with the newly created baseline, with all facts populated.
9. To edit or delete existing facts, use the More Options menu  next to a fact.
10. To add a new fact or category, click **Add fact or category**. You can also add a value to a fact.

Once you have completed adding or editing facts, and their values, navigate to [Operations > Drift > Baselines](#) to view and manage the baseline you created.



#### NOTE

Based on typical expectations about fact behavior, the drift service alerts users to facts that might require attention. It does not flag facts that differ from each other if the differences are expected or which are unimportant. It flags only important differences or similarities that might cause problems. You can then address these exceptions to the typical fact behavior in a baseline.

## 4.2. EDITING OR DELETING BASELINES

You can rename a baseline and change, add, or delete facts, categories, and sub facts.

### 4.2.1. Editing a baseline

#### Procedure

1. To edit a baseline, click the More options icon at the end of the row containing the baseline name. The Edit screen opens.
2. Click **Add fact or category** to add facts or categories to the baseline. The Add fact screen opens.
3. If this is a category (parent fact) under which you will add sub fact(s), select **This is a category** and enter only the **Category name**. Click **Save**.
4. To edit facts, select the item you want to edit and click **Edit fact** on the More options menu, located on the right side of the line that contains the fact. The Edit fact screen opens.
5. Change the name or value and click **Save**.

6. To delete facts or categories, select the items you want to delete and click **Delete fact** on the More options menu, located on the right side of the line that contains the fact.
7. For fact categories, click the More options menu, located on the right side of the line that contains the fact category.
8. Click to **Add sub fact**, **Edit category**, or **Delete category**. The corresponding screen opens. Enter or edit the information requested, and then click **Save**.

## 4.2.2. Deleting a baseline

### Procedure


1. Navigate to the [Operations > Drift > Baselines](#) page. The Baselines screen opens.
2. Use the drop-down checkbox to select the baseline (or baselines) that you want to delete.



When you make a selection, the

field indicates the number of baselines that

you selected.

3. To delete a baseline in the list, click **Delete** on the More options menu  at the top of the screen. An alert opens with a reminder that deleting a baseline cannot be undone. If you still want to proceed, click **Delete baseline**.

## 4.3. COMPARING SYSTEM CONFIGURATIONS TO BASELINES

You can compare system configurations to one or more baselines to identify discrepancies in your environment and perform drift analysis.

### Procedure

1. Navigate to the [Operations > Drift > Comparison](#) page. The Comparison screen opens.
2. Click **Add systems or baselines**.
3. On the **Systems** tab, select one or more systems from the list. Alternatively, you can search for a system by name, and then select the system.
4. On the **Baselines** tab, select one or more baselines from the list. Alternatively, you can search for a baseline by name, and then select the baseline.
5. Click **Submit**.

### Optional procedures


Comparison results can also be filtered by baseline facts. This feature allows you to focus on the most relevant data when creating comparisons.

1. Click the **Fact Name** drop down and choose **Fact type**.
2. Next, click on the **Show** drop down and then check the box next to **Baseline facts only**.

At any time, you can add more systems and baselines by clicking the **Add System** button on the right side of the systems already added for comparison.

To remove a particular system or baseline from a comparison, click the **X** sign in the upper right corner of the individual system or baseline name panel.

To remove all systems and baselines from a comparison, click **Clear all comparisons** from the Options

menu  located at the top of the screen.

You can view the displayed comparison result and filter as necessary by fact name, comparison state, and category. To export the result and any current selections (such as filters) to a comma-separated values (CSV) or JavaScript Object Notation (JSON) file, click **Export as CSV** or **Export as JSON** in the Options menu.

## 4.4. EXPORTING BASELINES


You can export all system baselines, or individual baseline facts or categories, to a comma-separated values (CSV) or JavaScript Object Notation (JSON) file. You can then analyze them externally. If you applied any search filters to the baselines you are exporting, the filters are preserved in the downloaded CSV or JSON file.

### 4.4.1. Exporting individual baselines, facts, and categories

You can export individual baselines to CSV or JSON files. You can also use the same procedure to export individual facts or categories.

#### Procedure

1. Navigate to the [Operations > Drift > Baselines](#) page. The Baselines screen opens.
2. Select a baseline from the list.
3. Click the More options menu for the baseline and select **Edit**. The Edit screen appears.


4. Click the Export icon (  ) and choose **Export to CSV** or **Export to JSON**.

### 4.4.2. Exporting all system baselines

You can export all system baselines to a comma-separated values (CSV) or JavaScript Object Notation (JSON) file.

#### Procedure

1. Navigate to the [Operations > Drift > Baselines](#) page. The Baselines screen opens.

2. Click the Export icon (  ) next to the **Create baseline** button.
3. Choose **Export to CSV** or **Export to JSON**.

## CHAPTER 5. USING SYSTEM FACTS IN COMPARISONS

System facts are important components that help you understand your system comparisons. Examining them reveals detailed information about the performance and changes in your Insights for Red Hat Enterprise Linux system inventory. System facts also alert you to system components whose state is unknown, as well as identifying parts of your systems that require attention.

### 5.1. THE ROLE OF SYSTEM FACTS IN COMPARISONS

Comparison states based on observed fact values can help you manage your system. The drift service identifies facts whose behavior differs from expectations and facts whose state is unknown. It also alerts you to facts that require attention.

The drift service displays the observed fact values in different colors.

- A red icon indicates an issue you should examine.



- A green icon denotes an expected state or value.



- A state shown in black, marked with a question mark icon, indicates that the expected state of the fact is unknown.



Some facts are system-specific and are considered unique. Their state is marked in red if values are equal for a given comparison. For example, if facts for **fqdn** and IP addresses are marked in red, they require your attention.

The drift service expects other facts, such as **last\_boot\_time**, to be different for all compared systems. For such facts, it does not highlight the differences. The service marks the comparison state as unknown (no opinion).

### 5.2. HOW FACTS ARE USED IN COMPARISONS

Comparison states based on observed fact values provide guidance in managing your system. The drift service identifies facts whose behavior differs from expectations, facts whose state is unknown, and alerts users to facts that require attention.

### 5.3. FILTERING SYSTEM FACTS USING THE USER INTERFACE (UI)

You can filter system facts in multiple ways:

- by the fact comparison state
- by fact name
- by fact category.

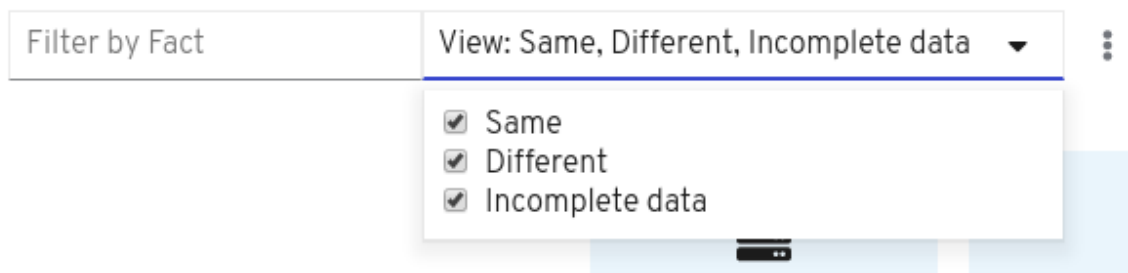
#### 5.3.1. Selecting systems to compare

**Procedure**

1. Navigate to the [Operations > Drift > Comparison](#) page. The Comparison screen opens.
2. On the Comparison screen, click **Add systems or baselines**.
3. On the Add to Comparison screen, select the **Systems** tab to see the systems that are checked into your Insights for Red Hat Enterprise Linux inventory.
4. In the Name column, select the checkboxes for two or more systems and click **Submit**. The Comparison screen opens, showing the state of facts in the systems. Comparison results can also be filtered by baseline facts. This feature allows you to focus on the most relevant data when creating comparisons.
5. Click the **Fact Name** drop down and choose **Fact type**.
6. Next, click on the **Show** drop down and then check the box next to **Baseline facts only**.

**5.3.2. Filtering by comparison state****Procedure**

1. Click the **View** drop-down list and select
  - a. **Same** to show only the facts for which values are the same,
  - b. **Different** to show only the facts that are different, or
  - c. **Incomplete Data** to show only the facts where the information is incomplete.
 You can also select a combination of **Same**, **Different**, and **Incomplete data** states, and clear selections as necessary. When you first add systems for comparison, all three options are selected by default.

**5.3.3. Filtering by fact name****Procedure**

1. Enter your first fact name in the search box at the top. For example, entering **kernel** as a filter displays all facts with names that contain **kernel**.
2. To view all packages, enter **installed\_packages** in the search box.
3. To add additional fact names, type the fact name in the search box and press **Enter**. You can add as many fact names as you need for your filter.

### 5.3.4. Filtering by fact category


#### Procedure

1. Enter your first fact category in the search box to compare systems by that category. Examples include **installed\_packages**, **installed\_services**, **kernel\_modules**, **network\_interfaces**, **yum\_repos**, **cpu\_flags**, and **enabled\_services**.
2. To add additional fact categories, type the category name in the search box and press **Enter**. You can add as many fact categories as you need for your filter.

### 5.3.5. Using the UI to sort system facts

You can sort system facts in the UI in alphabetical order.


#### Procedure

1. To switch sorting between ascending and descending alphabetical order, click the arrow next to **Fact** ( **Fact**  ).  
The list of system facts appears in ascending order by default.

### 5.3.6. Sorting facts by comparison state

You can also sort system facts by the comparison **State**.

#### Procedure

1. To switch to sorting by state, click the arrow next to **State** ( **State**  ).  
If you have any search filters applied to the list of system facts, the search filters are also applied to the list.

### 5.3.7. Examples and screenshots

In the following example screen, you can see the system comparison data filtered by facts that show a difference across systems. Some facts, such as the **fqdn**, are expected to be different for each system, but the installed packages are expected to remain the same. Over time, some packages have been upgraded on system 1, but not on system 2 and system 3. To see these changes, expand the fact category **installed\_packages**.

Comparison

Filter by fact View: Same, Different, Incomplete data Add to comparison 1 - 24 of 24

State Same Different Incomplete data

Fact ↑	State ↓	rhel6-chicago.example.com ☆ 20 Apr 2020, 13:37 UTC	rhel6-tokyo.example.com ☆ 20 Apr 2020, 13:19 UTC	soundwave.infra.pnhc ☆ 20 Apr 2020, 14:34 UTC
bios_release_date	!	04/01/2014	01/01/2011	12/12/2018
bios_vendor	!	SeaBIOS	Seabios	Phoenix Technologies LT
bios_version	!	1.11.0-2.el7	0.5.1	6.00

To lock in multiple filters on facts, type the fact name in the text box and press **Enter**. This functionality is powered by an "OR" operator that allows you to filter out anything that does not match any of the facts you selected.

In the following example, the filtered list of facts shows only facts with names that match **bios** or **arch**.

Comparison

Filter by fact View: Same, Different, Incomplete data Add to comparison 1 - 6 of 6

State Same Different Incomplete data Fact name bios arch Clear all filters

Fact ↑	State ↓	test-client-msager-rhel83-obf ☆ 08 Feb 2021, 05:45 UTC
arch	✓	x86_64
bios_release_date	✓	04/01/2014
bios_vendor	✓	SeaBIOS
bios_version	✓	113.0-2.module+el8.2.1+7284+aa32a2c4
cpu_flags	✓	
arch_capabilities	✓	enabled
installed_packages	✓	
libarchive	✓	3.3.2-9.el8.x86_64

## 5.4. FILTERING SYSTEM FACTS USING URL PARAMETERS

The drift service enables multi-fact filtering, which enables you to create custom comparisons. You can filter system facts in several ways:

- by the fact comparison state
- by fact name, or
- by fact category

### 5.4.1. Sorting by editing the URL

To expedite filtering, you can edit the URL parameters. The following example displays a sample URL and its parameters. The numbers of the parameters correspond to the numbered items in the parameter descriptions that follow.

```
[package]insights/drift/?baseline_ids=<baseline-id>&system_ids=<system-id>&hsp_ids=<hsp-id>&reference_id=<reference-id>&filter[name]=bios,arch&filter[state]=same,different,incomplete_data&sort=-state,fact
```

### Parameters

1. `[package]insights/drift/`
2. `[package]?`
3. `[package]baseline_ids=<baseline-id>&system_ids=<system-id>&hsp_ids=<hsp-id>`
4. `[package]&reference_id=<reference-id>`
5. `[package]&filter[name]=bios,arch`
6. `[package]&filter[state]=same,different,incomplete_data`
7. `[package]&sort=-state,fact`
8. `[package]&filter[show]=baseline`

### Parameter descriptions

You can manually enter these parameters if desired, but changes that you make in the UI automatically populate to this parameter list.



1. **App service:** This reflects the Red Hat Insights for Red Hat Enterprise Linux you are using. This example uses the drift service on Red Hat Insights for Red Hat Enterprise Linux.
2. **Search parameter:** This is the character that tells drift you want to search on the parameters that follow.
3. **IDs of systems/baselines/historical profiles:** These are the IDs of the systems, baselines and historical profiles in your comparison. Each is preceded by the & symbol after the first ? symbol, and the respective parameter type (`baseline_ids`, `system_ids` or `hsp_ids`).
4. **ID of system/baseline/historical profile to use as reference:** This is the ID of the system, baseline or historical system profile used as a reference to which all other facts compare. The `reference-id` must be specified in one of the parameters (`baseline_ids`, `system_ids` or `hsp_ids`). If it is not specified, the parameter does not set a reference for comparison.
5. **Fact name filters:** This fact name filter takes the format `&filter[name]=fact`. For example, `&filter[name]=bios,arch` To specify multiple facts, separate them with commas and no spaces.
6. **State filters:** These filters take a format similar to as fact name filters, but use `&filter[state]`. Valid values for `filter[state]` are: **same**, **different**, and **incomplete\_data**. To specify multiple facts, separate them with commas and no spaces.
7. **Table sorting:** This parameter uses the form `&sort=state`. To specify multiple facts, separate them with commas and no spaces. To sort in descending order, add a minus sign (-); for example, `&sort=-fact`. If no minus sign appears, the sort occurs in ascending order. To omit state sort ("no sort"), omit state sort from the parameter. You cannot omit fact sorting. If you do not specify a value for fact sorting, the sort defaults to ascending order.



## 5.5. SORTING SYSTEM FACTS

You can sort system facts in the same way that you can sort using the User Interface (UI) or by editing URL parameters.

### Sorting using the UI

You can sort system facts in the UI alphabetically. Click the arrow next to **Fact** (  ) to switch sorting between ascending and descending order. Note that facts are shown in ascending order by default. You can also sort system facts by the comparison **State**. Click the arrow next to **State** (  ) to switch to sorting by state.



#### NOTE

Sorting works in combination with any applied filters. That is, if you have filtered for installed packages or viewing facts by comparison state, the filtered data can be sorted alphabetically or by comparison state.

### Sorting by editing the URL

Sorting can be expedited by editing the URL parameters. Examine the following URL for an explanation of how to use this functionality:

**[package]insights/drift/?baseline\_ids=<baseline-id>&system\_ids=<system-id>&hsp\_ids=<hsp-id>&reference\_id=<reference-id>&filter[name]=bios,arch&filter[state]=same,different,incomplete\_data&sort=-state,fact**

#### Parameters

1. **[package]insights/drift/**
2. **[package]?**
3. **[package]baseline\_ids=<baseline-id>&system\_ids=<system-id>&hsp\_ids=<hsp-id>**
4. **[package]&reference\_id=<reference-id>**
5. **[package]&filter[name]=bios,arch**
6. **[package]&filter[state]=same,different,incomplete\_data**
7. **[package]&sort=-state,fact**

#### How to use parameters

1. **App service:** This reflects the app service you are using. In this case, drift on Red Hat Insights for Red Hat Enterprise Linux.
2. **Search parameter:** This is the character that tells drift you want to search on the parameters that follow.
3. **IDs of systems/baselines/historical profiles:** These are the IDs of the systems, baselines and historical profiles in your comparison. Each is preceded by the & symbol after the first ? symbol, and the respective parameter type (baseline\_ids, system\_ids or hsp\_ids).

4. **ID of system/baseline/historical profile to use as reference:**This is the ID of the system, baseline or historical system profile that will be used as a reference to compare all other facts to. The reference-id must be specified in one of the parameters (baseline\_ids, system\_ids or hsp\_ids). If not specified, no reference is set for the comparison.
5. **Fact name filters:**Begins with the & symbol and filter[name]. Each fact name filter applied is added after the = symbol and separated by a comma with no spaces.
6. **State filters:**Same as fact name filters, but preceded by filter[state]. Valid values for filter[state] are: same, different and incomplete\_data. Multiple values can be specified by separating them with a comma and no space.
7. **Table sorting:** Preceded by the & symbol, state and/or fact are added after sort= and are comma separated. If state or fact is preceded by a - symbol, then it is sorted in descending order; otherwise, it sorts in ascending order. State sort has the ability to have no sort. In this case, state will not be added in the url parameter. Fact sorting, on the other hand, if left off, will default to ascending order.

These parameters can be entered manually, but their primary function is to auto-populate as you make changes in the UI.

## 5.6. USING OBFUSCATED VALUES IN COMPARISONS

The Insights client provides both IP address obfuscation and host name obfuscation.

If one of your fact values has been redacted to protect sensitive information, drift informs you that your comparison contains hidden (obfuscated) data.

Obfuscated fact values show the following characteristics:

- The value cell is grayed out.
- A lock icon appears in the grayed-out cell, along with a tooltip that states that the value has been redacted.
- A tooltip appears on the "state" icon that describes why the row has the "incomplete data" state.

The screenshot displays two examples of data redaction in the Insights client interface. The top example shows a system named 'test-client-msager-rhel83-obf' with a fact 'host0' that has been redacted. A tooltip indicates: 'This data has been redacted from the insights-client upload.' The bottom example shows a system named 'test-client-msager-rhel83' with a fact 'fqdn' that is redacted. A tooltip explains the 'incomplete data' state: 'Incomplete data - At least one system fact value in this row is redacted.'

If one of the values in the comparison is redacted, the state of the comparison for that fact shows "incomplete data."

### Additional resources

- For more information about setting up client data obfuscation, see the [Client Configuration Guide for Red Hat Insights](#).

## 5.7. UNDERSTANDING MULTI-VALUE FACTS

Multi-value facts provide more detailed information to help troubleshoot system issues. For example, drift stores a list of all installed versions for a given package name. This results in multi-value facts being available for the package. Inventory and drift APIs also provide facts with multiple values.

The following example shows a system with two kernel packages. Multi-value fact support makes it possible to view values for both packages on one system.

Comparison

The screenshot shows a comparison interface with the following elements:

- Search bar: Fact name, Filter by fact, Add to comparison, 1-2 of 2
- Filters: State, Same, Different, Incomplete data, Fact name: kernel, Reset filters
- Table with columns for Fact, State, and two system columns (msager-rhel84-mssql and msager-rhel9).

Fact	State	msager-rhel84-mssql ☆ 04 Aug 2022, 05:34 UTC	msager-rhel9 ☆ 04 Aug 2022, 19:49 UTC
installed_packages	ⓘ		
kernel	ⓘ		
kernel	ⓘ	4.18.0-305.el8.x86_64	
kernel	ⓘ	4.18.0-372.16.1.el8_6.x86_64	
kernel	ⓘ		5.14.0-70131.el9_0.x86_64
kernel-core	ⓘ		
kernel-headers	ⓘ	4.18.0-372.16.1.el8_6.x86_64	
kernel-modules	ⓘ		
kernel-tools	ⓘ	4.18.0-372.16.1.el8_6.x86_64	5.14.0-70131.el9_0.x86_64
kernel-tools-libs	ⓘ	4.18.0-372.16.1.el8_6.x86_64	5.14.0-70131.el9_0.x86_64
os_kernel_version	ⓘ	4.18.0	5.14.0

With this enhanced level of detail, you can correctly evaluate and compare all installed versions when performing an analysis.

The example below shows a system that contains a package compiled for two architectures. It has **glibc** packages for both i686 and x86\_64 architectures.

Comparison

Fact name:  Add to comparison 1-1 of 1

State: Same  Different  Incomplete data  Fact name: glibc  Reset filters

Fact	State	msager-rhel8 ☆ 08 Aug 2022, 06:52 UTC	msager-rhel82 ☆ 08 Aug 2022, 15:12 UTC
installed_packages	!		
glibc	!		
	!		2.28-189.5.el8_6.i686
	!		2.28-189.5.el8_6.x86_64
	!	2.28-42.el8.x86_64	
glibc-all-langpacks	!	2.28-42.el8.x86_64	2.28-189.5.el8_6.x86_64
glibc-common	!	2.28-42.el8.x86_64	2.28-189.5.el8_6.x86_64
glibc-gconv-extra	!		

1-1 of 1 << < 1 of 1 > >>

## Additional resources

- For more information about multiple values in the Inventory and drift APIs, see the Red Hat Insights for Red Hat Enterprise Linux API documentation: [https://access.redhat.com/documentation/en-us/red\\_hat\\_insights/1-latest/html/system\\_comparison\\_api\\_documentation/](https://access.redhat.com/documentation/en-us/red_hat_insights/1-latest/html/system_comparison_api_documentation/) [System Comparison API Documentation].

## 5.8. USING MULTI-FACT FILTERING

Use multi-fact filtering to create custom comparisons for your systems. You can filter your comparison queries by specific groups of facts and tags.

You can use multi-fact filtering to:

- Have multiple inputs in the fact name field.
- Avoid swapping back and forth between multiple filters.
- Exclude irrelevant facts.
- Compare facts related to a specific issue for improved troubleshooting.
- Share comparisons with other administrators or colleagues.

## 5.9. AVAILABLE FACTS AND THEIR FUNCTIONS

The following table lists the system facts that you can use in system comparisons.

Table 5.1. System Facts

Fact Name	Description	Example Value
<b>Ansible</b>	Category with a list of Ansible-related facts	controller_version with a value of 4.0.0

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, azure, aws, 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, vmware, 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> .
<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>

Fact Name	Description	Example Value
<b>mssql</b>	Category with a list of MSSQL-related facts	mssql_version with a value of 15.0.4153.1
<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>sap_instance_number</b>	SAP instance number	<b>42</b>
<b>sap_sids</b>	SAP system ID (SID)	<b>A42</b>
<b>sap_system</b>	Boolean field that indicates if SAP is installed on the system	<b>True</b>
<b>sap_version</b>	SAP version number	<b>2.00.052.00.1599 235305</b>

Fact Name	Description	Example Value
<b>satellite_managed</b>	Boolean field that indicates is a system is registered to a Satellite server.	<b>FALSE</b>
<b>selinux_current_mode</b>	Current SELinux mode	<b>enforcing</b>
<b>selinux_config_file</b>	SELinux mode set in the config file	<b>enforcing</b>
<b>system_memory</b>	Total system memory in human-readable form	<b>3.45 GiB</b>
<b>tuned_profile</b>	Current profile resulting from the command <b>tuned-adm active</b>	<b>desktop</b>
<b>yum_repos</b>	List of yum repositories. The repository name is added to the beginning of the fact. Each repository has the associated facts <b>base_url,enabled,</b> and <b>gpgcheck.</b>	<b>Red Hat Enterprise Linux 7 Server (RPMs).base_ur</b> I would have the value <a href="https://cdn.redhat.com/content/dist/rhel/server/7/\$releasever/\$basearch/os">https://cdn.redhat.com/content/dist/rhel/server/7/\$releasever/\$basearch/os</a>

## CHAPTER 6. COMPARING SYSTEMS AND SYSTEM PROFILES

You can compare systems to each other, to profiles, and to a reference point. When you have selected all the systems you want to compare, you can use the Comparison page to compare the facts in the systems.

### 6.1. COMPARING SYSTEMS TO A REFERENCE POINT

In some cases, you might want to compare all systems to a single reference point, rather than compare all systems with each other, or compare them as a group. For example, you might need to compare all systems to a baseline so that all systems are calculated against that baseline. You can also compare a system against time-stamped profiles to understand where and when changes have occurred.

You might also want to invert a comparison. For example, instead of comparing profiles with old timestamps against the latest profile, you might want to compare all profiles against the oldest known working version of a system. Such a comparison enables you to identify changes that deviate from the reference point.

#### 6.1.1. Comparing systems to a single point of reference

You can compare multiple systems, or all systems, to a single reference point.

##### Procedure

- On the Comparison screen, click the star icon (  ) on the system header.

The drift service compares each of your selected systems with the fact values in the reference system. The reference values are displayed in the first position column, with the column header highlighted in blue.

The fact values are displayed with every difference highlighted in red.



##### NOTE


A fact category with multiple values has no highlights until you expand the fact to view details. When you do so, drift shows the specific fact details highlighted in red.

### 6.2. USING THE HISTORY OF SYSTEM PROFILES

Every time you submit a system for comparison, the submission checks in the profile and marks it with a timestamp. By examining the different profile versions, you can view the system over time.



##### NOTE

On the Choose systems screen, if a system is marked with a time stamp  icon, you might be able to open that system directly. Otherwise, add the system you want to compare into drift before comparing systems.

### 6.3. EXPORTING SYSTEM COMPARISON OUTPUT

The drift service allows you to export system comparison output, including any current selections (such




as filters), to a comma-separated values (CSV) file or a JavaScript Object Notation (JSON) file. You can then open the CSV or JSON file with the tool of your choice to compare exported facts and analyze discrepancies in your systems.



#### NOTE

The exported CSV or JSON report preserves all your current selections on the system comparison output, including any applied filters. To see all the facts in nested fact categories (such as **installed\_packages**), expand all the nested fact categories before exporting the system comparison output to the report.


#### Procedure

1. On the Comparison output screen for two or more systems, click the **Export** icon , and then select either **Export as CSV**, or **Export as JSON**.
2. Open the CSV or JSON file with the tool of your choice so that you can easily compare exported facts and analyze discrepancies in systems.

### 6.3.1. Exporting system comparison output to a CSV or JSON file

You can export your system profile output, along with any filters you used, to a comma-separated values (CSV) or JavaScript Object Notation (JSON) file. The filters highlight the differences and incomplete data in your system profiles. You can use this information for researching characteristics about your systems, and troubleshooting any issues you discover.

#### Procedure

1. On the Comparison output screen for two or more systems, click the **Export** icon  and then select either **Export to CSV** or **Export to JSON**.

# CHAPTER 7. ENABLING NOTIFICATIONS AND INTEGRATIONS FOR DRIFT EVENTS

## 7.1. ENABLING NOTIFICATIONS AND INTEGRATIONS

You can enable the notifications service on Red Hat Hybrid Cloud Console to send notifications whenever the drift service detects an issue and generates an alert. Using the notifications service frees you from having to continually check the [Red Hat Insights for Red Hat Enterprise Linux dashboard](#) for alerts.

For example, you can configure the notifications service to automatically send an email message whenever the drift service detects a baseline change, or to send an email digest of all the alerts that the drift service generates each day.

In addition to sending email messages, you can configure the notifications service to send event data in other ways:

- Using an authenticated client to query Red Hat Insights APIs for event data
- Using webhooks to send events to third-party applications that accept inbound requests
- Integrating notifications with applications such as Splunk to route drift events to the application dashboard

Enabling the notifications service requires three main steps:

- First, an Organization Administrator creates a User access group with the Notifications administrator role, and then adds account members to the group.
- Next, a Notifications administrator sets up behavior groups for events in the notifications service. Behavior groups specify the delivery method for each notification. For example, a behavior group can specify whether email notifications are sent to all users, or just to Organization administrators.
- Finally, users who receive email notifications from events must set their user preferences so that they receive individual emails for each event.

### Additional resources

For more information about how to set up notifications for drift alerts, see [Configuring notifications on the Red Hat Hybrid Cloud Console](#) and [Integrating the Red Hat Hybrid Cloud Console with third-party applications](#).

# PROVIDING FEEDBACK ON RED HAT DOCUMENTATION

We appreciate and prioritize your feedback regarding our documentation. Provide as much detail as possible, so that your request can be quickly addressed.

## Prerequisites

- You are logged in to the Red Hat Customer Portal.

## Procedure

To provide feedback, perform the following steps:

1. Click the following link: [Create Issue](#)
2. Describe the issue or enhancement in the **Summary** text box.
3. Provide details about the issue or requested enhancement in the **Description** text box.
4. Type your name in the **Reporter** text box.
5. Click the **Create** button.

This action creates a documentation ticket and routes it to the appropriate documentation team. Thank you for taking the time to provide feedback.