



## Red Hat Insights 2022

# System Comparison API のドキュメント

Insights for RHEL ドリフトサービスでの REST API の使用



# Red Hat Insights 2022 System Comparison API のドキュメント

---

Insights for RHEL ドリフトサービスでの REST API の使用

Enter your first name here. Enter your surname here.

Enter your organisation's name here. Enter your organisational division here.

Enter your email address here.

## 法律上の通知

Copyright © 2022 | You need to change the HOLDER entity in the en-US/System\_Comparison\_API\_Documentation.ent file |.

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.

## 概要

このドキュメントは、System Comparison/ドリフトサービスを外部アプリケーションと統合するために利用可能な Web サービスを提供します。ここでは、標準の REST HTTP リクエストとして実装される System Comparison RESTful API の仕様と、コンテンツタイプ JSON の応答を説明します。

## 目次

多様性を受け入れるオープンソースの強化 .....	3
RED HAT ドキュメントへのフィードバック .....	4
<b>第1章 クエリーでの API の使用 .....</b>	<b>5</b>
1.1. スキーマ .....	5
1.2. 基本認証 .....	5
1.3. 仕様 .....	5
1.3.1. REST API エントリーポイント .....	5
1.3.2. HTTP メソッド .....	5
1.3.3. データタイプ .....	6
1.3.4. HTTP ステータスコード .....	6
<b>第2章 比較レポート API .....</b>	<b>7</b>
2.1. システムのみのレポートクエリー .....	8
2.2. システムおよびベースラインレポートクエリー .....	8
2.3. システムおよび過去のシステムプロファイルのレポートクエリー .....	9
2.4. システム、ベースライン、過去のシステムプロファイルレポートクエリー .....	9
2.5. COMPARISON REPORT API のエクスポート .....	9
2.6. 参照クエリー .....	9
<b>第3章 ベースライン API .....</b>	<b>11</b>
3.1. ゼロからの新規ベースラインの作成 .....	11
3.2. 新しいベースラインを作成するために既存のシステムをコピーする .....	12
3.3. 新規作成用の既存の文献のコピー .....	13
3.4. 新規ベースラインの作成への古いシステムプロファイルのコピー .....	13
3.5. ベースラインのソート .....	14
3.6. ベースラインの編集 .....	14
3.7. ベースラインのエクスポート .....	14
3.8. ベースラインの削除 .....	15
<b>第4章 過去のシステムプロファイル API .....</b>	<b>16</b>
4.1. システムメソッド .....	16
4.2. プロファイルメソッド .....	17



## 多様性を受け入れるオープンソースの強化

Red Hat では、コード、ドキュメント、Web プロパティにおける配慮に欠ける用語の置き換えに取り組んでいます。まずは、マスター (master)、スレーブ (slave)、ブラックリスト (blacklist)、ホワイトリスト (whitelist) の 4 つの用語の置き換えから始めます。この取り組みは膨大な作業を要するため、今後の複数のリリースで段階的に用語の置き換えを実施して参ります。詳細は、[弊社の CTO、Chris Wright のメッセージ](#) を参照してください。

## RED HAT ドキュメントへのフィードバック

弊社のドキュメントに関するご意見やご感想をお寄せください。フィードバックを提供するには、ドキュメントのテキストを強調表示し、コメントを追加します。

### 前提条件

- Red Hat カスタマーポータルにログインしている。
- Red Hat カスタマーポータルでは、このドキュメントは **Multi-page HTML** 表示形式です。

### 手順

フィードバックを提供するには、以下の手順を実施します。

1. ドキュメントの右上隅にある **フィードバック** ボタンをクリックして、既存のフィードバックを確認します。



#### 注記

フィードバック機能は、**マルチページ HTML** 形式でのみ有効です。

2. フィードバックを提供するドキュメントのセクションを強調表示します。
3. ハイライトされたテキスト近くに表示される **Add Feedback** ポップアップをクリックします。ページの右側のフィードバックセクションにテキストボックスが表示されます。
4. テキストボックスにフィードバックを入力し、**Submit** をクリックします。ドキュメントに関する問題が作成されます。
5. 問題を表示するには、フィードバックビューで問題リンクをクリックします。



# 第1章 クエリーでの API の使用

drift サービスを使用すると、Insights for RHEL インベントリに登録されている他のシステムやベースラインと、特定のシステムの設定を比較できます。これにより、REST API を使用してシステム設定をクエリーし、ベースラインの作成および管理を行い、ファクトの値と比較の状態を返すことができます。また、比較しているシステムおよびベースラインの CSV 出力を生成することもできます。



## 注記

- Red Hat Enterprise Linux API の追加の Insights for RHEL サービスドキュメントは、<https://cloud.redhat.com/docs/api-docs> を参照してください。
- 最新の OpenAPI Specification ドキュメントは、『[OpenAPI Specification](#)』を参照してください。

## 1.1. スキーマ

API アクセスは HTTPS 経由で行われ、[https://access.redhat.com/documentation/ja-jp/red\\_hat\\_insights/2022/html/system\\_comparison\\_api\\_documentation/](https://access.redhat.com/documentation/ja-jp/red_hat_insights/2022/html/system_comparison_api_documentation/) [System Comparison API Documentation] のルート URL からアクセスされます。すべてのデータが JSON として送受信されます。JSON ファイルには以下が含まれます。

```
drift-openapi.json
historical-system-profiles-openapi.json
system-baseline-openapi.json
```

## 1.2. 基本認証

ユーザーとパスワードの認証情報は、各 HTTP リクエストとともに渡されます。

### 要求

```
$ curl --user username:password -i https://cloud.redhat.com/api/drift/v1/comparison_report?
system_ids[]=<UUID>
```

または

```
$ curl --user username:password -i https://cloud.redhat.com/api/system-baseline/v1/baselines
```

## 1.3. 仕様

### 1.3.1. REST API エントリーポイント

REST API は、/api URL プレフィックスから利用できます。これは、以下のようにサーバー上でアクセスされます。

```
"/api/drift/v1"
```

### 1.3.2. HTTP メソッド

API リクエストに対して現在サポートされている HTTP メソッドは GET および POST です。

### 1.3.3. データタイプ

データタイプ	説明
整数	整数値
文字列	JSON 文字列
アレイ	<b>item</b> キーワードがアレイに必要です。項目の値は、アレイアイテムのタイプと形式を記述するスキーマです。アレイはネスト化できます。
ブール型	True または False の値
タイムスタンプ	ISO8601 形式のタイムスタンプ

### 1.3.4. HTTP ステータスコード

コード	テキスト	説明
200	OK	成功。
400	不適切な要求	構文が間違っているため、リクエストがサーバーで認識できませんでした。
500	内部サーバーエラー	サーバーに予期しない状況が発生し、リクエストを満たせませんでした。

## 第2章 比較レポート API

比較レポートの目的は、要求されたシステムのシステムプロファイルを取得して、それらのファクトをインベントリーサービスからプルし、それらのファクトを比較レポートに変換することです。システムの情報の取得が必要なだけで比較情報が不要な場合は、インベントリー API を直接使用できます。



### 注記

現在、Web ユーザーインターフェースまたは API のいずれかを使用してシステムを比較する場合は、45 のシステム制限があります。一度に 45 を超えるシステムを比較するリクエストでは、400 エラーが発生することがあります。このようなリクエストは API インフラストラクチャーによって設定されるタイムアウト制限に到達することがあります。

### 例

以下の例では、GET メソッドを使用してシステム設定を比較するために、compare\_report API を使用してインベントリーに登録されているシステムと対話しています。この要求は最新のシステム状態を取得し、指定のシステム ID のインベントリーサービスからデータを使用し、比較レポートを生成します。

### 要求:

GET

api/drift/v1/comparison\_report?system\_ids[]=<UUID>&system\_ids[]=<UUID>

openapi: 3.0.1

info:

version: "1.0"

title: Drift Backend Service

description: Service that returns differences between systems.

license:

name: Apache 2.0

url: <https://www.apache.org/licenses/LICENSE-2.0.html>

servers:

- url: "/api/drift/v1"

paths:

/comparison\_report:

parameters:

- \$ref: '#/components/parameters/rhIdentityHeader'

get:

summary: fetch comparison report

description: "Fetch a comparison report for given system IDs. This will only fetch the most current system state using data from inventory service."

operationId: drift.views.v1.comparison\_report

parameters:

- name: system\_ids[]

in: query

schema:

type: array

items:

type: string

```

format: uuid
required: true
description: list of system IDs to generate report for

```

**応答:**

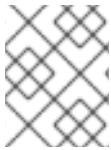
```

'200':
  description: a comparison report for the given system IDs
  content:
    application/json:
      schema:
        $ref: "#/components/schemas/ComparisonReport"
      examples:
        jsonObject:
          summary: A sample comparison report
          externalValue: "https://git.io/fhQ00"
'400':
  $ref: '#/components/responses/BadRequest'
'500':
  $ref: '#/components/responses/InternalServerError'

```

**例**

以下の例では、comparison\_report API を使用してイベントリーに登録されているシステムと対話し、JSON ファイルを POST メソッドを使用した入力として提供し、システムの比較を行います。

**注記**

API 経由で POST メソッドを使用してシステムを比較する場合、システム制限はありません。

**要求:**

```

$ curl -k -u username:password -X POST --header "Content-Type: application/json"
https://cloud.redhat.com/api/drift/v1/comparison_report -d @/directory/example-uuid-file.json

```

JSON ファイルの内容の例を以下に示します。

```

{"system_ids": ["UUID1", "UUID2", "UUID3"]}

```

## 2.1. システムのみのレポートクエリー

API を使用してシステムのみをクエリーできます。

```

GET
api/drift/v1/comparison_report?system_ids[]=55e47a1d-05d7-4abb-963
a-ba513a0a57ad&system_ids[]=....

```

## 2.2. システムおよびベースラインレポートクエリー

API を使用してシステムおよびベースラインレポートをクエリーできます。

■

GET

```
api/drift/v1/comparison_report?system_ids[]=55e47a1d-05d7-4abb-963
a-ba513a0a57ad&baseline_ids[]=79032c7b-8284-44d0-b820-ebbb6d25c5e2
```

## 2.3. システムおよび過去のシステムプロファイルのレポートクエリー

API を使用して、システムおよび過去のシステムプロファイルレポートをクエリーできます。

GET

```
api/drift/v1/comparison_report?system_ids[]=55e47a1d-05d7-4abb-963
a-ba513a0a57ad&historical_system_profile_ids[]=5d7dd2e9-18e5-412e-8f25-efa8a96a4289
```

## 2.4. システム、ベースライン、過去のシステムプロファイルレポートクエリー

API を使用して、システム、ベースライン、および過去のシステムプロファイルレポートをクエリーできます。

GET

```
api/drift/v1/comparison_report?system_ids[]=55e47a1d-05d7-4abb-963
a-ba513a0a57ad&baseline_ids[]=79032c7b-8284-44d0-b820-ebbb6d25c5e2
&historical_system_profile_ids[]=5d7dd2e9-18e5-412e-8f25-efa8a96a4
289
```

## 2.5. COMPARISON REPORT API のエクスポート

比較レポート API を使用して、フィルターなしで指定のシステム、ベースライン、過去のシステムプロファイル ID の比較レポート全体の出力をエクスポートできます。出力には、カテゴリ (親ファクト) およびサブファクトを含むすべてのファクトと、コンマ区切りの値 (CSV) 形式の値が含まれます。

### 要求:

```
$ curl -X GET 'https://username:password@cloud.redhat.com/api/drift/v1/
comparison_report?baseline_ids[]=UUID1&system_ids[]=UUID2&system_ids[]=UUID3'
-H 'accept: text/csv'
```

## 2.6. 参照クエリー

API を使用する場合は、クエリーに参照を設定できます。

GET

```
api/drift/v1/comparison_report?reference_id=UUID1&system_ids[]=
55e47a1d-05d7-4abb-963a-ba513a0a57ad&baseline_ids[]=79032c7b-
8284-44d0-b820-ebbb6d25c5e2&historical_system_profile_ids[]=
5d7dd2e9-18e5-412e-8f25-efa8a96a4289
```



## 注記

**reference\_id** パラメーターはオプションで、クエリー内で1つの参照 ID のみを使用できます。これは、比較クエリーで使用されるパラメーター (システム ID、ベースライン ID、または過去のシステムプロファイル ID) に設定できます。リファレンス ID を使用する場合、Comparison サービスはリスト内のすべてのシステム、ベースライン、および過去のシステムプロファイルを、それと比較します。

## 第3章 ベースライン API

ベースラインは、API を使用して作成および管理できるファクトのセットです。

以下のコマンドラインの例は、ベースライン API を使用してベースラインを作成および管理する方法を示しています。ベースラインをゼロから作成するか、システムインベントリ ID、ベースライン ID、過去のシステムプロファイル ID からコピーしたりできます。

### 3.1. ゼロからの新規ベースラインの作成

ゼロから新しいベースラインを作成できます。

**create.json** ファイルの例:

```
{
  "baseline_facts": [
    {
      "name": "fact name 1",
      "value": "fact value 1"
    },
    {
      "name": "fact name 2",
      "value": "fact value 2"
    },
    {
      "name": "category 1",
      "values": [
        {
          "name": "sub fact name 1",
          "value": "sub fact value 1"
        }
      ]
    }
  ],
  "display_name": "demo baseline"
}
```

以下の例は、JSON データメソッドで POST を使用し、CURL コマンドを使用して、ゼロから新しいベースラインを作成しています。

**要求:**

```
$ curl -u username:password
https://cloud.redhat.com/api/system-baseline/v1/baselines -X
POST -d @create.json -H "content-type: application/json"
```

**応答:**

```
{
  "account": "[account number]",
  "baseline_facts": [
    {
      "name": "category 1",
      "values": [
```

```

    {
      "name": "sub fact name 1",
      "value": " sub fact value 1"
    }
  ],
  },
  { :context: {parent-context}
    "name": "fact name 1",
    "value": "fact value 1"
  },
  {
    "name": "fact name 2"
    "value": "fact value 2"
  }
],
"created": "2020-05-18T20:25:36.328741Z",
"display_name": "demo baseline",
"fact_count": 3,
"id": "9565f205-5816-43b4-953e-a29fed8b40ec",
"updated": "2020-05-18T20:25:36.328745Z"
}

```

### 3.2. 新しいベースラインを作成するために既存のシステムをコピーする

既存のシステムをコピーして、API 経由で新しいベースラインを作成できます。

**create-from-inv.json** ファイルの例:

```

{"inventory_uuid": "<UUID>",
 "display_name": "from-inv1"}

```

**要求:**

```

$ curl -k -X POST -d @create-from-inv.json
https://username:password@cloud.redhat.com/api/system-baseline/v1/baselines
-H 'content-type: application/json'

```

**応答:**

読みやすさを向上するため、一部のファクトが応答から削除されていることに注意してください。

```

{
  [
  {
    "Name": "number_of_sockets",
    "Value": "1"
  },
  {
    "Name": "cores_per_socket",
    "Value": "1"
  },
  {
    "Name": "number_of_cpus",
    "Value": "None"
  }
  ]
}

```



```

    },
    {
      "Name": "system_memory",
      "Value": "488.35 MiB"
    }
  ],
  "Created": "2019-08-27T15:34:38.504298Z",
  "Display_name": "from-inv1",
  "Fact_count": 20,
  "Id": "UUID",
  "Updated": "2019-08-27T15:34:38.504298Z"
}

```

### 3.3. 新規作成用の既存の文献のコピー

GET メソッドおよび POST メソッドを使用して、既存のベースラインをコピーして、API 経由で新しいベースラインを作成できます。

1. GET を使用して既存のベースライン ID の一覧を取得します。

```
$ curl -X GET https://username:password@cloud.redhat.com/api/system-baseline/v1/baselines-H 'content-type: application/json'
```

2. コピー元となるベースライン ID を選択して、新しいベースラインを作成します。
3. POST メソッドを使用して選択したベースライン ID を使用して新しいベースラインを作成します。新しい表示名を指定する必要があります。指定しないと 400 エラーが発生します。

```
$ curl -X POST https://username:password@cloud.redhat.com/api/system-baseline/v1/baselines/<ID>?display_name=copied-from-baseline-name
```

### 3.4. 新規ベースラインの作成への古いシステムプロファイルのコピー

過去のシステムプロファイルをコピーして、API 経由で新しいベースラインを作成できます。

**create-from-hsp.json** ファイルの例を以下に示します。

```

{"hsp_uuid": "<UUID>",
 "display_name": "from-hsp1"}

```

#### 要求:

```
$ curl -k -X POST -d @create-from-hsp.json
https://username:password@cloud.redhat.com/api/system-baseline/v1/baselines-H 'content-type:
application/json'
```

#### 応答:

読みやすさを向上するため、一部のファクトが応答から削除されていることに注意してください。

```

{
[
{

```

```

    "Name": "number_of_sockets",
    "Value": "1"
  },
  {
    "Name": "cores_per_socket",
    "Value": "1"
  },
  {
    "Name": "number_of_cpus",
    "Value": "None"
  },
  {
    "Name": "system_memory",
    "Value": "488.35 MiB"
  }
],
"Created": "2019-08-27T15:34:38.504298Z",
"Display_name": "from-inv1",
"Fact_count": 20,
"HSP_uuid": "UUID",
"Updated": "2019-08-27T15:34:38.504298Z"
}

```

### 3.5. ベースラインのソート

**order\_by** パラメーターおよび **order\_how** パラメーターを追加して GET メソッドを使用することでベースラインをソートできます。

- **order\_by** パラメーターは、**display\_name**、**created\_on**、**updated** の値を受け入れます。
- **order\_how** パラメーターは、結果をソートするために **ASC** (ascending) および **DESC** (descending) の値を受け入れます。

デフォルトでは、結果は表示名のアルファベット順でソートされます。

要求:

```

$ curl -X GET https://username:password@cloud.redhat.com/api/system-baseline/v1/baselines?
order_by=created_on&order_how=DESC

```

### 3.6. ベースラインの編集

**curl** コマンドを使用して、API 経由でベースラインを編集できます。

```

$ curl -u username:password https://cloud.redhat.com/api/system-baseline/v1$/baselines/[uuid]

```

### 3.7. ベースラインのエクスポート

GET メソッドを使用して、API 経由でベースラインを CSV ファイルとしてエクスポートできます。

```

$ curl -X GET https://username:password@cloud.redhat.com/api/system-baseline/v1/baselines -H
'accept: text/csv'

```

### 3.8. ベースラインの削除

API を使用してベースラインを削除できます。

以下の curl コマンドを使用して、API 経由でベースラインを削除します。

```
$ curl -u username:password https://cloud.redhat.com/api/system-baseline/v1/baselines/[ID] -X  
DELETE
```

## 第4章 過去のシステムプロファイル API

過去のシステムプロファイルは、Insights for RHEL インベントリーのシステムプロファイルの時点をキャプチャーします。プロファイルは、特定システムのシステムプロファイルの一覧を返します。

過去のシステムプロファイルには、以下の2つの主要な方法があります。

- システムメソッド
- プロファイルメソッド

### 4.1. システムメソッド

システムメソッドは、**uuid** パラメーターで指定したシステムの過去のシステムプロファイルの一覧を返します。

以下の **curl** コマンドを使用して、システム **uuid** の過去のシステムプロファイルをすべて取得します。

```
$ curl -u username:password https://cloud.redhat.com/api/historical-system-profiles/v1/system/[ID]
```

各プロファイルに対して、サービスは以下を返します。

- **captured\_date**: システムプロファイルがキャプチャーされたタイミングを示す日付
- **id**: プロファイル **uuid** です。
- **system\_id**: システムの **uuid**。要求で指定されたものと同じです。

以下は、想定される Systems メソッドの出力の一部の例です。

```
{
  "data": [
    {
      "profiles": [
        {
          "captured_date": "2020-10-28T06:23:25+00:00",
          "id": "35054dcf-da10-489e-b383-8580511c8c10",
          "system_id": "563f782e-6266-41f0-a761-0e0eab29463b"
        },
        {
          "captured_date": "2020-10-27T18:43:28+00:00",
          "id": "472ad292-e2b1-40ac-b514-502df11df765",
          "system_id": "563f782e-6266-41f0-a761-0e0eab29463b"
        },
        {
          "captured_date": "2020-10-27T06:14:13+00:00",
          "id": "89235117-e9f5-4a2c-86d4-c3e908a392a7",
          "system_id": "563f782e-6266-41f0-a761-0e0eab29463b"
        },
        {
          "captured_date": "2020-10-26T07:34:17+00:00",
```

```

        "id": "6652ba31-f5e5-45c6-ae59-8f8d51181358",
        "system_id": "563f782e-6266-41f0-a761-0e0eab29463b"
    },
    {
        "captured_date": "2020-10-25T06:40:18+00:00",
        "id": "b66e8bb8-916c-409b-96d8-93119944e71d",
        "system_id": "563f782e-6266-41f0-a761-0e0eab29463b"
    },
    {
        "captured_date": "2020-10-24T06:23:19+00:00",
:context: {parent-context}
        "id": "b7dadf95-90ec-4289-b50a-6c8d19124ccf",
        "system_id": "563f782e-6266-41f0-a761-0e0eab29463b"
    },
    {
        "captured_date": "2020-10-23T06:34:14+00:00",
        "id": "3c9ce38d-22da-4d06-876d-35133eb5959e",
        "system_id": "563f782e-6266-41f0-a761-0e0eab29463b"
    },
    {
        "captured_date": "2020-10-22T05:59:09+00:00",
        "id": "18cd803d-3f1f-4ce7-9bbf-6a6585dc9c03",
        "system_id": "563f782e-6266-41f0-a761-0e0eab29463b"
    }
]
}

```

## 4.2. プロファイルメソッド

プロファイルエンドポイントは、**uuid** パラメーターで指定した各プロファイル ID のシステムプロファイル情報を返します。API は、コンマ区切りの過去のシステムプロファイル ID を1つ以上取得し、各 ID のシステムプロファイル情報を返します。

以下の curl コマンドを使用して、コンマ区切りのパラメーターとして渡される過去のシステムプロファイルを取得します。

```
$ curl -u username:password https://cloud.redhat.com/api/historical-system-profiles/v1/profiles/[ID1],[ID2]
```

以下は、確認できる Profile メソッド出力の一部の例です。

```

{
  "data": [
    {
      "account": "ACCOUNTID",
      "captured_date": "2020-10-27T18:43:28+00:00",
      "created": "2020-10-27T18:43:48.340185+00:00",
      "display_name": "dsmith-rhel82kvm",
      "id": "472ad292-e2b1-40ac-b514-502df11df765",
      "inventory_id": "563f782e-6266-41f0-a761-0e0eab29463b",

```

```
"system_profile": {
  "arch": "x86_64",
  "bios_release_date": "04/01/2014",
  "bios_vendor": "SeaBIOS",
  "bios_version": "1.12.0-2.fc30",
  "captured_date": "2020-10-27T18:43:28+00:00",
  "cores_per_socket": 2,

  "cpu_flags": [],

:context: {parent-context}      "display_name": "dsmith-rhel82kvm",

  "dnf_modules": [],

  "enabled_services": [],
  "fqdn": "dsmith-rhel82kvm",
  "id": "472ad292-e2b1-40ac-b514-502df11df765",
  "infrastructure_type": "virtual",
  "infrastructure_vendor": "kvm",
  "insights_client_version": "3.0.12-0",
  "insights_egg_version": "3.0.162-1",

  "installed_packages": [],

  "installed_products": [

  "installed_services": [],

  "kernel_modules": [],
  "last_boot_time": "2020-05-01T02:11:28",

  "network_interfaces": [],
  "number_of_cpus": 8,
  "number_of_sockets": 4,
  "os_kernel_version": "4.18.0",
  "os_release": "8.2",

  "running_processes": [],
  "satellite_managed": false,
  "selinux_config_file": "enforcing",
  "selinux_current_mode": "enforcing",
  "system_memory_bytes": 1914204160,

  "tags": [],
  "tuned_profile": "virtual-guest",

  "yum_repos": []
}
}
]
```