Red Hat Developer Hub 1.8

Telemetry data collection and analysis

Collecting and analyzing web analytics and system observability data to enhance Red Hat Developer Hub experience

Last Updated: 2026-01-22

Red Hat Developer Hub 1.8 Telemetry data collection and analysis

Collecting and analyzing web analytics and system observability data to enhance Red Hat Developer Hub experience

Legal Notice

Abstract

As a Red Hat Developer Hub (RHDH) administrator, you can collect and analyze two distinct types of telemetry data: web analytics using Segment and system observability using OpenTelemetry, to enhance the Red Hat Developer Hub experience.

Table of Contents

CHAPTER 1. TELEMETRY DATA COLLECTION AND ANALYSIS

The telemetry data collection feature helps in collecting and analyzing the telemetry data to improve your experience with Red Hat Developer Hub. This feature is enabled by default.

Red Hat collects and analyzes the following data:

Web Analytics

Web Analytics use the Segment tool. It is the tracking of user behavior and interactions with Red Hat Developer Hub. Specifically, it tracks the following:

- Events of page visits and clicks on links or buttons.
- System-related information, for example, locale, time zone, user agent including browser and operating system details.
- Page-related information, for example, title, category, extension name, URL, path, referrer, and search parameters.
- Anonymized IP addresses, recorded as **0.0.0.0**.
- Anonymized username hashes, which are unique identifiers used solely to identify the number of unique users of the RHDH application.

System Observability

System Observability uses the OpenTelemetry tool. It is the tracking of the performance of the RHDH. Specifically, it tracks the following metrics:

- Key system metrics such as CPU usage, memory usage, and other performance indicators.
- Information about system components, such as the locale, time zone, and user agent (including details of the browser and operating system).
- Traces and logs monitor system processes, allowing you to troubleshoot potential issues impacting the performance of RHDH.

With RHDH, you can customize the *Web Analytics* and *System Observability* configuration based on your needs.

CHAPTER 2. DISABLING TELEMETRY DATA COLLECTION IN RHDH

To disable telemetry data collection, you must disable the **analytics-provider-segment** plugin either using the Helm Chart or the Red Hat Developer Hub Operator configuration.

As an administrator, you can disable the telemetry data collection feature based on your needs. For example, in an air-gapped environment, you can disable this feature to avoid needless outbound requests affecting the responsiveness of the RHDH application.

2.1. DISABLING TELEMETRY DATA COLLECTION USING THE OPERATOR

You can disable the telemetry data collection feature by using the Operator.

Prerequisites

- You have logged in as an administrator in the OpenShift Container Platform web console.
- You have installed Red Hat Developer Hub on OpenShift Container Platform using the Operator.

Procedure

1. Perform one of the following steps:
   - If you have created the **dynamic-plugins-rhdh** ConfigMap file and not configured the **analytics-provider-segment** plugin, add the plugin to the list of plugins and set its **plugins.disabled** parameter to **true**.
   - If you have created the **dynamic-plugins-rhdh** ConfigMap file and configured the **analytics-provider-segment** plugin, search the plugin in the list of plugins and set its **plugins.disabled** parameter to **true**.
   - If you have not created the ConfigMap file, create it with the following YAML code:

```
kind: ConfigMap
apiVersion: v1
metadata:
  name: dynamic-plugins-rhdh
data:
  dynamic-plugins.yaml: |
    includes:
      - dynamic-plugins.default.yaml
    plugins:
      - package: './dynamic-plugins/dist/backstage-community-plugin-analytics-provider-segment'
        disabled: true
```

2. Set the value of the **dynamicPluginsConfigMapName** parameter to the name of your **dynamic-plugins-rhdh** config map in your **Backstage** custom resource:

```
# ...
```

```
spec:
  application:
    dynamicPluginsConfigMapName: dynamic-plugins-rhdh
# ...
```

3. Save the configuration changes.

2.2. DISABLING TELEMETRY DATA COLLECTION USING THE HELM CHART

You can disable the telemetry data collection feature by using the Helm Chart.

Prerequisites

- You have logged in as an administrator in the OpenShift Container Platform web console.
- You have installed Red Hat Developer Hub on OpenShift Container Platform using the Helm Chart.

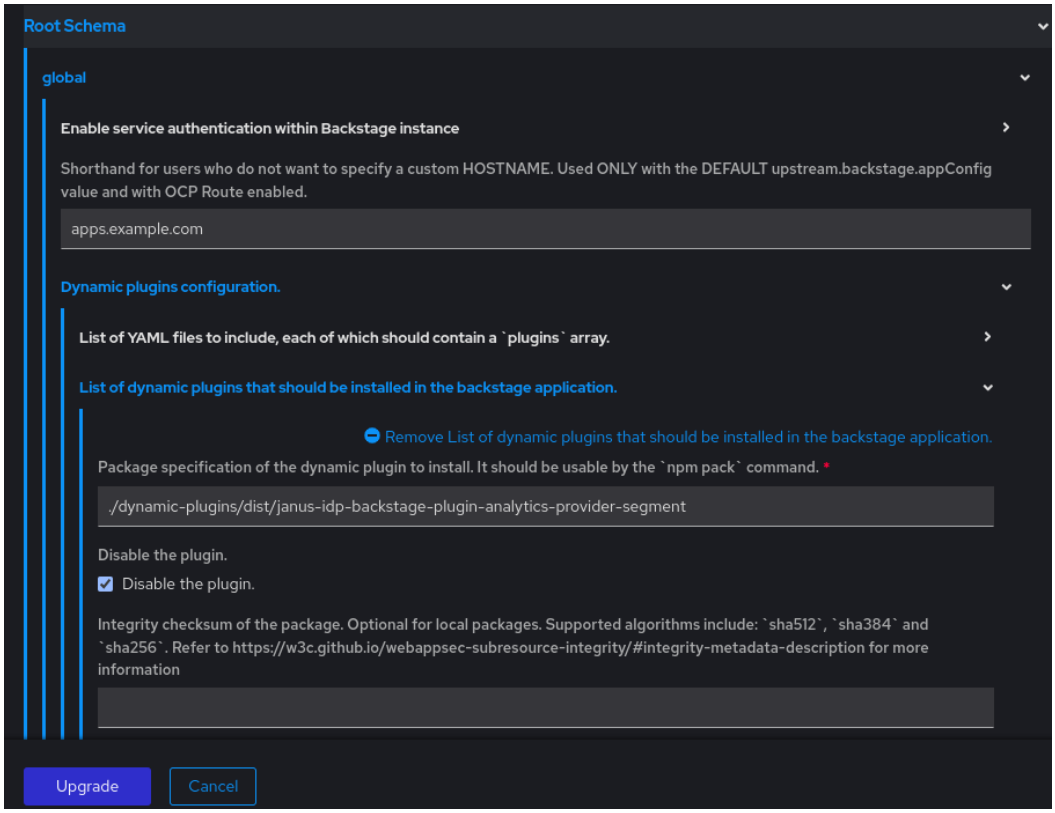
Procedure

1. In the **Developer** perspective of the OpenShift Container Platform web console, go to the **Helm** view to see the list of Helm releases.
2. Click the **overflow** menu on the Helm release that you want to use and select **Upgrade**.

NOTE

You can also create a new Helm release by clicking the **Create** button and edit the configuration to disable telemetry.

3. Use either the **Form** view or **YAML** view to edit the Helm configuration:
   - Using **Form view**
     a. Expand **Root Schema → global → Dynamic plugins configuration. → List of dynamic plugins that should be installed in the backstage application**.
     b. Click the **Add list of dynamic plugins that should be installed in the backstage application.** link.
     c. Perform one of the following steps:
        - If you have not configured the plugin, add the following value in the **Package specification of the dynamic plugin to install. It should be usable by the npm pack command.** field: **./dynamic-plugins/dist/backstage-community-plugin-analytics-provider-segment**



- If you have configured the plugin, find the **Package specification of the dynamic plugin to install. It should be usable by the npm pack command.** field with the **./dynamic-plugins/dist/backstage-community-plugin-analytics-provider-segment** value.

d. Select the **Disable the plugin** checkbox.

e. Click **Upgrade**.

- Using **YAML view**

a. Perform one of the following steps:

- If you have not configured the plugin, add the following YAML code in your **values.yaml** Helm configuration file:

```
# ...
global:
  dynamic:
    plugins:
      - package: './dynamic-plugins/dist/backstage-community-plugin-analytics-provider-segment'
        disabled: true
# ...
```

- If you have configured the plugin, search it in your Helm configuration and set the value of the **plugins.disabled** parameter to **true**.

b. Click **Upgrade**.

CHAPTER 3. ENABLING TELEMETRY DATA COLLECTION IN RHDH

The telemetry data collection feature is enabled by default. However, if you have disabled the feature and want to re-enable it, you must enable the **analytics-provider-segment** plugin either by using the Helm Chart or the Red Hat Developer Hub Operator configuration.

3.1. ENABLING TELEMETRY DATA COLLECTION USING THE OPERATOR

You can enable the telemetry data collection feature by using the Operator.

Prerequisites

- You have logged in as an administrator in the OpenShift Container Platform web console.
- You have installed Red Hat Developer Hub on OpenShift Container Platform using the Operator.

Procedure

1. Perform one of the following steps:
   - If you have created the **dynamic-plugins-rhdh** ConfigMap file and not configured the **analytics-provider-segment** plugin, add the plugin to the list of plugins and set its **plugins.disabled** parameter to **false**.
   - If you have created the **dynamic-plugins-rhdh** ConfigMap file and configured the **analytics-provider-segment** plugin, search the plugin in the list of plugins and set its **plugins.disabled** parameter to **false**.
   - If you have not created the ConfigMap file, create it with the following YAML code:

```
kind: ConfigMap
apiVersion: v1
metadata:
  name: dynamic-plugins-rhdh
data:
  dynamic-plugins.yaml: |
    includes:
      - dynamic-plugins.default.yaml
    plugins:
      - package: './dynamic-plugins/dist/backstage-community-plugin-analytics-provider-segment'
        disabled: false
```

2. Set the value of the **dynamicPluginsConfigMapName** parameter to the name of your **dynamic-plugins-rhdh** config map in your **Backstage** custom resource:

```
# ...
spec:
  application:
```

```
    dynamicPluginsConfigMapName: dynamic-plugins-rhdh
# ...
```

3. Save the configuration changes.

3.2. ENABLING TELEMETRY DATA COLLECTION USING THE HELM CHART

You can enable the telemetry data collection feature by using the Helm Chart.

Prerequisites

- You have logged in as an administrator in the OpenShift Container Platform web console.
- You have installed Red Hat Developer Hub on OpenShift Container Platform using the Helm Chart.

Procedure

1. In the **Developer** perspective of the OpenShift Container Platform web console, go to the **Helm** view to see the list of Helm releases.
2. Click the **overflow** menu on the Helm release that you want to use and select **Upgrade**.

NOTE

You can also create a new Helm release by clicking the **Create** button and edit the configuration to enable telemetry.

3. Use either the **Form** view or **YAML** view to edit the Helm configuration:
   - Using **Form view**
     a. Expand **Root Schema → global → Dynamic plugins configuration. → List of dynamic plugins that should be installed in the backstage application**.
     b. Click the **Add list of dynamic plugins that should be installed in the backstage application.** link.
     c. Perform one of the following steps:
        - If you have not configured the plugin, add the following value in the **Package specification of the dynamic plugin to install. It should be usable by the npm pack command.** field: **./dynamic-plugins/dist/backstage-community-plugin-analytics-provider-segment**
        - If you have configured the plugin, find the **Package specification of the dynamic plugin to install. It should be usable by the npm pack command.** field with the **./dynamic-plugins/dist/backstage-community-plugin-analytics-provider-segment** value.
     d. Clear the **Disable the plugin** checkbox.
     e. Click **Upgrade**.

- Using **YAML view**

  a. Perform one of the following steps:

  - If you have not configured the plugin, add the following YAML code in your Helm configuration file:

```
# ...
global:
  dynamic:
    plugins:
      - package: './dynamic-plugins/dist/backstage-community-plugin-analytics-provider-segment'
        disabled: false
# ...
```

  - If you have configured the plugin, search it in your Helm configuration and set the value of the **plugins.disabled** parameter to **false**.

  b. Click **Upgrade**.

CHAPTER 4. CUSTOMIZING SEGMENT SOURCE

The **analytics-provider-segment** plugin sends the collected web analytics data to Red Hat by default. However, you can configure a new Segment source that receives web analytics data based on your needs. For configuration, you need a unique Segment write key that points to the Segment source.

NOTE

Create your own web analytics data collection notice for your application users.

4.1. CUSTOMIZING SEGMENT SOURCE USING THE OPERATOR

You can configure integration with your Segment source by using the Red Hat Developer Hub Operator.

Prerequisites

- You have logged in as an administrator in the OpenShift Container Platform web console.
- You have installed Red Hat Developer Hub on OpenShift Container Platform using the Operator.

Procedure

1. Add the following YAML code in your **Backstage** custom resource (CR):

```
# ...
spec:
  application:
    extraEnvs:
      envs:
        - name: SEGMENT_WRITE_KEY
          value: <segment_key> 1
# ...
```

1 Replace **<segment_key>** with a unique identifier for your Segment source.

2. Save the configuration changes.

4.2. CUSTOMIZING SEGMENT SOURCE USING THE HELM CHART

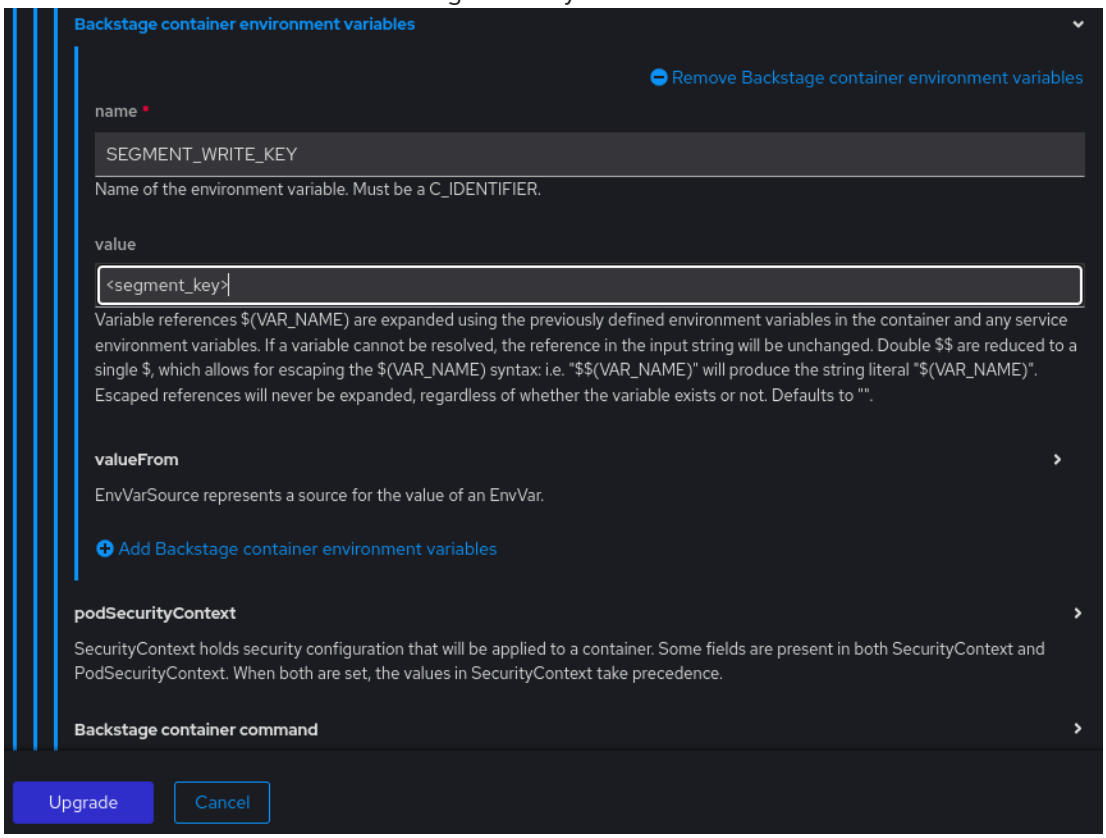
You can configure integration with your Segment source by using the Red Hat Developer Hub Helm Chart.

Prerequisites

- You have logged in as an administrator in the OpenShift Container Platform web console.
- You have installed Red Hat Developer Hub on OpenShift Container Platform using the Helm Chart.

Procedure

1. In the **Developer** perspective of the OpenShift Container Platform web console, go to the **Helm** view to see the list of Helm releases.
2. Click the **overflow** menu on the Helm release that you want to use and select **Upgrade**.
3. Use either the **Form** view or **YAML** view to edit the Helm configuration:
   - Using **Form view**
     a. Expand **Root Schema → Backstage Chart Schema → Backstage Parameters → Backstage container environment variables**.
     b. Click the **Add Backstage container environment variables** link.
     c. Enter the name and value of the Segment key.



     d. Click **Upgrade**.
   - Using **YAML view**
     a. Add the following YAML code in your Helm configuration file:

```
# ...
upstream:
  backstage:
    extraEnvVars:
      - name: SEGMENT_WRITE_KEY
        value: <segment_key> 1
# ...
```

     1 Replace **<segment_key>** with a unique identifier for your Segment source.

     b. Click **Upgrade**.