

The software described in this documentation is either no longer supported or is in extended support.  
Oracle recommends that you upgrade to a current supported release.

# Oracle® Cloud Native Environment Monitoring and Visualization for Release 1.3

**ORACLE®**

F50648-02

April 2022

The software described in this documentation is either no longer supported or is in extended support.  
Oracle recommends that you upgrade to a current supported release.

---

## Oracle Legal Notices

Copyright © 2020, 2022, Oracle and/or its affiliates.

---

## Table of Contents

Preface .....	v
1 Introduction to Monitoring and Visualization .....	1
1.1 What Is Grafana? .....	1
1.2 About Grafana .....	1
1.3 Grafana Components .....	2
1.3.1 Data Source .....	2
1.3.2 Query Editor .....	2
1.3.3 Panel .....	2
1.3.4 Dashboard .....	2
1.3.5 User .....	2
2 Installing Grafana .....	5
2.1 Deploying Grafana .....	5
3 Using Grafana .....	7
3.1 Getting the Grafana IP Address and Port Number .....	7
3.2 Accessing the Grafana Console .....	8
3.3 Creating a Dashboard .....	8



## Preface

This document contains information about using the monitoring and visualization tools available with Oracle Cloud Native Environment.

Document generated on: 2022-04-29 (revision: 1232)

## Audience

This document is written for system administrators and developers who want to use the monitoring and visualization tools available with Oracle Cloud Native Environment. It is assumed that readers have a general understanding of the Oracle Linux operating system and container concepts.

## Related Documents

The latest version of this document and other documentation for this product are available at:

<https://docs.oracle.com/en/operating-systems/olcne/>

## Conventions

The following text conventions are used in this document:

Convention	Meaning
<b>boldface</b>	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
<i>italic</i>	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
<code>monospace</code>	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

## Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at <https://www.oracle.com/corporate/accessibility/>.

For information about the accessibility of the Oracle Help Center, see the Oracle Accessibility Conformance Report at <https://www.oracle.com/corporate/accessibility/templates/t2-11535.html>.

## Access to Oracle Support for Accessibility

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit <https://www.oracle.com/corporate/accessibility/learning-support.html#support-tab>.

## Diversity and Inclusion

Oracle is fully committed to diversity and inclusion. Oracle respects and values having a diverse workforce that increases thought leadership and innovation. As part of our initiative to build a more inclusive culture

The software described in this documentation is either no longer supported or is in extended support.  
Oracle recommends that you upgrade to a current supported release.

## Diversity and Inclusion

---

that positively impacts our employees, customers, and partners, we are working to remove insensitive terms from our products and documentation. We are also mindful of the necessity to maintain compatibility with our customers' existing technologies and the need to ensure continuity of service as Oracle's offerings and industry standards evolve. Because of these technical constraints, our effort to remove insensitive terms is ongoing and will take time and external cooperation.

## Chapter 1 Introduction to Monitoring and Visualization

This chapter provides introductory information about Grafana, and contains information about the components in Grafana that are used to monitor and visualize the metrics that are stored in a time-series database.

### 1.1 What Is Grafana?

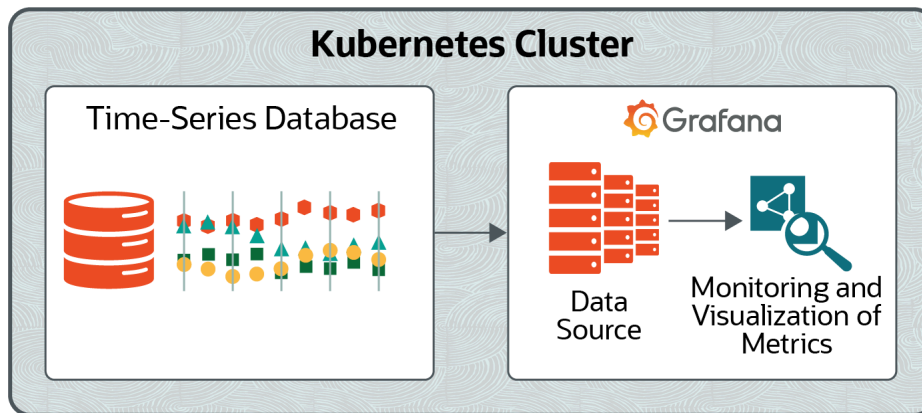
Grafana is an open-source platform that you can use to monitor the performance of your Kubernetes cluster. It contains dashboards that allow you to visualize the metrics of the cluster that are gathered and stored in an application such as Prometheus that records real-time metrics in a time-series database.

Grafana is a visualization tool for time-series data stored in a database. Time-series data is a sequence of values ordered by time. It measures how these values change over time. Examples of time-series data include self-driving cars collecting information about their local environments continually, homes monitoring themselves to regulate temperature or identify intruders, or a police department tracking its vehicles for efficiency purposes.

You use a data source to integrate Grafana with a time-series database such as Prometheus. A data source is Grafana's link to this type of database. By using a data source for Prometheus in Grafana, Grafana can retrieve, analyze, monitor, and visualize the metrics that are stored in a Prometheus time-series database.

The following architectural diagram illustrates how Grafana can use a data source to integrate with a time-series database such as Prometheus. As a result, Grafana can monitor and visualize the metrics that are stored in the database.

**Figure 1.1 Architectural Diagram for Integrating a Time-Series Database with Grafana**



### 1.2 About Grafana

Grafana is based on a stable release of the upstream Grafana project. Differences between Oracle versions of the software and upstream releases are limited to Oracle specific fixes and patches for specific bugs.

For upstream Grafana documentation, see the upstream documentation at:

<https://grafana.com/docs/>

For more information about Grafana, see the upstream documentation at:

<https://grafana.com>

## 1.3 Grafana Components

This section contains information about the components in Grafana that are used to monitor and visualize the metrics that are stored in a Prometheus database.

### 1.3.1 Data Source

Data sources are Grafana's links to the different storage databases that Grafana supports. Grafana contains out-of-the-box support to connect to many time-series databases (including Prometheus) that you can use to monitor and visualize the metrics in your Oracle Cloud Native Environment. Grafana refers to a connection to this type of database as a data source.

By adding a data source for a time-series database in Grafana, Grafana can retrieve, analyze, monitor, and visualize the metrics that are stored in the database.

### 1.3.2 Query Editor

Grafana has a query editor that exposes the capabilities of your data source and allows you to query the metrics that it contains. Grafana provides a custom query editor for each data source, including the data source used to integrate Grafana with Prometheus. You can use the query editor to structure queries that allow you to visualize the metrics that are stored in the Prometheus database.

### 1.3.3 Panel

The panel is the main element in Grafana used to visualize metrics from a time-series database such as Prometheus. Each type of panel has its own query editor that allows you to fine-tune the data that you want to visualize.

There are several panel types, including Graph, Singlestat, Table, Text, and Dashboard List. In this book, you use the Graph panel. This is the main panel type in Grafana and it provides a rich set of graphing options.

For more information about the other panel types, see the upstream documentation at:

<https://grafana.com/docs/grafana/latest/features/panels/panels/>

### 1.3.4 Dashboard

A dashboard is a grouping of panels prearranged into rows. A row is the divider between panels.

### 1.3.5 User

A user has an account in Grafana. A user is granted permissions in Grafana based on the following roles:

- **Admin:** An admin has superuser permissions in Grafana, and can do everything, including adding and editing data sources, generating queries for the data sources, and creating and modifying dashboards.
- **Editor:** An editor has limited permissions in Grafana. Although an editor can create and modify dashboards, they cannot create or edit data sources, or generate queries for the data sources.



The software described in this documentation is either no longer supported or is in extended support.  
Oracle recommends that you upgrade to a current supported release.

## User

---

- **Viewer:** A viewer has read-only access to the components in Grafana. For example, a viewer can view data sources and dashboards, but cannot modify them.

In this book, you access Grafana as an Admin. For more information about the Admin, Editor, and Viewer roles, see the upstream documentation at:

[https://grafana.com/docs/grafana/latest/permissions/organization\\_roles/](https://grafana.com/docs/grafana/latest/permissions/organization_roles/)



## Chapter 2 Installing Grafana

This chapter discusses installing Grafana.

### 2.1 Deploying Grafana

Grafana is deployed as part of the Istio module for Oracle Cloud Native Environment (the Istio module). This module is a fully featured service mesh for deploying microservices into Kubernetes clusters, and can handle most aspects of microservice management, including identity, authentication, certificate management, transport security, and metric gathering (scraping).

When you install the Istio module, it comes with default configurations. A default configuration provides everything that you want out of the box, including Prometheus and Grafana.

By installing the Istio module, Prometheus and Grafana are deployed automatically.

For more information about installing the Istio module, see [Service Mesh](#).



## Chapter 3 Using Grafana

When Grafana is deployed, a data source is configured for a Prometheus time-series database. A data source is Grafana's link to the database. Because this data source is configured, Grafana can retrieve and analyze the metrics of a Kubernetes cluster that are gathered and stored in the Prometheus database.

In this chapter, you learn how to:

- Get the IP address of the node on which Grafana is deployed and the port number that is reserved for Grafana. You need this information to access the Grafana console.
- Access the Grafana console.
- Create a dashboard in Grafana to monitor and visualize the metrics that are retrieved and analyzed from Prometheus through the data source.
- Formulate a query for the dashboard to fine-tune how the metrics for Prometheus appear in Grafana.

### 3.1 Getting the Grafana IP Address and Port Number

In this section, you get the IP address of the node on which Grafana is deployed and the port number for Grafana.

1. At the prompt of the machine where you installed the Istio module, enter the following command:

```
kubectl -n istio-system get svc grafana
```

There is no NodePort set up for the port on which the Grafana service is running. In this example, the port that is reserved for Grafana is [9090](#).

NAME	TYPE	CLUSTER-IP	EXTERNAL-IP	PORT(S)	AGE
grafana	ClusterIP	10.104.245.12	<none>	9090/TCP	6d1h

A NodePort is an open port on every node of your Kubernetes cluster. By opening a NodePort for Grafana, Kubernetes transparently routes incoming traffic on the NodePort to the Grafana service. The NodePort is assigned from a pool of cluster-configured NodePort ranges (typically 30000–32767).

2. Enter the following command to set up a NodePort for the Grafana service:

```
kubectl patch svc grafana -n istio-system -p '{"spec":{"type":"NodePort"}}'
```

3. Verify that you see the [service/grafana patched](#) status message.

You can now connect to Grafana via the NodePort.

4. Enter the following command again:

```
kubectl -n istio-system get svc grafana
```

This time, you can see that a NodePort is set up for the Grafana container. You can use this NodePort to connect to the service.

NAME	TYPE	CLUSTER-IP	EXTERNAL-IP	PORT(S)	AGE
grafana	ClusterIP	10.104.245.12	<none>	9090/30921	6d1h

In this example, the NodePort is [30921](#), which maps to the service port of [9090](#).

5. Enter the following command to get the IP address of either a control plane node or the IP address of the worker node on which the Grafana container is running:

```
ip addr
```

Make a note of the IP address and NodePort that you obtained in this procedure because you need these values to access the Grafana console.

## 3.2 Accessing the Grafana Console

As part of deploying Grafana, a user account is created, and the Admin role is assigned to the account. This account has superuser permissions in Grafana and can create dashboards and generate queries for the dashboards.

In this section, you access the Grafana console.

1. Open a web browser.
2. In the **Address** field, enter `http://[IP_address]:[NodePort]`

You obtained the IP address and NodePort in [Section 3.1, "Getting the Grafana IP Address and Port Number"](#).

After you provide the Grafana URL, you are taken to the **Home Dashboard** page. This page provides a workflow to help you configure Grafana, including creating a dashboard in Grafana and generating a query for the dashboard.

## 3.3 Creating a Dashboard

In Grafana, a dashboard is how you monitor and visualize the metrics that are retrieved from a data source such as Prometheus. The dashboard is a grouping of one or more panels, prearranged into rows.

In this section, you create a dashboard and select the Graph panel for it. The Graph panel is the main panel in Grafana, and has a rich set of graphing options.

As part of creating your dashboard, you use the query editor feature to formulate a query for the dashboard. A query is used to fine-tune how the metrics for Prometheus appear in Grafana. For this procedure, you specify the `prometheus_engine_query_duration_seconds` query, which allows Grafana to graph the metrics that Prometheus collects about itself.

1. In the **Home Dashboard** page, click the **Create your first dashboard** icon.
2. In the **New dashboard** page, click the **Choose Visualization** icon.
3. In the **Visualization** field, enter `Graph`.
4. Click the **Queries** icon. This icon appears to the left of the **Visualization** field and looks like a database.
5. In the **Query** field, select the Prometheus data source that was added as part of deploying Grafana.
6. In the query editor field, which appears to the right of the **Metrics** drop-down menu, enter the `prometheus_engine_query_duration_seconds` query, and then click the **Disable/enable query** button. This button appears to the right of the query that you entered and looks like an eye.
7. Verify that data associated with the metric appears, and then click **Save dashboard**.

8. In the **Save As** dialog box, give the dashboard a new name, select a folder location for the dashboard, and then click **Save**.
9. Verify that you see the **Dashboard saved** status message and a graph that displays information about the metric that you entered.

