



Working with Prometheus and Grafana Using Helm

Last Updated : 19 Jan, 2023

Pre-requisite: [HELM Package Manager](#)

Helm is a package manager for Kubernetes that allows you to install, upgrade, and manage applications on your Kubernetes cluster. With Helm, you can define, install, and upgrade your application using a single configuration file, called a Chart. Charts are easy to create, version, share, and publish, so you can collaborate with your team and the larger Kubernetes community.

There are many charts available in the Helm stable repository, which is maintained by the Helm community, and you can also find charts from other sources or create your own custom charts. Helm makes it easy to automate the deployment and management of applications on your Kubernetes cluster, so you can focus on building great software.

Prometheus

Prometheus is an open-source monitoring system that is especially well-suited for cloud-native environments, like Kubernetes. It can monitor the performance of your applications and services, and alert you if there are any issues. It has a powerful query language that allows you to analyze the data it collects, and it also has a rich set of integrations with other tools and systems. For example, you can use Prometheus to monitor the health of your Kubernetes cluster, and use its integration with Grafana to visualize the data it collects.

We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#).

Got It !

performance of your applications and services, and use its integrations with tools like PagerDuty to send alerts to the appropriate on-call personnel.

In summary, Prometheus is a powerful monitoring tool that is well-suited for cloud-native environments, and it has a wide range of integrations with other tools and systems to help you monitor and manage your applications and services.

Prometheus Architecture

Prometheus is a monitoring system that consists of the following components:

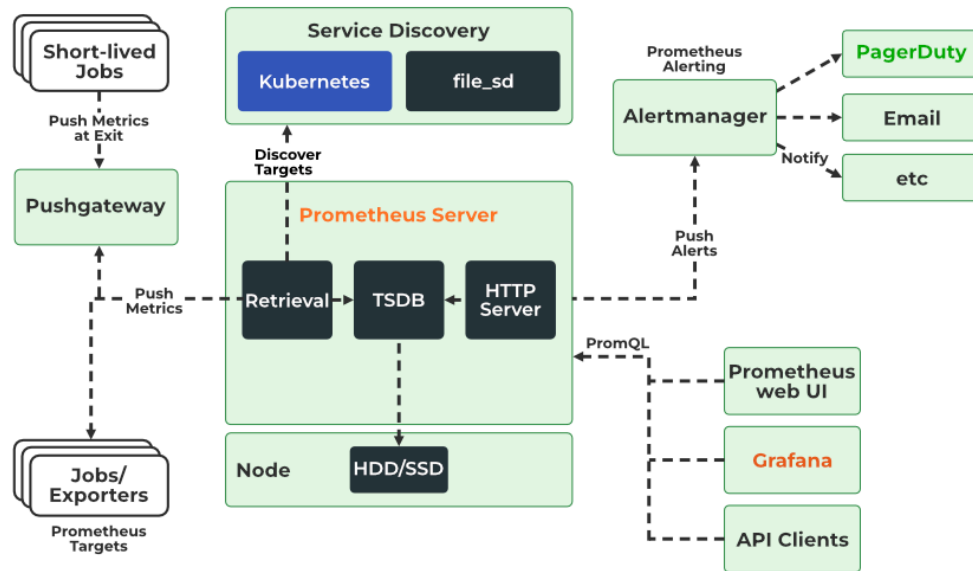
- A main server that scrapes and stores time series data
- A query language called PromQL is used to retrieve and analyze the data
- A set of exporters that are used to collect metrics from various systems and applications
- A set of alerting rules that can trigger notifications based on the data
- An alert manager that handles the routing and suppression of alerts

The Prometheus server stores the data in a time series database, and it also provides a web interface that you can use to query and analyze the data using PromQL. The web interface also includes a dashboard builder that allows you to create custom dashboards to visualize your data.

Prometheus also includes a built-in alert manager that can send notifications based on the data it collects. You can define alerting rules that specify when an alert should be triggered, and the alert manager will route and suppress the alerts as needed.

In summary, Prometheus is a powerful monitoring system that is

We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#).



Grafana

Grafana is an open-source data visualization and monitoring platform that allows you to create dashboards to visualize your data and metrics. It is a popular choice for visualizing time series data, and it integrates with a wide range of data sources, including Prometheus, Elasticsearch, and InfluxDB.

Grafana has a user-friendly interface that allows you to create and customize dashboards with panels that display your data in a variety of formats, including graphs, gauges, and tables. You can also use Grafana to set up alerts that trigger notifications when certain conditions are met.

In addition to its core features, Grafana has a rich ecosystem of plugins and integrations that extend its functionality. For example, you can use Grafana to integrate with other tools and services, such as Slack or PagerDuty, to receive alerts and notifications. Grafana is a powerful tool for visualizing and monitoring your data and metrics, and it is widely used in a variety of industries and contexts.

Why Prometheus and Grafana?

Prometheus and Grafana are often used together because they

We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#).

time series data. It has a flexible query language and a wide range of integrations with other tools and systems.

Grafana is a data visualization and monitoring platform that allows you to create interactive dashboards to visualize and analyze your data. It integrates with a wide range of data sources, including Prometheus, and it has a user-friendly interface that makes it easy to create and customize dashboards.

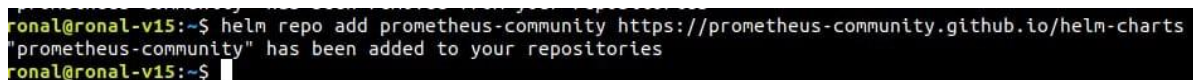
Together, Prometheus and Grafana provide a complete solution for monitoring and visualizing the performance of your applications and infrastructure. Prometheus can collect and store the data, and Grafana can provide visualizations and alerts to help you understand and act on that data. This can be especially useful in dynamic, cloud-native environments like Kubernetes, where you need a flexible and scalable solution for monitoring and managing your applications and services.

Installation Steps:

Pre-requisite: Install HELM

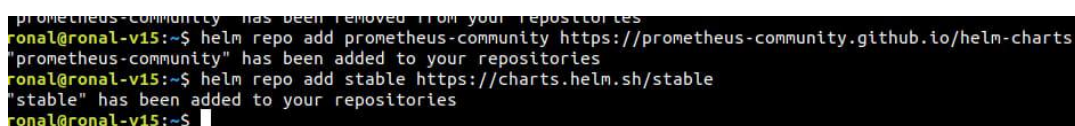
Step 1: Add Repositories. Add the following helm repositories by the commands below.

```
$ helm repo add prometheus-community https://prometheus-community.github.io/helm-charts
```



```
ronal@ronal-v15:~$ helm repo add prometheus-community https://prometheus-community.github.io/helm-charts
"prometheus-community" has been added to your repositories
ronal@ronal-v15:~$
```

```
$ helm repo add stable https://charts.helm.sh/stable
```



```
prometheus-community has been removed from your repositories
ronal@ronal-v15:~$ helm repo add prometheus-community https://prometheus-community.github.io/helm-charts
"prometheus-community" has been added to your repositories
ronal@ronal-v15:~$ helm repo add stable https://charts.helm.sh/stable
"stable" has been added to your repositories
ronal@ronal-v15:~$
```

Step 2: Update Helm Repositories. After we add the helm repositories,

We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#).

```
ronal@ronal-v15:~$ helm repo update
Hang tight while we grab the latest from your chart repositories...
...Successfully got an update from the "prometheus-community" chart repository
...Successfully got an update from the "nginx-stable" chart repository
...Successfully got an update from the "stable" chart repository
Update Complete. 🎉Happy Helming!🎉
ronal@ronal-v15:~$
```

Step 3: Install Prometheus Kubernetes

```
$ helm install prometheus prometheus-community/kube-
prometheus-stack
```

```
ronal@ronal-v15:~$ helm install prometheus prometheus-community/kube-prometheus-stack
NAME: prometheus
LAST DEPLOYED: Wed Jan  4 22:29:18 2023
NAMESPACE: default
STATUS: deployed
REVISION: 1
NOTES:
kube-prometheus-stack has been installed. Check its status by running:
  kubectl --namespace default get pods -l "release=prometheus"

Visit https://github.com/prometheus-operator/kube-prometheus for instructions on how to create & configure Alertmanager
ronal@ronal-v15:~$
```

Step 4: Kubernetes Prometheus Port Forward

```
$ kubectl port-forward deployment/prometheus-grafana 3000
```

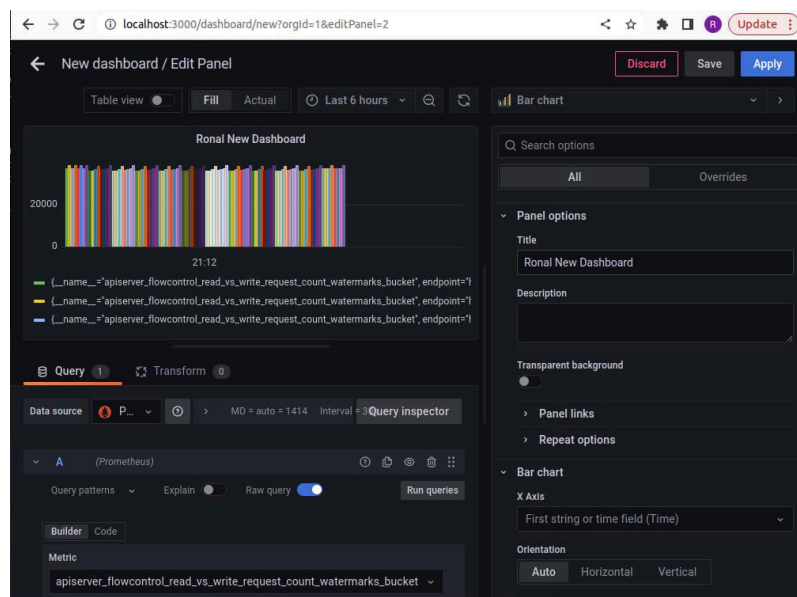
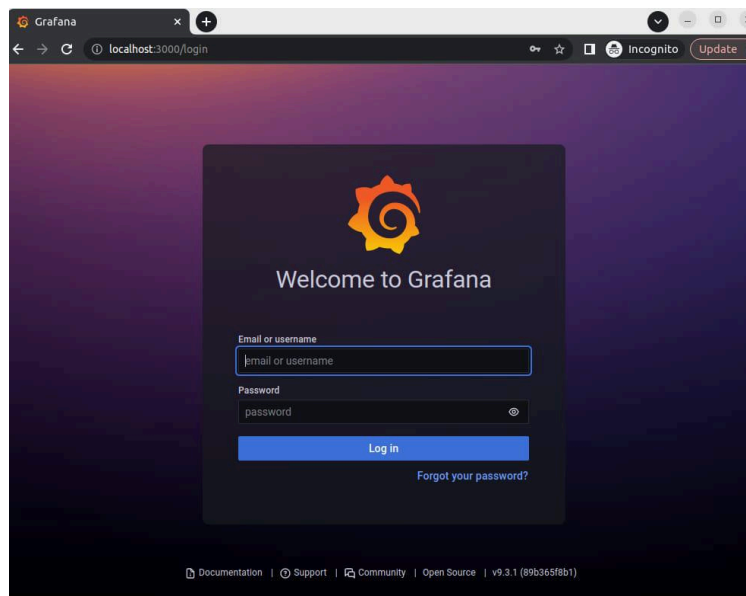
```
ronal@ronal-v15:~$ kubectl port-forward deployment/prometheus-grafana 3000
Forwarding from 127.0.0.1:3000 -> 3000
Forwarding from [::1]:3000 -> 3000
```

Step 5: Log in to Grafana.

Open your browser and type **localhost:3000**. Since you are logging in for the first time, you will have to use a default username and password. However, you can create new users and update passwords later. Use the default credentials for now

- **username:** admin
- **password:** prom-operator

We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#).

[Comment](#)[More info](#)[Advertise with us](#)

Next Article

Working with Monitoring and
Logging Services

Similar Reads

We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#).