[](https://devopscube.com/)

[](https://devopscube.com/)

* [Resources](https://devopscube.com/setup-grafana-kubernetes/)
* [Roadmaps](https://devopscube.com/setup-grafana-kubernetes/)
* [Newsletter](https://devopscube.com/newsletter/)
* [Certification Guides](https://devopscube.com/tag/devops-certification/)

[BEST DEVOPS DEALS](https://devopscube.com/black-friday-deals/)

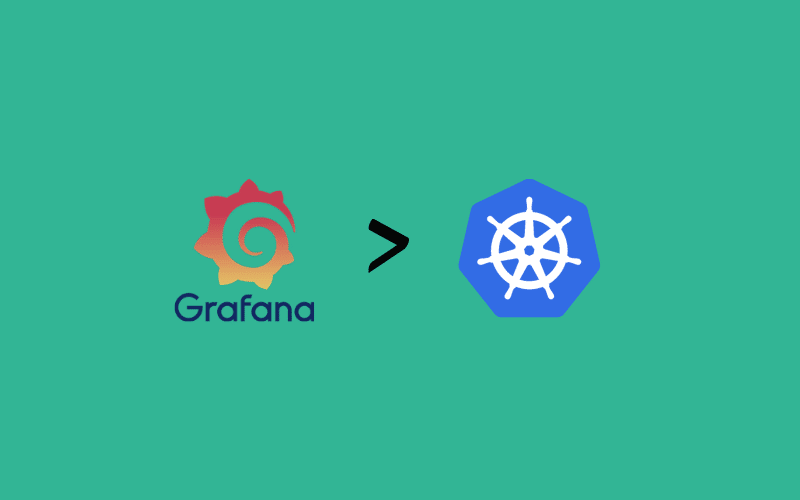
Top of Form

Bottom of Form

[KKUBERNETES](https://devopscube.com/category/kubernetes/)

How To Setup Grafana On Kubernetes

* by[**Bibin Wilson**](https://devopscube.com/author/bibinwilson/)
* January 29, 2022

[](https://devopscube.com/wp-content/uploads/2021/03/2.png)

Grafana is an open-source lightweight dashboard tool. It can be integrated with many data sources like Prometheus, [AWS cloud watch](https://devopscube.com/how-to-setup-and-push-serverapplication-logs-to-aws-cloudwatch/), Stackdriver, etc. Running Grafana on Kubernetes

In our previous posts, we have looked at the following.

1. [Setup Prometheus on Kubernetes](https://devopscube.com/setup-prometheus-monitoring-on-kubernetes/)
2. [Setup Kube State Metrics](https://devopscube.com/setup-kube-state-metrics/)
3. [Setup alert manager on Kubernetes](https://devopscube.com/alert-manager-kubernetes-guide/)

This tutorial explains how to run Grafana on [Kubernetes cluster](https://devopscube.com/setup-kubernetes-cluster-kubeadm/). Using Grafana you can simplify Kubernetes monitoring dashboards from Prometheus metrics.

**Grafana Kubernetes Manifests**

All the Kubernetes manifests (YAML files) used in this tutorial are [hosted on Github](https://github.com/bibinwilson/kubernetes-grafana) as well. You can clone it and use it for the setup.

git clone https://github.com/bibinwilson/kubernetes-grafana.git

**Deploy Grafana On Kubernetes**

Let’s look at the Grafana setup in detail.

**Step 1:** Create a file named grafana-datasource-config.yaml

vi grafana-datasource-config.yaml

Copy the following contents.

**Note:** The following data source configuration is for Prometheus. If you have more data sources, you can add more data sources with different YAMLs under the data section.

apiVersion: v1

kind: ConfigMap

metadata:

name: grafana-datasources

namespace: monitoring

data:

prometheus.yaml: |-

{

"apiVersion": 1,

"datasources": [

{

"access":"proxy",

"editable": true,

"name": "prometheus",

"orgId": 1,

"type": "prometheus",

"url": "http://prometheus-service.monitoring.svc:8080",

"version": 1

}

]

}

**Step 2:** Create the configmap using the following command.

kubectl create -f grafana-datasource-config.yaml

**Step 3:**Create a file named deployment.yaml

vi deployment.yaml

Copy the following contents on the file.

apiVersion: apps/v1

kind: Deployment

metadata:

name: grafana

namespace: monitoring

spec:

replicas: 1

selector:

matchLabels:

app: grafana

template:

metadata:

name: grafana

labels:

app: grafana

spec:

containers:

- name: grafana

image: grafana/grafana:latest

ports:

- name: grafana

containerPort: 3000

resources:

limits:

memory: "1Gi"

cpu: "1000m"

requests:

memory: 500M

cpu: "500m"

volumeMounts:

- mountPath: /var/lib/grafana

name: grafana-storage

- mountPath: /etc/grafana/provisioning/datasources

name: grafana-datasources

readOnly: false

volumes:

- name: grafana-storage

emptyDir: {}

- name: grafana-datasources

configMap:

defaultMode: 420

name: grafana-datasources

**Note:** This Grafana deployment **does not use a persistent volume**. If you restart the pod all changes will be gone. Use a persistent volume if you are deploying Grafana for your project requirements. It will persist all the configs and data that Grafana uses.

**Step 4:** Create the deployment

kubectl create -f deployment.yaml

**Step 5:**Create a service file named service.yaml

vi service.yaml

Copy the following contents. This will expose Grafana on NodePort 32000. You can also expose it [using ingress](https://devopscube.com/setup-ingress-kubernetes-nginx-controller/)or a Loadbalancer based on your requirement.

apiVersion: v1

kind: Service

metadata:

name: grafana

namespace: monitoring

annotations:

prometheus.io/scrape: 'true'

prometheus.io/port: '3000'

spec:

selector:

app: grafana

type: NodePort

ports:

- port: 3000

targetPort: 3000

nodePort: 32000

**Step 6:** Create the service.

kubectl create -f service.yaml

Now you should be able to access the Grafana dashboard **using any node IP** on port 32000. Make sure the port is allowed in the firewall to be accessed from your workstation.

http://<your-node-ip>:32000

You can also use port forwarding using the following command.

kubectl port-forward -n monitoring <grafana-pod-name> 3000 &

For example,

vagrant@dcubelab:~$ kubectl get po -n monitoring

NAME READY STATUS RESTARTS AGE

grafana-64c89f57f7-kjqrb 1/1 Running 0 10m

vagrant@dcubelab:~$ kubectl port-forward -n monitoring grafana-64c89f57f7-kjqrb 3000 &

You will be able to access Grafana a from http://localhost:3000

Use the following default username and password to log in. Once you log in with default credentials, it will prompt you to change the default password.

User: admin

Pass: admin

**Create Kubernetes Dashboards on Grafana**

**Creating a Kubernetes dashboard** from the Grafana template is pretty easy. There are many prebuilt Grafana templates available for Kubernetes. You can easily have prebuilt dashboards for ingress controllers, volumes, API servers, Prometheus metrics, and much more.

To know more, see [Grafana templates for Kubernetes monitoring](https://grafana.com/grafana/dashboards/?search=kubernetes)

Follow the steps given below to set up a Grafana dashboard to monitor kubernetes deployments.

**Step 1:** Get the template ID from [grafana public template.](https://grafana.com/grafana/dashboards/8588" \t "_blank) as shown below.

**Step 2:**Head over to the Grafana dashbaord and select the import option.

**Step 3:** Enter the dashboard ID you got in step 1

**Step 4:** Grafana will automatically fetch the template from the Grafana website. You can change the values as shown in the image below and click import.

**Note**: If you are behind the corporate firewall and cannot download the template using id, you can download the template JSON and paste the JSON in the text box to import it.

You should see the dashboard immediately.

When Grafana is used with Prometheus, it uses PromQL to query metrics from Prometheus. You can use the same PromQL Prometheus queries to **build custom dashboards on Grafana**.

**Conclusion**

Grafana is a very powerful tool when it comes to Kubernetes monitoring dashboards.

It is used by many organizations to monitor their Kubernetes workloads. With the wide range of pre-built templates, you can get started with the templates pretty quickly. What more can you ask for right?

Let me know how you are using Grafana in your organization.

Also, let me know if you want to add more information to this article.

* **TAGS:**
* [Grafana](https://devopscube.com/tag/grafana/)
* [Kubernetes](https://devopscube.com/tag/kubernetes/)
* [Prometheus](https://devopscube.com/tag/prometheus/)

[**Bibin Wilson**](https://devopscube.com/author/bibinwilson/)

Bibin Wilson is a cloud and DevOps consultant with over 10 years of IT experience. He has extensive hands-on experience with public cloud platforms, cloud hosting, Kubernetes and OpenShift deployments in production. He has authored over 300 tech tutorials, providing valuable insights to the DevOps community. His courses on techiescamp.com offer practical guidance and real-world examples for professionals aiming to excel in cloud, DevOps, and infrastructure automation.

VIEW COMMENTS (45)

**YOU MAY ALSO LIKE**

READ MORE

[HHOW TO GUIDES](https://devopscube.com/category/how-to/)

[Kubeconfig File Explained With Practical Examples](https://devopscube.com/kubernetes-kubeconfig-file/)

* by[**Bibin Wilson**](https://devopscube.com/author/bibinwilson/)
* January 25, 2024

In this blog, you will learn how to connect to a kubernetes cluster using the Kubeconfig file using…

READ MORE

[KKUBERNETES](https://devopscube.com/category/kubernetes/)

[How To Setup Kube State Metrics on Kubernetes](https://devopscube.com/setup-kube-state-metrics/)

* by[**Bibin Wilson**](https://devopscube.com/author/bibinwilson/)
* January 27, 2022

In this blog, we will look at what is Kube State Metrics and its setup on Kubernetes. This…

READ MORE

[HHOW TO GUIDES](https://devopscube.com/category/how-to/)

[How To Install Minikube: Comprehensive Tutorial for Beginners](https://devopscube.com/kubernetes-minikube-tutorial/)

* by[**Bibin Wilson**](https://devopscube.com/author/bibinwilson/)
* October 13, 2023

In this Kubernetes Minikube tutorial, I will go through the required steps for a beginner to install minikube…

READ MORE

[HHOW TO GUIDES](https://devopscube.com/category/how-to/)

[How to Create AWS EKS Cluster Using eksctl](https://devopscube.com/create-aws-eks-cluster-eksctl/)

* by[**Bibin Wilson**](https://devopscube.com/author/bibinwilson/)
* December 4, 2023

In this Kubernetes tutorial, you will learn to create an AWS EKS cluster using eksctl. I will also…

READ MORE

[KKUBERNETES](https://devopscube.com/category/kubernetes/)

[Kubernetes Pod Priority, PriorityClass, and Preemption Explained](https://devopscube.com/pod-priorityclass-preemption/)

* by[**Bibin Wilson**](https://devopscube.com/author/bibinwilson/)
* October 19, 2022

In this Kubernetes tutorial, you will learn about important Kubernetes pod scheduling concepts such as Pod priority, Preemption,…

READ MORE

[AAWS](https://devopscube.com/category/aws/)

[How to Set Up Cloud Controller Manager in AWS with Kubeadm](https://devopscube.com/aws-cloud-controller-manager/)

* by[**Arun Lal**](https://devopscube.com/author/arunlal/)
* November 1, 2023

In this guide, I have added detailed steps to set up Kubernetes cloud controller manager on an AWS…

DevopsCube

©devopscube 2022. All rights reserved.

* [Privacy Policy](https://devopscube.com/privacy-policy/)
* [About](https://devopscube.com/about-devopscube/)
* [Site Map](https://devopscube.com/post-sitemap.xml)
* [Disclaimer](https://devopscube.com/disclaimer/)
* [Contribute](https://devopscube.com/contribute/)
* [Advertise](https://devopscube.com/advertise/)
* [Archives](https://devopscube.com/archives/)
* LinkedIn
* Twitter
* Facebook
* More Networks