

KUBERNETES CHEAT SHEET



WHAT IS KUBERNETES?

Kubernetes or k8s is an open-source platform designed to automate deploying, scaling, and operating application containers. With Kubernetes, you can deploy applications quickly and predictably, scale your applications on the fly, roll out new features seamlessly and efficiently utilize hardware resources.

What is Kubectl?

kubectl is the Kubernetes command line tool. It allows you to run commands against K8s clusters. For example, you can use kubectl to deploy applications, inspect and manage cluster resources, and view logs.



HOW TO INSTALL KUBECTL ON LINUX?

1. Download the latest release with the command:

```
$ curl -LO "https://dl.k8s.io/release/$(curl -L -s https://dl.k8s.io/release/stable.txt)/bin/linux/amd64/kubectl"
```

2. Validate the binary (optional)
Download the kubectl checksum file:

```
$ curl -LO "https://dl.k8s.io/$(curl -L -s https://dl.k8s.io/release/stable.txt)/bin/linux/amd64/kubectl.sha256"
```

3. Validate the kubectl binary against the checksum file:

```
$ echo "$(cat kubectl.sha256) kubectl" | sha256sum --check
```

4. Install kubectl

```
$ sudo install -o root -g root -m 0755 kubectl /usr/local/bin/kubectl
```

* If you do not have root access on the target system, you can still install kubectl to the ~/.local/bin directory:

```
$ mkdir -p ~/.local/bin/kubectl  
$ mv ./kubectl ~/.local/bin/kubectl  
$ # and then add ~/.local/bin/kubectl to $PATH
```

5. Test to ensure the version you installed is up-to-date:

```
$ kubectl version --client
```

[Additional installations >](#)

USEFUL KUBECTL COMMANDS

Get the list of current pods

```
$ kubectl get pods
```

Get the list of current nodes

```
$ kubectl get nodes
```

Describe a pod

```
$ kubectl describe pod <pod_name>
```

Describe a node

```
$ kubectl describe node <node_name>
```

Get the configuration information

```
$ kubectl config view
```

Get the list of services

```
$ kubectl get svc
```

Show the logs from a service (you can also select a container)

```
$ kubectl logs -f <service_name> or kubectl logs -c <container_name>
```

Show metrics of the nodes

```
$ kubectl top nodes
```

Show metrics for pods

```
$ kubectl top pod
```

WHAT IS HELM?

Helm is a package manager for Kubernetes, Helm Charts help you define, install and upgrade K8s applications. Helm uses a packaging format called Charts.

A **Chart** is a collection of files that describes a related set of Kubernetes resources. A single chart can be used to deploy something simple, such as a memcached pod, or something complex, such as a full web app stack with HTTP servers, databases, caches, and so on.

Explore official Helm Charts from public repositories at [Artifact Hub >](#)



HOW TO INSTALL HELM?

You can easily Install Helm with a package manager like Homebrew, Chocolatey, mreo, or by [downloading the binary](#).

To install Helm with Homebrew run the following command:

```
$ brew install helm
```

If you are downloading the binary, once it's installed, unpack the helm binary and add it to your PATH and you are good to go!

Read more about [installing Helm](#) on the official Helm documentation.

INSTALL YOUR FIRST CHART EXAMPLE

Add the [JFrog Helm repository](#) to your helm client with the following commands:

```
$ helm repo add jfrog https://charts.jfrog.io  
$ helm repo update
```

Install jfrog-container-registry with this command:

```
$ helm upgrade --install jfrog-container-registry --set artifactory.postgresql.postgresqlPassword=<postgres_password> --namespace artifactory-jcr jfrog/artifactory-jcr
```

Open your browser on port 8082 to see the JFrog Container Registry application you just installed.

HELM & JFROG

JFrog Artifactory supports Helm repositories, providing full control of your deployment and resolution of Helm charts, with [local remote and virtual repositories](#).

Helm repositories in JFrog Artifactory offers privacy, access control, high availability and massively scalable storage providing advanced, enterprise-ready repository management for Helm charts.

See [additional package type support](#)>

