

Kubernetes Cheatsheet

What is Kubernetes Kapsule and Kubernetes Kosmos?

Kubernetes is an open-source platform that enables developers to manage their containerized applications. Kapsule and Kosmo both provide a managed environment for creating, configuring, and running clusters of pre-configured machines. Kapsule clusters are composed solely of Scaleway Instances whereas Kosmos is a managed Multi-Cloud Kubernetes Engine that allows you to connect Instances and virtual or dedicated servers from any cloud provider to a single managed Control-Plane.

Creating resources

```
# Create resource(s) from file
kubectl apply -f [manifest].yaml
```

Updating resources

```
# Apply a taint that has a key-value of
taint=test with a NoSchedule effect
```

```
kubectl taint nodes [node-name]
taint=test:NoSchedule
```

```
# Mark node as unschedulable
kubectl cordon [node-name]
```

```
# Mark node as schedulable
kubectl uncordon [node-name]
```

```
# Drain node in preparation for maintenance
kubectl drain [node-name]
```

Viewing and finding resources

NODES

```
# Display all node information
kubectl get no

# Show more information about all nodes
kubectl get no -o wide

# Display node details with verbose output
kubectl describe no

# Filter the node with the specified label
kubectl get node --
selector=[label_name]

# Display node (CPU/memory) usage
kubectl top node [node_name]
```

PODS

```
# Display all container group information
kubectl get po

# Show more information about all pods
kubectl get po -o wide

# Display pod details with verbose output
kubectl describe po

# View the labels of the container group
kubectl get po --show-labels

# Display pod usage (CPU/memory)
kubectl top pod [pod_name]
```

NAMESPACE

```
# Display all namespace information
kubectl get ns

# Display namespace details
kubectl describe ns
```

DEPLOYMENTS

```
# Display all deployments information
kubectl get deploy

# Display deployments details
kubectl describe deploy

# Show more information about all
deployments
kubectl get deploy -o wide
```

SERVICES

```
# Display all services information
kubectl get svc

# Display services details
kubectl describe svc

# Show more information about all services
kubectl get svc -o wide

# Display a pod's label
kubectl get svc --show-labels
```

DAEMON SETS

```
# Display all daemon sets information
kubectl get ds

# Display the detailed state of daemonsets
within all namespace
kubectl describe ds --all-
namespaces

# Display the detailed state of daemonsets
within a namespace
kubectl describe ds
[daemonset_name] -n
[namespace_name]
```

EVENTS

```
# Display all events information
kubectl get events

# Display events information within the
namespace kube-system
kubectl get events -n kube-system

# Lists the specific resources' events or the
entire cluster
kubectl get events -w
```

LOGS

```
# Display all logs information of a specific pod
kubectl logs [pod_name]

# Display all logs information of a specific pod
for the past hour
kubectl logs --since=1h
[pod_name]

# Display all logs information of a specific pod
in a specific container
kubectl logs -f -c
[container_name] [pod_name]

# Transfer all logs information of a specific
pod in the pod.log file
kubectl logs [pod_name] > pod.log
```

SERVICE ACCOUNT

```
# Display all service account information
kubectl get sa
```

REPLICA SETS

```
# Display all replica sets information
kubectl get rs

# Display replicat sets details
kubectl describe rs

# Show more information about replica sets
kubectl get rs -o wide
```

ROLES

```
# Display all roles information within all
namespaces
kubectl get roles --all-
namespaces
```

SECRETS

```
# Display all secrets information
kubectl get secrets

# Display all secrets information within all
namespaces
kubectl get secrets --all-
namespaces
```

CONFIG MAPS

```
# Display all config maps information
kubectl get cm

# Display all config maps within all
namespaces
kubectl get cm --all-namespaces
```

INGRESS

```
# Display all ingress information
kubectl get ing

# Display all ingress information within all
namespaces
kubectl get ing --all-namespaces
```

PERSISTENT VOLUMES

```
# Display all persistent volumes information
kubectl get pv

# Display persistent volumes details
kubectl describe pv
```

PERSISTENT VOLUME CLAIM

```
# Display all persistent volume claim
information
kubectl get pvc

# Display all persistent volume claim details
kubectl describe pvc
```

STORAGE CLASS

```
# Display all storage class information
kubectl get sc
```

MULTIPLE RESOURCES

```
# Display all services and pods information
kubectl get svc, po

# Display all deploys and nodes information
kubectl get deploy, no

# Display all the pods, services, statefulsets,
etc. in a namespace. Not all the resources are
listed using this command.
kubectl get all

# Display all the pods, services, statefulsets,
etc. in all namespaces. Not all the resources
are listed using this command.
kubectl get all --all-namespaces
```