



A perfect companion to all your cluster management needs

Pragyan Tripathi @pragyanatvade

Viewing Resource Information

Nodes

kubectl get no

kubectl get no -o wide

kubectl describe no

kubectl get no -o yaml

kubectl get node --selector=[label_name]

| kubectl get nodes -o | jsonpath='{.items[*].status.addresses[? (@.type="ExternalIP")].address}'

kubectl top node [node_name]

retrieve a list of all
nodes in the current cluster

retrieve a list of all nodes
in the current cluster, with
additional details such as the
node's IP address and role

retrieve detailed information
about a specific node

retrieve a list of all nodes
in the current cluster in YAML
format

retrieve a list of all nodes
with the specified label

retrieve a list of all
node external IP addresses

display resource usage
statistics for a specific
node

Roles

kubectl get roles --all-namespaces

kubectl get roles --all-namespaces -o yaml

display a list of all roles in all namespaces

display the roles in all
namespaces in YAML format



Viewing Resource Information

Pods

kubectl get po

kubectl get po -o wide

kubectl describe po

kubectl get po --show-labels

kubectl get po -l app=[app_name]

kubectl get po −o yaml

kkubectl get pod [pod_name] -o yaml --export
namooffile.yaml

kubectl get pods --field-selector status.phase=Running

get a list of pods

get a wide view of pods, including Node name and IP

describe all the pods in the current namespace

get list of pods with labels

get list of pods with label
"app" equal to [app_name]

get yaml definition of all the pods in the current namespace

save the yaml definition of the specific pod in the file

get list of pods with the
status "Running"

Pragyan Tripathi @pragyanatvade

Viewing Resource Information

Namespaces

kubectl get ns

kubectl get ns -o yaml

kubectl describe ns

display a list of all
namespaces in the current
cluster

display the namespaces in the cluster in a yaml format

display detailed information
about a namespace in the cluster

Deployments

kubectl get deploy

kubectl describe deploy

kubectl get deploy -o wide

kubectl get deploy -o yaml

display list of all
deployments in the current
namespace

display detailed information
about deployments in the current
namespace

display list of deployments in the current namespace with additional details

display the deployments
in the current namespace
in yaml format

Pragyan Tripathi @pragyanatvade

Viewing Resource Information

Services

kubectl get svc

kubectl describe svc

kubectl get svc -o wide

kubectl get svc -o yaml

kubectl get svc --show-labels

display a list of all services
in the current namespace

display detailed information
about services in the current
namespace

display a list of services in the current namespace with additional details

display the services in the
current namespace in yaml format

display a llist of services
in the current namespace
including their labels

DaemonSets

kubectl get ds

kubectl get ds --all-namespaces
kubectl describe ds [ds_name] -n [ns_name]

display list of all daemon sets
in the current namespace

display detailed information
about a daemon set in a specific
namespace.

Pragyan Tripathi @pragyanatvade

Viewing Resource Information

kubectl get ds [ds_name] -n [ns_name] -o yaml

display a daemon set in a specific namespace in YAML format.

Events

kubectl get events

kubectl get events -n kube-system

kubectl get events -w

display list of events in the current namespace

display a list of events in
the
kube-system namespace.

watch for new events in the current namespace

Logs

kubectl logs [pod_name]

kubectl logs --since=1h [pod_name]

kubectl logs --tail=20 [pod_name]

| kubectl logs -f -c [container_name] [pod_name] # display logs of the specified
pod

display the logs of a pod for
the past 1 hour

display the last 20 lines for the logs for a pod

follow the logs for a specific
container in the pod

Pragyan Tripathi @pragyanatvade

Viewing Resource Information

kubectl logs [pod_name] > pod.log

save the logs for a pod
to a file

Service Accounts

kubectl get sa

display a list of service
accounts in the current namespace

kubectl get sa -o yaml

display a list of service
accounts in the current
namespace in yaml format.

| kubectl get serviceaccounts default -o yaml → sa.yaml # save the "default" service
account in the yaml format to a
file

kubectl replace service account default - f sa.yaml

replace the "default" service
account with the contents of a
yaml file

ReplicaSets

kubectl get rs

display a list of all replica
sets in the current namespace

Pragyan Tripathi @pragyanatvade

Viewing Resource Information

kubectl describe rs

kubectl get rs -o wide

kubectl get rs -o yaml

Multiple Resources

kubectl get svc, po

kubectl get deploy, no

kubectl get all

kubectl get all --all-namespaces

display detailed information
about replica sets in the
current namespace

display a list of replica
sets in the current namespace
with additional details

display the replica sets in the current namespace in yaml format

display list of services and pods in the current namespace

display list of deployments in the current namespace

display list of all resources
in the current namespace

display list of all resources
in all namespaces



Viewing Resource Information

Secrets

kubectl get secrets

kwbæctllggetsecrets --all-namespaces

kubectl get secrets -o yaml

display a list of secrets in the current namespace

display a list of secrets in all namespaces

display the secrets in the
current namespace in YAML format

ConfigMaps

kubectl get cm

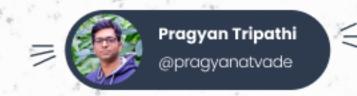
kubectl get cm --all-namespaces

kubectl get cm --all-namespaces -o yaml

display a list of config maps
in the current namespace

display a list of config maps
in all namespaces

display a list of config maps
in yaml format



Viewing Resource Information

Ingress

kubectl get ing

kubectl get ing --all-namespaces

display a list of ingresses in the current namespace

display list of all ingresses
in all namespaces

Persistent Volume

kubectl get pvc

kubectl describe pvc

display a list of persistent
volume claims in the current
namespace

display list of all persistent
volume claims in all namespaces

StorageClass

kubectl get sc

kubectl get sc -o yaml

display list of all storage
classes in the cluster

display the storage classes in
yaml format



Viewing Resource Information

API Call

kubectl get --raw /apis/metrics.k8s.io/

get raw json data for the metrics API

Cluster Info

kubectl config

kubectl cluster-info

kubectl get component statuses

view and manage Kubernetes cluster

display information about the kubernetes cluster

get the status of the various components in the kubernetes cluster

Changing Resource Attributes

Taint

kubectl taint [node_name] [taint_name]

add taint to a node

Labels

kubectl label [node_name] disktype=ssd

add label to a node

kubectl label [pod_name] env=prod

add label to a pod

Cordon / Uncordon

kubectl cordon [node_name]

mark a node as unschedulable

kubectl uncordon [node_name]

mark a node as schedulable

Drain

kubectl drain [node_name]

drain a node in preparation for maintenance

Nodes

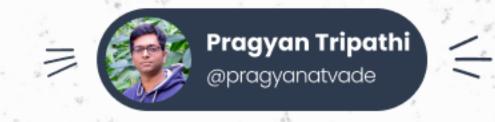
kubectl delete node [node_name]

delete a node from the cluster

kubectl edit node [node_name]

edit a node's configuration

Changing Resource Attributes



Namespaces

kubectl delete ns [namespace_name]

kubectl edit ns [namespace_name]

delete a namespace

edit a namespace's
configuration

Deployments

kubectl delete deploy [deploy_name]

kubectl edit deploy [deploy_name]

| kubectl get roles --all-namespaceskubectl expose deploy [deploy_name] --port=80 -- type=NodePort

kubectl scale deploy [deploy_name] -replicas=5

delete a deployment

edit a deployment's
configuration

expose a deployment as a NodePort service

scale a deployment to 5
replicas

Pods

kubectl delete pod [pod_name]

| kubectl edit pod [pod_name]

delete a pod

edit a pod's configuration

Adding Resources

Creating A Pod

kubectl create -f [name_of_file]

kubectl apply -f [name_of_file]

kubectl run [pod_name] --image=nginx -restart=Never

| kubectl run [pod_name] --generator=runpod/v1 --image=nginx

Creating A Service

| kubectl create svc nodeport [svc_name] --tcp=8080:80

Creating A Deployment

kubectl create -f [name_of_file]

kubectl apply -f [name_of_file]

kubectl create deploy [deploy_name] -image=nginx

create resources from a file

apply changes from a file

run a single instance of an
nginx container

run a single instance of nginx
container using "run-pod/v1"
generator

create a NodePort service that exposes TCP port 8080 on the nodes and maps it to the port 80 in the pods

create resources from a file

apply changes from a file

create a deployment with nginx
container

Adding Resources



Interactive Pod

| kubectl run [pod_name] --image=busybox -rm --it --restart=Never -- sh # run a temporary busybox
container, open an interactive
shell and delete it when it exits

Output YAML to a File

| kubectl create deploy [deploy_name] --image nginx --dry-run -o yaml > deploy.yaml

create a deployment with an nginx container and output the configuration to a file in YAML format without creating the resources.

kubectl get po [pod_name] -o yaml -export > pod.yaml

get a pod and output the configuration to a file in YAML format, including all resource details but ignoring cluster-specific information.



This Content Helpful?



If you benefit from it, Hit the like



Share your thoughts in the comment



Share with others to learn



