

Kubectl Kubernetes CheatSheet

KUBERNETES

- PDF Link: [cheatsheet-kubernetes-A4.pdf](#), Category: kubernetes
- Blog URL: <https://cheatsheet.dennyzhang.com/cheatsheet-kubernetes-A4>

1.1 Common Commands

Name	Command
List everything	<code>kubectl get all --all-namespaces</code>
Validate yaml file with dry run	<code>kubectl create --dry-run --validate -f pod-dummy.yaml</code>
Start a temporary pod for testing	<code>kubectl run --rm -i -t --image=alpine test-\$RANDOM -- sh</code>
Run wget test temporarily	<code>kubectl run --rm mytest --image=busybox -it</code>
Run curl test temporarily	<code>kubectl run --rm mytest --image=yauritux/busybox-curl -it</code>
kubectl run shell command	<code>kubectl exec -it mytest -- ls -l /etc/hosts</code>
Get system conf via configmap	<code>kubectl -n kube-system get cm kubeadm-config -o yaml</code>
kubectl run instance with replicas	<code>kubectl run my-nginx --image=nginx --replicas=2 --port=80</code>
Explain resource	<code>kubectl explain pods, kubectl explain svc</code>
Get all services	<code>kubectl get service --all-namespaces</code>
Query healthcheck endpoint	<code>curl -L http://127.0.0.1:10250/healthz</code>
Open a bash terminal in a pod	<code>kubectl exec -it storage sh</code>
Check pod environment variables	<code>kubectl exec redis-master-ft9ex env</code>
Enable kubectl shell autocompletion	<code>echo "source <(kubectl completion bash)" > ~/.bashrc, then reconnect</code>
Use minikube dockerd in your laptop	<code>eval \$(minikube docker-env)</code> , No need to docker push any more
Get services sorted by name	<code>kubectl get services --sort-by=.metadata.name</code>
Get pods sorted by restart count	<code>kubectl get pods --sort-by='.status.containerStatuses[0].restartCount'</code>
List pods, containers and images	<code>kubectl get pods -o='custom-columns=PODS:.metadata.name,CONTAINERS:.spec.containers'</code>
Links	minikube cheatsheet, docker cheatsheet

1.2 Check Performance

Name	Command
Get node resource usage	<code>kubectl top node</code>
Get pod resource usage	<code>kubectl top pod</code>
Get resource usage for a given pod	<code>kubectl top <podname> --containers</code>
List resource utilization for all containers	<code>kubectl top pod --all-namespaces --containers=true</code>

1.3 Resources Deletion

Name	Command
Delete pod	<code>kubectl delete pod <pod-name> -n <my-namespace></code>
Delete pods by labels	<code>kubectl delete pod -l env=test</code>
Delete deployments by labels	<code>kubectl delete deployment -l app=wordpress</code>
Delete persist volumes by labels	<code>kubectl delete pvc -l app=wordpress</code>
Delete statefulset only (not pods)	<code>kubectl delete sts <stateful_set_name> --cascade=false</code>

1.4 Pod

Name	Command
List all pods	<code>kubect1 get pods</code>
List pods for all namespace	<code>kubect1 get pods -all-namespaces</code>
List all critical pods	<code>kubect1 get -n kube-system pods -a</code>
List pods with more info	<code>kubect1 get pod -o wide, kubect1 get pod -o yaml</code>
Get pod info	<code>kubect1 describe pod srv-mysql-server</code>
List all pods with labels	<code>kubect1 get pods --show-labels</code>
Get Pod initContainer status	<code>kubect1 get pod --template '{{.status.initContainerStatuses}}' <pod-name></code>
kubect1 run command	<code>kubect1 exec -it -n "\$ns" "\$podname" - sh -c "echo \$msg »/dev/err.log"</code>
Get pod by selector	<code>podname=\$(kubect1 get pods -n \$namespace --selector="app=syslog" -o jsonpath='{.items[*]'</code>
List pods and containers	<code>kubect1 get pods -o='custom-columns=PODS:.metadata.name,CONTAINERS:.spec.container'</code>
List pods, containers and images	<code>kubect1 get pods -o='custom-columns=PODS:.metadata.name,CONTAINERS:.spec.container'</code>
Kubernetes Yaml Examples	link: kubernetes yaml templates

1.5 Label & Annotatation

Name	Command
Filter pods by label	<code>kubect1 get pods -l owner=denny</code>
Manually add label to a pod	<code>kubect1 label pods dummy-input owner=denny</code>
Remove label	<code>kubect1 label pods dummy-input owner-</code>
Manually add annotation to a pod	<code>kubect1 annotate pods dummy-input my-url=https://www.dennyzhang.com</code>

1.6 Deployment & Scale

Name	Command
Scale out	<code>kubect1 scale --replicas=3 deployment/nginx-app</code>
online rolling upgrade	<code>kubect1 rollout app-v1 app-v2 --image=img:v2</code>
Roll backup	<code>kubect1 rollout app-v1 app-v2 --rollback</code>
List rollout	<code>kubect1 get rs</code>
Check update status	<code>kubect1 rollout status deployment/nginx-app</code>
Check update history	<code>kubect1 rollout history deployment/nginx-app</code>
Pause/Resume	<code>kubect1 rollout pause deployment/nginx-deployment, resume</code>
Rollback to previous version	<code>kubect1 rollout undo deployment/nginx-deployment</code>
Kubernetes Yaml Examples	link: kubernetes yaml templates , link: Pausing and Resuming a Deployment

1.7 Quota & Limits

Name	Command
List Resource Quota	<code>kubect1 get resourcequota</code>
List Limit Range	<code>kubect1 get limitrange</code>
Kubernetes Yaml Examples	link: kubernetes yaml templates

1.8 Service

Name	Command
List all services	<code>kubect1 get services</code>
List service endpoints	<code>kubect1 get endpoints</code>
Get service detail	<code>kubect1 get service nginx-service -o yaml</code>
Get service cluster ip	<code>kubect1 get service nginx-service -o go-template='{{.spec.clusterIP}}'</code>
Get service cluster port	<code>kubect1 get service nginx-service -o go-template='{{(index .spec.ports 0).port}}'</code>
Kubernetes Yaml Examples	link: kubernetes yaml templates

1.9 StatefulSet

Name	Command
List statefulset	<code>kubect1 get sts</code>
Scale statefulset	<code>kubect1 scale sts <stateful_set_name> --replicas=5</code>
Delete statefulset only (not pods)	<code>kubect1 delete sts <stateful_set_name> --cascade=false</code>
Kubernetes Yaml Examples	link: kubernetes yaml templates

1.10 Volumes & Volume Claims

Name	Command
Check the mounted volumes	<code>kubectl exec storage ls /data</code>
Check persist volume	<code>kubectl describe pv pv0001</code>
List storage class	<code>kubectl get storageclass</code>
Copy files	<code>kubectl cp /tmp/foo <namespace1>/<pod1>:/tmp/bar</code>
Kubernetes Yaml Examples	link: kubernetes yaml templates

1.11 Security

Name	Command
List certificates	<code>kubectl get csr</code>
Kubernetes Yaml Examples	link: kubernetes yaml templates

1.12 Resources

Name	Command
Customize resource definition	<code>kubectl set resources deployment nginx -c=nginx --limits=cpu=200m,memory=512Mi</code>

1.13 Components & Services

- Services on Master Nodes

Name	Summary
kube-apiserver	exposes the Kubernetes API from master nodes
etcd	reliable data store for all k8s cluster data
kube-scheduler	schedule pods to run on selected nodes
kube-controller-manager	node controller, replication controller, endpoints controller, and service account & token controllers

- Services on Worker Nodes

Name	Summary
kubelet	makes sure that containers are running in a pod
kube-proxy	perform connection forwarding
Container Runtime	Kubernetes supported runtimes: Docker, rkt, runc and any OCI runtime-spec implementation.

- Addons: pods and services that implement cluster features

Name	Summary
DNS	serves DNS records for Kubernetes services
Web UI	a general purpose, web-based UI for Kubernetes clusters
Container Resource Monitoring	collect, store and serve container metrics
Cluster-level Logging	save container logs to a central log store with search/browsing interface

- Tools

Name	Summary
kubectl	the command line util to talk to k8s cluster
kubeadm	the command to bootstrap the cluster
kubefed	the command line to control a Kubernetes Cluster Federation
Kubernetes Components	link: Kubernetes Components

1.14 Other Components

1.14.1 Log files

Name	Command
API Server.log= in master node	<code>/var.log=/kube-apiserver.log</code>
Scheduler.log= in master node	<code>/var.log=/kube-scheduler.log</code>
Controller.log= in master node	<code>/var.log=/kube-controller-manager.log</code>
Kubelet.log= in worker node	<code>/var.log=/kubelet.log</code>
Kube Proxy.log= in worker node	<code>/var.log=/kubelet-proxy.log</code>

1.14.2 Events & Metrics

Name	Command
View all events	<code>kubectl get events --all-namespaces</code>

1.14.3 Namespace & Security

Name	Command
List authenticated contexts	<code>kubectl config get-contexts</code>
Switch context	<code>kubectl config use-context <cluster-name></code>
List all namespaces defined	<code>kubectl get namespaces</code>
kubectl config file	<code>~/.kube/config</code>
Kubernetes Yaml Examples	link: kubernetes yaml templates

1.14.4 Network

Name	Command
Temporarily add a port-forwarding	<code>kubectl port-forward redis-izl09 6379</code>
Add port-forwarding for deployment	<code>kubectl port-forward deployment/redis-master 6379:6379</code>
Add port-forwarding for replicaset	<code>kubectl port-forward rs/redis-master 6379:6379</code>
Add port-forwarding for service	<code>kubectl port-forward svc/redis-master 6379:6379</code>
Get network policy	<code>kubectl get NetworkPolicy</code>

1.15 Basic

1.15.1 Key Concepts

Name	Summary
CNCF	Cloud Native Computing Foundation
CRI	Container Runtime Interface
CNI	Container Network Interface
CSI	Container Storage Interface

1.15.2 Kubernetes Critical Files

Name	Comment
Config folder	<code>/etc/kubernetes/</code>
Certificate files	<code>/etc/kubernetes/pki/</code>
Credentials to API server	<code>/etc/kubernetes/kubelet.conf</code>
Superuser credentials	<code>/etc/kubernetes/admin.conf</code>
Kubernetes working dir	<code>/var/lib/kubelet/</code>
Docker working dir	<code>/var/lib/docker/</code>
Etd working dir	<code>/var/lib/etcd/</code>
Network cni	<code>/etc/cni/net.d/</code>
Docker container log	<code>/var/log/containers/</code>
Log files	<code>/var/log/pods/</code>
Env	<code>export KUBECONFIG=/etc/kubernetes/admin.conf</code>
Env	<code>/etc/systemd/system/kubelet.service.d/10-kubeadm.conf</code>

1.15.3 Check status

Name	Summary
List everything	<code>kubectl get all --all-namespaces</code>
Get cluster info	<code>kubectl cluster-info</code>
Get configuration	<code>kubectl config view</code>
Get kubectl version	<code>kubectl version</code>
Get component status	<code>kubectl get componentstatus</code>
Similar to <code>docker ps</code>	<code>kubectl get nodes</code>
Similar to <code>docker inspect</code>	<code>kubectl describe pod nginx-app-413181-cn</code>
Similar to <code>docker logs</code>	<code>kubectl logs</code>
Similar to <code>docker exec</code>	<code>kubectl exec</code>
Get services for current namespace	<code>kubectl get svc</code>
Get node status	<code>kubectl describe node <node_name></code>

1.15.4 Kubernetes Developer Resources

Name	Summary
API Conventions	link: API Conventions

1.16 Minikube

Name	Command
Get minikube version	<code>minikube version</code> , link: all minikube releases
Start minikube with a specific k8s version	<code>minikube start --kubernetes-version v1.10.0</code>
Start minikube env with a bigger machine flavor	<code>minikube start --memory 5120 --cpus=4</code>
Gets all available Kubernetes versions for minikube	<code>minikube get-k8s-versions</code>
Mount host OS's folder to minikube VM	<code>minikube mount /host-mount-path:/vm-mount-path</code>
Check minikube config in your host OS desktop folder of <code>k8s.io/minikube-hostpath</code> provisioner	<code>~/minikube/machines/minikube/config.json</code> <code>/tmp/hostpath-provisioner, /tmp/hostpath_pv</code>
Critical minikube folder	<code>/var/lib/localkube, /var/lib/docker, /data</code>
minikube <code>docker-env</code>	<code>eval \$(minikube docker-env)</code>
Get minikube log	<code>minikube logs</code>
Get dashboard	<code>minikube dashboard</code>
SSH to minikube vm	<code>minikube ssh</code>
Get ip	<code>minikube ip</code>
Get cluster info	<code>kubectl cluster-info</code>
List addons	<code>minikube addons list</code>
Get service info	<code>minikube service <srv_name></code>

1.17 Misc scripts

- Tail pod log by label

```
namespace="mynamespace"
mylabel="app=mylabel"
kubectl get pod -l "$mylabel" -n "$namespace" | tail -n1 \
  | awk -F' ' '{print $1}' | xargs -I{} \
  kubectl logs -n "$namespace" -f {}
```

- Get node hardware resource utilization

```
kubectl get nodes --no-headers \
  | awk '{print $1}' | xargs -I {} \
  sh -c 'echo {}; kubectl describe node {} | grep Allocated -A 5'
```

```
kubectl get nodes --no-headers | awk '{print $1}' | xargs -I {} \
  sh -c 'echo {}; kubectl describe node {} | grep Allocated -A 5 \
  | grep -ve Event -ve Allocated -ve percent -ve -- ; echo'
```

- Apply the configuration in `manifest.yaml` and delete all the other configmaps that are not in the file.

```
kaubectl apply --prune -f manifest.yaml --all --prune-whitelist=core/v1/ConfigMap
```

1.18 More Resources

License: Code is licensed under MIT License.

<https://kubernetes.io/docs/reference/kubectl/cheatsheet/>

https://github.com/kubecamp/kubernetes_in_2_days

<https://marc.xn--wckerlin-0za.ch/computer/kubernetes-on-ubuntu-16-04>

<https://codefresh.io/kubernetes-guides/kubernetes-cheat-sheet/>

File me Issues or star this repo.

See more CheatSheets from Denny: [#denny-cheatsheets](#)

- Check kubernetes cheatsheet: <https://cheatsheet.dennyzhang.com/cheatsheet-minikube-A4>
- Yaml Templates: <https://cheatsheet.dennyzhang.com/cheatsheet-kubernetes-yaml>