1 Kubectl Kubernetes CheatSheet KUBERNETES

- PDF Link: cheatsheet-kubernetes-A4.pdf, Category: kubernetes
- Blog URL: `https://cheatsheet.dennyzhang.com/cheatsheet-kubernetes-A4`
- Related posts: Kubernetes Yaml, #denny-cheatsheets

File me Issues or star this repo.

1.1 Common Commands

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

1.2 Check Performance

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

1.3 Resources Deletion

Name	Command
Delete pod	`kubectl delete pod/<pod-name> -n <my-namespace>`
Delete pods by labels	`kubectl delete pod -l env=test`
Delete deployments by labels	`kubectl delete deployment -l app=wordpress`
Delete all resources filtered by labels	`kubectl delete pods,services -l name=myLabel`
Delete resources under a namespace	`kubectl -n my-ns delete po,svc --all`
Delete persist volumes by labels	`kubectl delete pvc -l app=wordpress`
Delete statefulset only (not pods)	`kubectl delete sts/<stateful_set_name> --cascade=false`

1.4 Pod

Name	Command
List all pods	`kubectl get pods`
List pods for all namespace	`kubectl get pods -all-namespaces`
List all critical pods	`kubectl get -n kube-system pods -a`
List pods with more info	`kubectl get pod -o wide`, `kubectl get pod/<pod-name> -o yaml`
Get pod info	`kubectl describe pod/srv-mysql-server`
List all pods with labels	`kubectl get pods --show-labels`
Get Pod initContainer status	`kubectl get pod --template '{{.status.initContainerStatuses}}' <pod-name>`
kubectl run command	kubectl exec -it -n "$ns" "$podname" – sh -c "echo $msg »/dev/err.log"
Kubernetes Yaml Examples	Link: kubernetes yaml templates

1.5 Pod Advanced

Name	Command
Watch pods	`kubectl get pods -n wordpress --watch`
Get pod by selector	podname=$(kubectl get pods -n $namespace –selector="app=syslog" -o jsonpath='{.items[*]
List pods and containers	kubectl get pods -o='custom-columns=PODS:.metadata.name,CONTAINERS:.spec.container
List pods, containers and images	kubectl get pods -o='custom-columns=PODS:.metadata.name,CONTAINERS:.spec.container

1.6 Label & Annontation

Name	Command
Filter pods by label	`kubectl get pods -l owner=denny`
Manually add label to a pod	`kubectl label pods dummy-input owner=denny`
Remove label	`kubectl label pods dummy-input owner-`
Manually add annonation to a pod	`kubectl annotate pods dummy-input my-url=https://www.dennyzhang.com`

1.7 Deployment & Scale

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

1.8 Quota & Limits & Resource

Name	Command
List Resource Quota	`kubectl get resourcequota`
List Limit Range	`kubectl get limitrange`
Customize resource definition	`kubectl set resources deployment nginx -c=nginx --limits=cpu=200m,memory=512Mi`
Kubernetes Yaml Examples	Link: kubernetes yaml templates

1.9 Service

Name	Command
List all services	`kubectl get services`
List service endpoints	`kubectl get endpoints`
Get service detail	`kubectl get service nginx-service -o yaml`
Get service cluster ip	kubectl get service nginx-service -o go-template='{{.spec.clusterIP}}'
Get service cluster port	kubectl get service nginx-service -o go-template='{{(index .spec.ports 0).port}}'
Expose deployment as lb service	`kubectl expose deployment/my-app --type=LoadBalancer --name=my-service`
Expose service as lb service	`kubectl expose service/wordpress-1-svc --type=LoadBalancer --name=wordpress-lb`
Kubernetes Yaml Examples	Link: kubernetes yaml templates

1.10 StatefulSet

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

1.11 Patch

Name	Command
Patch service to loadbalancer	`kubectl patch svc "$APP_INSTANCE_NAME-grafana" -p '{"spec": {"type": "LoadBalan`

1.12 Volumes & Volume Claims

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

1.13 Security

Name	Command
List certificates	`kubectl get csr`
Kubernetes Yaml Examples	Link: kubernetes yaml templates

1.14 Extenstions

Name	Summary
List api group	`kubectl api-versions`
List all CRD	`kubectl get crd`

1.14.1 Common Commands

Name	Command
List everything	`kubectl get all --all-namespaces`
Validate yaml file with dry run	`kubectl create --dry-run --validate -f pod-dummy.yaml`
Start a temporary pod for testing	`kubectl run --rm -i -t --image=alpine test-$RANDOM -- sh`
Run wget test temporarily	`kubectl run --rm mytest --image=busybox -it`
Run curl test temporarily	`kubectl run --rm mytest --image=yauritux/busybox-curl -it`
Get system conf via configmap	`kubectl -n kube-system get cm kubeadm-config -o yaml`
Explain resource	`kubectl explain pods`, `kubectl explain svc`
Get all services	`kubectl get service --all-namespaces`
Get services sorted by name	kubectl get services –sort-by=.metadata.name
Get pods sorted by restart count	kubectl get pods –sort-by='.status.containerStatuses[0].restartCount'
Query healthcheck endpoint	`curl -L http://127.0.0.1:10250/healthz`
Open a bash terminal in a pod	`kubectl exec -it storage sh`
Check pod environment variables	`kubectl exec redis-master-ft9ex env`
Enabling shell autocompletion for kubectl	`echo "source <(kubectl completion bash)" » ~/.bashrc`, then reconnect
In mac desktop, use minikube dockerd	`eval $(minikube docker-env)`, No need to docker push any more

1.15 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

1.16 Check Performance

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

1.17 Resources Deletion

Name	Command
Delete pod	`kubectl delete pod hello-node-95913-n63qs -n $my-namespace`
Delete pods by labels	`kubectl delete pod -l env=test`
Delete deployments by labels	`kubectl delete deployment -l app=wordpress`
Delete persist volumes by labels	`kubectl delete pvc -l app=wordpress`
Delete statefulset only (not pods)	`kubectl delete sts <stateful_set_name> --cascade=false`

1.18 Pod

Name	Command
List all pods	`kubectl get pods`
List pods for all namespace	`kubectl get pods -all-namespaces`
List all critical pods	`kubectl get -n kube-system pods -a`
List pods with more info	`kubectl get pod -o wide`, `kubectl get pod -o yaml`
Get pod info	`kubectl describe pod srv-mysql-server`
List all pods with labels	`kubectl get pods --show-labels`
Get Pod initContainer status	`kubectl get pod --template '{{.status.initContainerStatuses}}' <pod-name>`
kubectl run command	kubectl exec -it -n "$ns" "$podname" – sh -c "echo $msg »/dev/err.log"
Get pod by selector	podname=$(kubectl get pods -n $namespace –selector="app=syslog" -o jsonpath='{.items[*].me
List pods with docker images	kubectl get pods -o=jsonpath='{range .items[*]}{.metadata.name}:{.spec.containers[0].name}{"'
Kubernetes Yaml Examples	link: kubernetes yaml templates

1.19 Label & Annotation

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

1.20 Deployment & Scale

link: Pausing and Resuming a Deployment

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

1.21 Quota & Limits

Name	Command
List Resource Quota	`kubectl get resourcequota`
List Limit Range	`kubectl get limitrange`
Kubernetes Yaml Examples	link: kubernetes yaml templates

1.22 Service

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

1.23 StatefulSet

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

1.24 Volumes & Volume Claims

Name	Command
Check the mounted volumes	`kubectl exec storage ls /data`
Check persist volume	`kubectl describe pv pv0001`
List storage class	`kubectl get storageclass`
Kubernetes Yaml Examples	link: kubernetes yaml templates

1.25 Security

Name	Command
List certificates	`kubectl get csr`
Kubernetes Yaml Examples	link: kubernetes yaml templates

1.26 Resources

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

1.27 Other Components

1.27.1 Log files

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

1.27.2 Events & Metrics

Name	Command
View all events	`kubectl get events --all-namespaces`

1.27.3 Namespace & Security

Name	Command
List authenticated contexts	`kubectl config get-contexts`
Load context from config file	`kubectl get cs --kubeconfig kube_config.yml`
List contexts	`kubectl config get-contexts`
Switch context	`kubectl config use-context <cluster-name>`
Delete the specified context	`kubectl config delete-context <cluster-name>`
List all namespaces defined	`kubectl get namespaces`
kubectl config file	`~/.kube/config`
Kubernetes Yaml Examples	link: kubernetes yaml templates

1.27.4 Network

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

1.28 Basic

1.28.1 Key Concepts

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

1.28.2 Kubernets Critical Files

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

1.28.3 Check status

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

1.28.4 Kubernetes Developer Resources

Name	Summary
API Conventions	Link: API Conventions

1.29 Minikube

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

1.30 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
```

1.31 More Resources

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/