1 Kubectl Kubernetes CheatSheet CLOUD

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

File me Issues or star this repo.

1.1 Common Commands

Name	Command
Run curl test temporarily	`kubectl run --generator=run-pod/v1 --rm mytest --image=yauritux/busybox-cu`
Run wget test temporarily	`kubectl run --generator=run-pod/v1 --rm mytest --image=busybox -it wget`
Run nginx deployment with 2 replicas	`kubectl run my-nginx --image=nginx --replicas=2 --port=80`
Run nginx pod and expose it	`kubectl run my-nginx --restart=Never --image=nginx --port=80 --expose`
Run nginx deployment and expose it	`kubectl run my-nginx --image=nginx --port=80 --expose`
List authenticated contexts	`kubectl config get-contexts`, `~/.kube/config`
Set namespace preference	`kubectl config set-context <context_name> --namespace=<ns_name>`
List pods with nodes info	`kubectl get pod -o wide`
List everything	`kubectl get all --all-namespaces`
Get all services	`kubectl get service --all-namespaces`
Get all deployments	`kubectl get deployments --all-namespaces`
Show nodes with labels	`kubectl get nodes --show-labels`
Get resources with json output	`kubectl get pods --all-namespaces -o json`
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`
Get deployment yaml	`kubectl -n denny-websites get deployment mysql -o yaml`
Explain resource	`kubectl explain pods`, `kubectl explain svc`
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
Kubectl apply a folder of yaml files	`kubectl apply -R -f .`
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'
List pods and images	kubectl get pods -o='custom-columns=PODS:.metadata.name,Images:.spec.containers[*
List all container images	list-all-images.sh
kubeconfig skip tls verification	skip-tls-verify.md
Ubuntu install kubectl	`"deb https://apt.kubernetes.io/ kubernetes-xenial main"`
Reference	GitHub: kubernetes releases
Reference	minikube cheatsheet, docker cheatsheet, OpenShift 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 pod by force	`kubectl delete pod/<pod-name> --grace-period=0 --force`
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 state fulset only (not pods)	`kubectl delete sts/<stateful_set_name> --cascade=false`

1.4 Log & Conf 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`
kubectl config file	`~/.kube/config`
Kubernets working dir	`/var/lib/kubelet/`
Docker working dir	`/var/lib/docker/`, `/var/log/containers/`
Etcd working dir	`/var/lib/etcd/`
Network cni	`/etc/cni/net.d/`
Log files	`/var/log/pods/`
log in worker node	`/var/log/kubelet.log`, `/var/log/kube-proxy.log`
log in master node	`kube-apiserver.log`, `kube-scheduler.log`, `kube-controller-manager.log`
Env	`/etc/systemd/system/kubelet.service.d/10-kubeadm.conf`
Env	export KUBECONFIG=/etc/kubernetes/admin.conf

1.5 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`
List all unhealthy pods	kubectl get pods –field-selector=status.phase!=Running –all-namespaces
List running pods	kubectl get pods –field-selector=status.phase=Running
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"
Watch pods	`kubectl get pods -n wordpress --watch`
Get pod by selector	kubectl get pods –selector="app=syslog" -o jsonpath='{.items[*].metadata.name}'
List pods and images	kubectl get pods -o='custom-columns=PODS:.metadata.name,Images:.spec.containers[*].image'
List pods and containers	-o='custom-columns=PODS:.metadata.name,CONTAINERS:.spec.containers[*].name'
Reference	Link: kubernetes yaml templates

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://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`
Reference	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`
Customize resource definition	`kubectl set resources deployment nginx -c=nginx --limits=memory=512Mi`
Reference	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=ns1`
Reference	Link: kubernetes yaml templates

1.10 Secrets

Name	Command
List secrets	`kubectl get secrets --all-namespaces`
Generate secret	`echo -n 'mypasswd', then redirect to base64 --decode`
Get secret	`kubectl get secret denny-cluster-kubeconfig`
Get a specific field of a secret	kubectl get secret denny-cluster-kubeconfig -o jsonpath="{.data.value}"
Create secret from cfg file	kubectl create secret generic db-user-pass –from-file=./username.txt
Reference	Link: kubernetes yaml templates, Link: Secrets

1.11 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`
Reference	Link: kubernetes yaml templates

1.12 Volumes & Volume Claims

Name	Command
List storage class	`kubectl get storageclass`
Check the mounted volumes	`kubectl exec storage ls /data`
Check persist volume	`kubectl describe pv/pv0001`
Copy local file to pod	`kubectl cp /tmp/my <some-namespace>/<some-pod>:/tmp/server`
Copy pod file to local	`kubectl cp <some-namespace>/<some-pod>:/tmp/server /tmp/my`
Reference	Link: kubernetes yaml templates

1.13 Events & Metrics

Name	Command
View all events	`kubectl get events --all-namespaces`
List Events sorted by timestamp	kubectl get events –sort-by=.metadata.creationTimestamp

1.14 Node Maintenance

Name	Command
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`

1.15 Namespace & Security

Name	Command
List authenticated contexts	`kubectl config get-contexts`, `~/.kube/config`
Set namespace preference	`kubectl config set-context <context_name> --namespace=<ns_name>`
Switch context	`kubectl config use-context <cluster-name>`
Load context from config file	`kubectl get cs --kubeconfig kube_config.yml`
Delete the specified context	`kubectl config delete-context <cluster-name>`
List all namespaces defined	`kubectl get namespaces`
List certificates	`kubectl get csr`
Check user privilege	kubectl –as=system:serviceaccount:ns-denny:test-privileged-sa -n ns-denny auth can-i use pods/lis
Check user privilege	`kubectl auth can-i use pods/list`
Reference	Link: kubernetes yaml templates

1.16 Network

Name	Command
Temporarily add a port-forwarding	`kubectl port-forward redis-134 6379:6379`
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`

1.17 Patch

Name	Summary
Patch service to loadbalancer	kubectl patch svc svc_{name}$ -p '{"spec": {"type": "LoadBalancer"}}'

1.18 Extenstions

Name	Summary
Enumerates the resource types available	`kubectl api-resources`
List api group	`kubectl api-versions`
List all CRD	`kubectl get crd`
List storageclass	`kubectl get storageclass`

1.19 Components & Services

1.19.1 Services on Master Nodes

Name	Summary
kube-apiserver	API gateway. 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	Reconcile the states. node/replication/endpoints/token controller and service account, etc
cloud-controller-manager	

1.19.2 Services on Worker Nodes

Name	Summary
kubelet	A node agent makes sure that containers are running in a pod
kube-proxy	Manage network connectivity to the containers. e.g, iptable, ipvs
Container Runtime	Kubernetes supported runtimes: dockerd, cri-o, runc and any OCI runtime-spec implementation.

1.19.3 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

1.19.4 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

1.20 More Resources

`https://kubernetes.io/docs/reference/kubectl/cheatsheet/`
`https://codefresh.io/kubernetes-guides/kubernetes-cheat-sheet/`