

```
2001 clear
2002 kubectl get all
2003 kubectl delete deployments nginx-app redis-app
2004 clear
2005 kubectl get all
2006 cd yaml_files/
2007 ls
2008 cd replicaset/
2009 ls
2010 vi replicaset.yaml
2011 kubectl create -f replicaset.yaml
2012 kubectl delete -f replicaset.yaml
2013 kubectl create -f replicaset.yaml
2014 kubectl delete pod pod/web-app-rs-7sthj
2015* kubectl delete po web-app-rs-7sthj
2016 kubectl get replicaset web-app-rs
2017 kubectl edit replicaset.apps web-app-rs
2018 vi replicaset.yaml
2019 kubectl apply -f replicaset.yaml
2020 vi replicaset.yaml
2021 kubectl replace -f replicaset.yaml
2022 cat replicaset.yaml
2023 clear
2024 kubectl scale --replicas=3 rs web-app-rs
2025 kubectl --help
2026 clear
2027 kubectl run nginx1 --image nginx
2028 kubectl delete -f replicaset.yaml
2029 clear
2030 kubectl delete pod nginx1
2031 clear
2032 kubectl create deployment redis-app --image redis -o yaml --dry-run=client >
redis-dep1.yaml
2033 ls
2034 clear
2035 vi redis-dep1.yaml
2036 kubectl api-resources --api-group=
2037 kubectl api-resources --api-group=apps
2038 clear
2039 kubectl explain pods
2040 kubectl explain --recursive pod.spec.containers.image
2041 clear
2042 vi redis-dep1.yaml
2043 kubectl apply -f redis-dep1.yaml
2044 kubectl delete pod redis-app-ffcff4cbb-rkl9k
2045 kubectl delete deploy redis-app
2046 clear
2047 vi nginx-app-deployment.yaml
2048 cd ../deployment/
2049 clear
```

```
2050 vi nginx-app-deployment.yaml
2051 kubectl apply -f nginx-app-deployment.yaml
2052 vi nginx-app-deployment.yaml
2053 kubectl apply -f nginx-app-deployment.yaml
2054 kubectl edit deployments.apps nginx-app
2055 kubectl describe deployment nginx-app
2056 vi nginx-app-deployment.yaml
2057 kubectl rollout status deployment nginx-app
2058 clear
2059 kubectl rollout history deployment nginx-app
2060 kubectl rollout status deployment nginx-app
2061 kubectl set image deployment nginx-app nginx=nginx:1.18 --record
2062 kubectl rollout status deployment nginx-app
2063 kubectl rollout history deployment nginx-app
2064 kubectl describe deployments.apps nginx-app | grep Annotations -A 2
2065 kubectl rollout history deployment nginx-app --revision=2
2066 kubectl rollout history deployment nginx-app --revision=1
2067 kubectl rollout undo deployment nginx-app --to-revision=1
2068 kubectl describe deployments.apps nginx-app | grep Annotations -A 2
2069 kubectl rollout status deployment nginx-app
2070 kubectl rollout history deployment nginx-app --revision=1
2071 kubectl rollout history deployment nginx-app
2072 kubectl set image deployment nginx-app nginx=nginx:1.19 --record
2073 kubectl rollout status deployment nginx-app
2074 kubectl rollout history deployment nginx-app --revision=2
2075 kubectl rollout history deployment nginx-app --revision=3
2076 kubectl rollout restart deployment nginx-app
2077 kubectl rollout status deployment nginx-app
2078 kubectl rollout history deployment nginx-app
2079 kubectl describe deployments.apps nginx-app | grep Annotations -A 2
2080 kubectl rollout restart deployment nginx-app
2081 kubectl delete deployment nginx-app
2082 clear
2083 cd ../hpa/
2084 pwd
2085 ls
2086 top
2087 lscpu
2088 clear
2089 ls
2090 mv php-apache.yaml bkp_php-apache.yaml
2091 ls
2092 cd metrics-server/
2093 ls
2094 clear
2095 kubectl get all -A
2096 kubectl get all -A -o wide
2097 kubectl delete -f metrics.yaml
2098 clear
2099 kubectl get all -A
```

```
2100 ls
2101 kubectl apply -f metrics.yaml
2102 kubectl get all -A
2103 kubectl top nodes
2104 free
2105 free -g
2106 clear
2107 kubectl top nodes
2108 kubectl top pods
2109 kubectl top nodes --sort-by=memory
2110 vi php-apache.yaml
2111 kubectl apply -f https://k8s.io/examples/application/php-apache.yaml
2112 vi php-apache.yaml
2113 kubectl autoscale deployment php-apache --cpu-percent=50 --min=1 --max=10
2114 kubectl edit hpa php-apache
```

-----  
-----  
\*\*YAML Files:

#### 1. Replicaset:

```
apiVersion: apps/v1
kind: ReplicaSet
metadata:
  name: web-app-rs
spec:
  replicas: 5
  selector:
    matchLabels:
      tier: web-app
  template:
    metadata:
      labels:
        tier: web-app
    name: web-app-pod
    spec:
      containers:
        - image: nginx
          name: web-server
```

#### 2. Deployment:

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx-app
  labels:
    app: nginx-dep
spec:
```

```
replicas: 3
selector:
  matchLabels:
    app: nginx
template:
  metadata:
    labels:
      app: nginx
  spec:
    containers:
      - name: nginx
        image: nginx:1.17
```

### 3. Metric Server:

```
apiVersion: v1
kind: ServiceAccount
metadata:
  labels:
    k8s-app: metrics-server
  name: metrics-server
  namespace: kube-system
---
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRole
metadata:
  labels:
    k8s-app: metrics-server
  rbac.authorization.k8s.io/aggregate-to-admin: "true"
  rbac.authorization.k8s.io/aggregate-to-edit: "true"
  rbac.authorization.k8s.io/aggregate-to-view: "true"
  name: system:aggregated-metrics-reader
rules:
- apiGroups:
  - metrics.k8s.io
  resources:
  - pods
  - nodes
  verbs:
  - get
  - list
  - watch
---
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRole
metadata:
  labels:
    k8s-app: metrics-server
  name: system:metrics-server
rules:
```

```

- apiGroups:
  - ""
  resources:
  - pods
  - nodes
  - nodes/stats
  - namespaces
  - configmaps
  verbs:
  - get
  - list
  - watch
---
apiVersion: rbac.authorization.k8s.io/v1
kind: RoleBinding
metadata:
  labels:
    k8s-app: metrics-server
  name: metrics-server-auth-reader
  namespace: kube-system
roleRef:
  apiGroup: rbac.authorization.k8s.io
  kind: Role
  name: extension-apiserver-authentication-reader
subjects:
- kind: ServiceAccount
  name: metrics-server
  namespace: kube-system
---
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRoleBinding
metadata:
  labels:
    k8s-app: metrics-server
  name: metrics-server:system:auth-delegator
roleRef:
  apiGroup: rbac.authorization.k8s.io
  kind: ClusterRole
  name: system:auth-delegator
subjects:
- kind: ServiceAccount
  name: metrics-server
  namespace: kube-system
---
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRoleBinding
metadata:
  labels:
    k8s-app: metrics-server
  name: system:metrics-server

```

```

roleRef:
  apiGroup: rbac.authorization.k8s.io
  kind: ClusterRole
  name: system:metrics-server
subjects:
- kind: ServiceAccount
  name: metrics-server
  namespace: kube-system
---
apiVersion: v1
kind: Service
metadata:
  labels:
    k8s-app: metrics-server
  name: metrics-server
  namespace: kube-system
spec:
  ports:
  - name: https
    port: 443
    protocol: TCP
    targetPort: https
  selector:
    k8s-app: metrics-server
---
apiVersion: apps/v1
kind: Deployment
metadata:
  labels:
    k8s-app: metrics-server
  name: metrics-server
  namespace: kube-system
spec:
  selector:
    matchLabels:
      k8s-app: metrics-server
  strategy:
    rollingUpdate:
      maxUnavailable: 0
  template:
    metadata:
      labels:
        k8s-app: metrics-server
    spec:
      containers:
      - args:
        - --cert-dir=/tmp
        - --secure-port=443
        - --kubelet-preferred-address-types=InternalIP,ExternalIP,Hostname
        - --kubelet-use-node-status-port

```

```
- --metric-resolution=15s
- --kubelet-insecure-tls
image: k8s.gcr.io/metrics-server/metrics-server:v0.5.0
imagePullPolicy: IfNotPresent
livenessProbe:
  failureThreshold: 3
  httpGet:
    path: /livez
    port: https
    scheme: HTTPS
    periodSeconds: 10
name: metrics-server
ports:
- containerPort: 443
  name: https
  protocol: TCP
readinessProbe:
  failureThreshold: 3
  httpGet:
    path: /readyz
    port: https
    scheme: HTTPS
    initialDelaySeconds: 20
    periodSeconds: 10
resources:
  requests:
    cpu: 100m
    memory: 200Mi
securityContext:
  readOnlyRootFilesystem: true
  runAsNonRoot: true
  runAsUser: 1000
volumeMounts:
- mountPath: /tmp
  name: tmp-dir
nodeSelector:
  kubernetes.io/os: linux
priorityClassName: system-cluster-critical
serviceAccountName: metrics-server
volumes:
- emptyDir: {}
  name: tmp-dir
```

---

```
apiVersion: apiregistration.k8s.io/v1
kind: APIService
metadata:
  labels:
    k8s-app: metrics-server
  name: v1beta1.metrics.k8s.io
spec:
```

```
group: metrics.k8s.io
groupPriorityMinimum: 100
insecureSkipTLSVerify: true
service:
  name: metrics-server
  namespace: kube-system
version: v1beta1
versionPriority: 100
```