

```
kubectl create clusterrolebinding sa-ns1-view --clusterrole=view --serviceaccount=ns1:sa-ns1
kubectl create clusterrolebinding sa-ns2-view --clusterrole=view --serviceaccount=ns2:sa-ns2
```

```
kubectl auth can-i get pods --as=system:serviceaccount:ns1:sa-ns1 -n ns1
```

```
kubectl auth can-i get pods --as=system:serviceaccount:ns1:sa-ns1 -n ns1
kubectl auth can-i create deployments --as=system:serviceaccount:ns1:sa-ns1 -n ns1
kubectl auth can-i delete deployments --as=system:serviceaccount:ns1:sa-ns1 -n ns1
kubectl auth can-i get pods --as=system:serviceaccount:ns2:sa-ns2 -n ns2
kubectl auth can-i create deployments --as=system:serviceaccount:ns2:sa-ns2 -n ns2
kubectl auth can-i delete deployments --as=system:serviceaccount:ns2:sa-ns2 -n ns2
```

```
k create clusterrole view --verb=create,delete --resource=deployments -n ns1
```

```
apiVersion: rbac.authorization.k8s.io/v1
```

```
kind: clusterrole
```

```
metadata:
```

```
name: deployment-manager
```

```
rules:
```

```
- apiGroups: ["apps"]
```

```
resources: ["deployments"]
```

```
verbs: ["create", "delete"]
```

```
kubectl apply -f deployment-role.yaml -n ns1
```

```
kubectl apply -f deployment-role.yaml -n ns2
```

## 1st question:

```
controlplane $ kubectl create clusterrole deployment-clusterrole --verb=create --resource=deployment,statefulSet,daemonSet
clusterrole.rbac.authorization.k8s.io/deployment-clusterrole created
controlplane $
controlplane $ k create ns app-team1
namespace/app-team1 created
controlplane $
controlplane $
controlplane $ k create sa ckd-token -n app-team1
serviceaccount/ckd-token created
controlplane $
controlplane $ kubectl create clusterrolebinding ckd-token-clusterrole --clusterrole= --serviceaccount=app-team1:ckd-token
error: failed to create clusterrolebinding: ClusterRoleBinding.rbac.authorization.k8s.io "ckd-token-clusterrole" is invalid: roleRef.name: Required value
controlplane $ kubectl create clusterrolebinding ckd-clusterrole --clusterrole= --serviceaccount=app-team1:ckd-token
error: failed to create clusterrolebinding: ClusterRoleBinding.rbac.authorization.k8s.io "ckd-clusterrole" is invalid: roleRef.name: Required value
controlplane $
controlplane $
controlplane $ kubectl create clusterrolebinding ckd-clusterrole --clusterrole=deployment-clusterrole --serviceaccount=app-team1:ckd-token
clusterrolebinding.rbac.authorization.k8s.io/ckd-clusterrole created
controlplane $
controlplane $
controlplane $
```

```
controlplane $
controlplane $
controlplane $
controlplane $ k get sa -n app-team1
NAME          SECRETS  AGE
cicd-token    0        7m57s
default       0        8m20s
controlplane $
controlplane $
controlplane $
controlplane $ k get clusterrole | grep deployment
deployment-clusterrole                2024-09-14T04:11:04Z
system:controller:deployment-controller 2024-09-09T12:39:33Z
controlplane $
controlplane $
controlplane $ k get clusterrolebindings.rbac.authorization.k8s.io -n app-team1 | grep deployment
cicd-clusterrole                      ClusterRole/deployment-clusterrole
system:controller:deployment-controller ClusterRole/system:controller:deployment-controller
controlplane $
controlplane $
```

## 2<sup>nd</sup> question:

```
controlplane $  
controlplane $ k create ns ns1  
namespace/ns1 created  
controlplane $  
controlplane $ k create ns ns2  
namespace/ns2 created  
controlplane $  
controlplane $  
controlplane $ k create sa sa1 -n ns1  
serviceaccount/sa1 created  
controlplane $  
controlplane $ k create sa sa2 -n ns2  
serviceaccount/sa2 created  
controlplane $  
controlplane $  
controlplane $ k get ns  
NAME          STATUS    AGE  
app-team1      Active    10m  
default        Active    4d15h  
kube-node-lease Active    4d15h  
kube-public    Active    4d15h  
kube-system    Active    4d15h  
local-path-storage Active    4d15h  
ns1            Active    87s  
ns2            Active    74s  
controlplane $
```

```
controlplane $  
controlplane $ k get sa -n ns1  
NAME          SECRETS  AGE  
default       0        3m25s  
sa1           0        2m15s  
controlplane $  
controlplane $  
controlplane $ k get sa -n ns2  
NAME          SECRETS  AGE  
default       0        3m32s  
sa2           0        2m29s  
controlplane $
```

```
controlplane $  
controlplane $ kubectl create clusterrole sa1 --verb=create,delete --resource=deployment --namespace=ns1  
clusterrole.rbac.authorization.k8s.io/sa1 created  
controlplane $  
controlplane $ kubectl create clusterrole sa2 --verb=create,delete --resource=deployment --namespace=ns2  
clusterrole.rbac.authorization.k8s.io/sa2 created  
controlplane $
```

```
controlplane $  
controlplane $ kubectl create clusterrolebinding sa1-binding --clusterrole=view --clusterrole=sa1 --serviceaccount=ns1:sa1  
clusterrolebinding.rbac.authorization.k8s.io/sa1-binding created  
controlplane $  
controlplane $  
controlplane $ kubectl create clusterrolebinding sa2-binding --clusterrole=view --clusterrole=sa2 --serviceaccount=ns2:sa2  
clusterrolebinding.rbac.authorization.k8s.io/sa2-binding created  
controlplane $
```

```
controlplane $  
controlplane $  
controlplane $ kubectl auth can-i create pod --as=system:serviceaccount:ns1:sa1  
no  
controlplane $  
controlplane $ kubectl auth can-i create deployment --as=system:serviceaccount:ns1:sa1  
yes  
controlplane $  
controlplane $  
controlplane $ kubectl auth can-i create pod --as=system:serviceaccount:ns2:sa2  
no  
controlplane $  
controlplane $ kubectl auth can-i create deployment --as=system:serviceaccount:ns2:sa2  
yes  
controlplane $
```