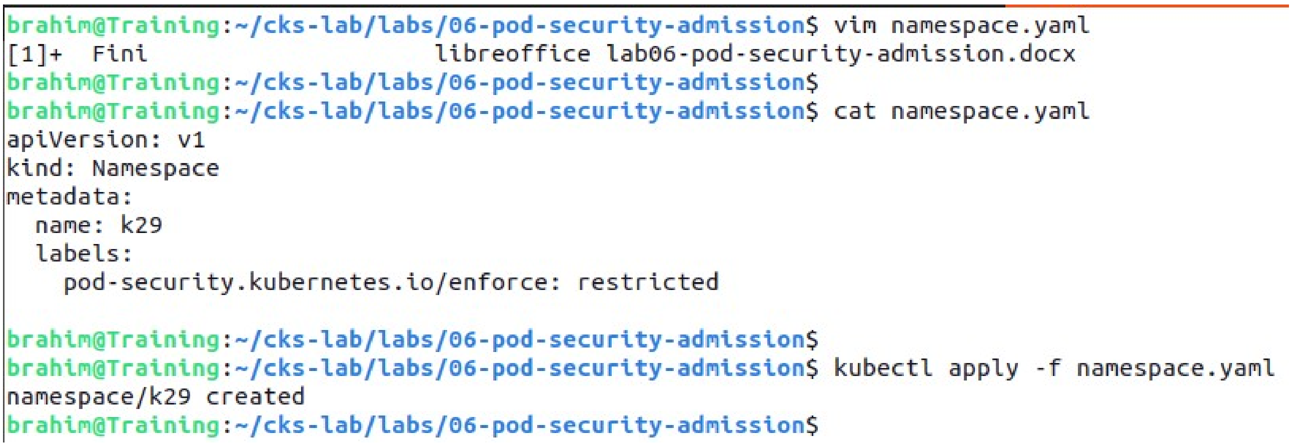
Lab06 - Enforcing a Pod Security Standard Upon Pod Creation

You are tasked with defining a Pod Security Admission rule that should control the creation of Pods in the namespace `k29`.

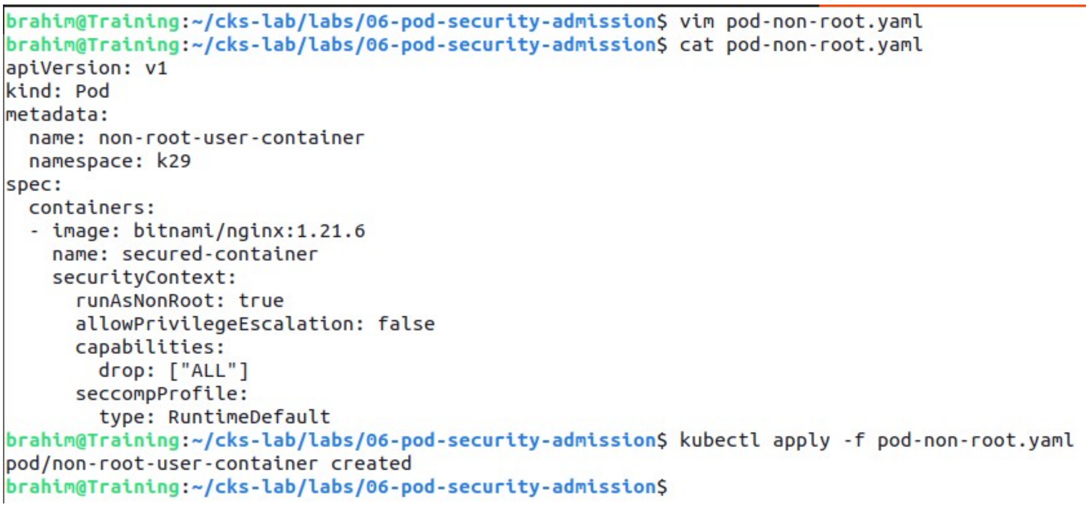
1. Create the namespace `k29`. In the namespace, define a Pod Security Standard (PSS) with the level `restricted` that will cause a Pod to be rejected upon violation.



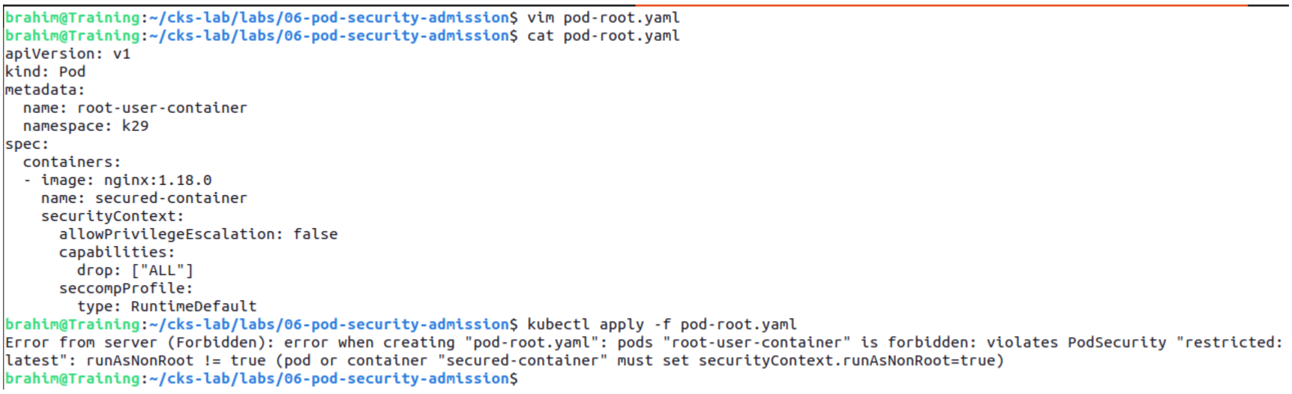
```
brahim@Training:~/cks-lab/labs/06-pod-security-admission$ vim namespace.yaml
[1]+  Fini                    libreoffice lab06-pod-security-admission.docx
brahim@Training:~/cks-lab/labs/06-pod-security-admission$
brahim@Training:~/cks-lab/labs/06-pod-security-admission$ cat namespace.yaml
apiVersion: v1
kind: Namespace
metadata:
  name: k29
  labels:
    pod-security.kubernetes.io/enforce: restricted

brahim@Training:~/cks-lab/labs/06-pod-security-admission$
brahim@Training:~/cks-lab/labs/06-pod-security-admission$ kubectl apply -f namespace.yaml
namespace/k29 created
brahim@Training:~/cks-lab/labs/06-pod-security-admission$
```

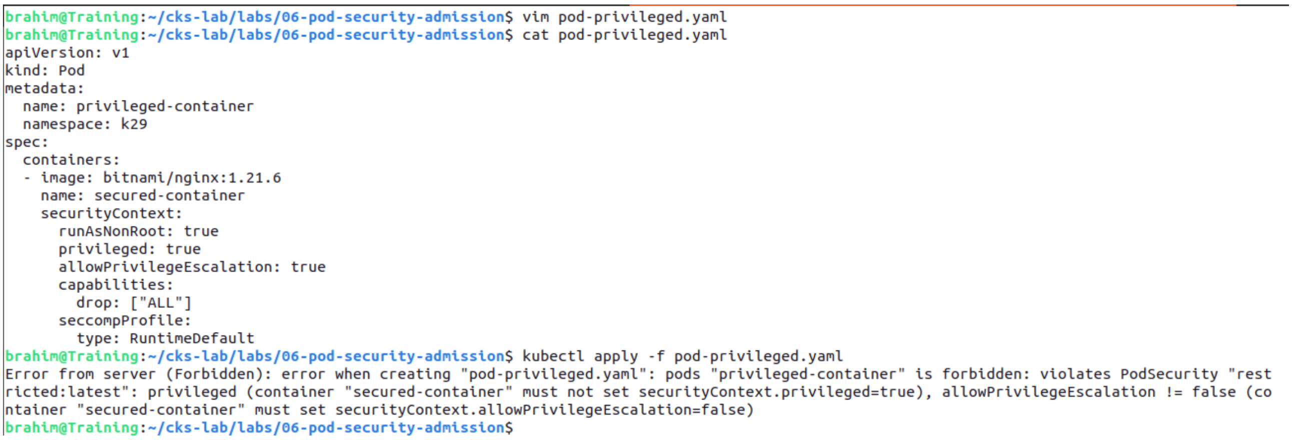
2. Create objects from the YAML manifests `pod-non-root.yaml`, `pod-root.yaml` and `pod-privileged.yaml`.



```
brahim@Training:~/cks-lab/labs/06-pod-security-admission$ vim pod-non-root.yaml
brahim@Training:~/cks-lab/labs/06-pod-security-admission$ cat pod-non-root.yaml
apiVersion: v1
kind: Pod
metadata:
  name: non-root-user-container
  namespace: k29
spec:
  containers:
  - image: bitnami/nginx:1.21.6
    name: secured-container
    securityContext:
      runAsNonRoot: true
      allowPrivilegeEscalation: false
      capabilities:
        drop: ["ALL"]
      seccompProfile:
        type: RuntimeDefault
brahim@Training:~/cks-lab/labs/06-pod-security-admission$ kubectl apply -f pod-non-root.yaml
pod/non-root-user-container created
brahim@Training:~/cks-lab/labs/06-pod-security-admission$
```

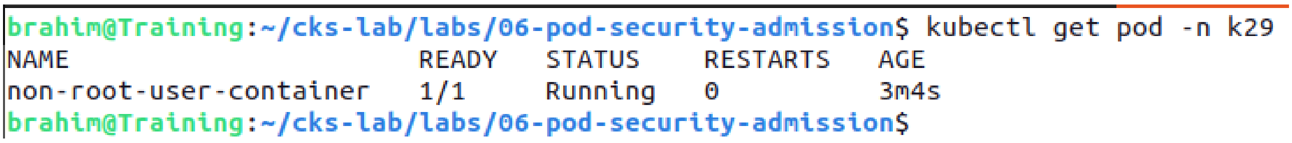


```
brahim@Training:~/cks-lab/labs/06-pod-security-admission$ vim pod-root.yaml
brahim@Training:~/cks-lab/labs/06-pod-security-admission$ cat pod-root.yaml
apiVersion: v1
kind: Pod
metadata:
  name: root-user-container
  namespace: k29
spec:
  containers:
  - image: nginx:1.18.0
    name: secured-container
    securityContext:
      allowPrivilegeEscalation: false
      capabilities:
        drop: ["ALL"]
      seccompProfile:
        type: RuntimeDefault
brahim@Training:~/cks-lab/labs/06-pod-security-admission$ kubectl apply -f pod-root.yaml
Error from server (Forbidden): error when creating "pod-root.yaml": pods "root-user-container" is forbidden: violates PodSecurity "restricted:
latest": runAsNonRoot != true (pod or container "secured-container" must set securityContext.runAsNonRoot=true)
brahim@Training:~/cks-lab/labs/06-pod-security-admission$
```



```
brahim@Training:~/cks-lab/labs/06-pod-security-admission$ vim pod-privileged.yaml
brahim@Training:~/cks-lab/labs/06-pod-security-admission$ cat pod-privileged.yaml
apiVersion: v1
kind: Pod
metadata:
  name: privileged-container
  namespace: k29
spec:
  containers:
  - image: bitnami/nginx:1.21.6
    name: secured-container
    securityContext:
      runAsNonRoot: true
      privileged: true
      allowPrivilegeEscalation: true
      capabilities:
        drop: ["ALL"]
      seccompProfile:
        type: RuntimeDefault
brahim@Training:~/cks-lab/labs/06-pod-security-admission$ kubectl apply -f pod-privileged.yaml
Error from server (Forbidden): error when creating "pod-privileged.yaml": pods "privileged-container" is forbidden: violates PodSecurity "rest
ricted:latest": privileged (container "secured-container" must not set securityContext.privileged=true), allowPrivilegeEscalation != false (co
ntainer "secured-container" must set securityContext.allowPrivilegeEscalation=false)
brahim@Training:~/cks-lab/labs/06-pod-security-admission$
```

- ○ Which of the Pods will be created and why?



```
brahim@Training:~/cks-lab/labs/06-pod-security-admission$ kubectl get pod -n k29
NAME                     READY   STATUS    RESTARTS   AGE
non-root-user-container  1/1     Running   0          3m4s
brahim@Training:~/cks-lab/labs/06-pod-security-admission$
```