Custom SCC Creation

Create **Security Context Constraints (SCC)** and assign it to a specific service account, then run a pod that uses the custom SCC.

oc new-project scc-lab

---

oc create sa demo-sa

Check:

oc get sa

Create SCC

File: custom-scc.yaml

```
apiVersion: security.openshift.io/v1
kind: SecurityContextConstraints
metadata:
  name: custom-scc
allowHostDirVolumePlugin: true
allowPrivilegedContainer: false
runAsUser:
  type: MustRunAsRange
  uidRangeMin: 1000
  uidRangeMax: 2000
seLinuxContext:
  type: MustRunAs
volumes:
- hostPath
- configMap
- emptyDir
users: []
groups: []
```

```yaml
apiVersion: security.openshift.io/v1
kind: SecurityContextConstraints
metadata:
  name: custom-scc1
allowHostDirVolumePlugin: true
allowPrivilegedContainer: false
runAsUser:
  type: MustRunAsRange
  uidRangeMin: 1000
  uidRangeMax: 2000
seLinuxContext:
  type: MustRunAs
volumes:
- hostPath
```

```
[student@workstation ~]$ vim custom-scc1.yaml
[student@workstation ~]$ oc apply -f custom-scc1.yamldsfsfd
error: the path "custom-scc1.yamldsfsfd" does not exist
[student@workstation ~]$ vim custom-scc1.yaml
[student@workstation ~]$ oc apply -f custom-scc1.yaml
securitycontextconstraints.security.openshift.io/custom-scc1 creat
ed
[student@workstation ~]$ oc get scc custom-scc1 -o yaml
allowHostDirVolumePlugin: true
allowHostIPC: false
allowHostNetwork: false
```

Apply:

oc apply -f custom-scc.yaml

Verify:

oc get scc custom-scc -o yaml

SCC to service Account

Assign:

```
oc adm policy add-scc-to-user custom-scc \
  system:serviceaccount:scc-lab:demo-sa
```

Check assignment:

```
oc adm policy who-can use scc custom-scc
```

Pod using SCC

File: pod-using-sa.yaml

```
apiVersion: v1
kind: Pod
metadata:
  name: test-pod
spec:
  serviceAccountName: demo-sa
  containers:
    - name: nginx
      image: nginx
      securityContext:
        runAsUser: 1500
```

Apply:

```
oc apply -f pod-using-sa.yaml
```

Check pod running:

```
oc get pod test-pod
```

Check the SCC actually used:

```
oc describe pod test-pod | grep -i scc
```

```
[student@workstation ~]$ oc get pod test-pod
NAME       READY   STATUS              RESTARTS   AGE
test-pod   0/1     ContainerCreating   0          20s
[student@workstation ~]$ oc describe pod test-pod | grep -i scc
Namespace:        scc-lab1
                  openshift.io/scc: anyuid
  Normal   Scheduled       40s   default-scheduler  Successfully a
ssigned scc-lab1/test-pod to master01
```

Expected:

```
openshift.io/scc: custom-scc
```

If SCC allows hostPath:

File: pod-hostpath.yaml

```
apiVersion: v1
kind: Pod
metadata:
  name: pod-hostpath
spec:
  serviceAccountName: demo-sa
  containers:
    - name: test
      image: centos
      command: ["sleep", "3600"]
      volumeMounts:
        - name: data
          mountPath: /data
  volumes:
    - name: data
      hostPath:
        path: /tmp
```

Apply:

```
oc apply -f pod-hostpath.yaml
```

Describe:

```
oc describe pod pod-hostpath | grep scc
```

```
[student@workstation ~]$ vim pod-hostpath.yaml
[student@workstation ~]$ oc apply -f pod-hostpath.yaml
pod/pod-hostpath created
[student@workstation ~]$ oc describe pod pod-hostpath | grep scc
Namespace:        scc-lab1
                  openshift.io/scc: hostmount-anyuid
  Normal  Scheduled       34s   default-scheduler  Successfully as
signed scc-lab1/pod-hostpath to master01
```

Should show it is using **custom-scc**.

```
oc adm policy remove-scc-from-user custom-scc \
  system:serviceaccount:scc-lab:demo-sa
```

Delete SCC:

```
oc delete scc custom-scc
```
