



CKS^{Q&As}

Certified Kubernetes Security Specialist (CKS) Exam

Pass Linux Foundation CKS Exam with 100% Guarantee

Free Download Real Questions & Answers **PDF** and **VCE** file from:

<https://www.passapply.com/cks.html>

100% Passing Guarantee
100% Money Back Assurance

Following Questions and Answers are all new published by Linux Foundation Official Exam Center

- ⚙️ **Instant Download** After Purchase
- ⚙️ **100% Money Back** Guarantee
- ⚙️ **365 Days** Free Update
- ⚙️ **800,000+** Satisfied Customers





QUESTION 1

Create a network policy named allow-np, that allows pod in the namespace staging to connect to port 80 of other pods in the same namespace.

Ensure that Network Policy:

1.

Does not allow access to pod not listening on port 80.

2.

Does not allow access from Pods, not in namespace staging.

A. See the explanation below:

B. Placeholder

Correct Answer: A

apiVersion: networking.k8s.io/v1

kind: NetworkPolicy

metadata:

name: network-policy

spec:

podSelector: {} #selects all the pods in the namespace deployed policyTypes:

-Ingress ingress:

-ports: #in input traffic allowed only through 80 port only

-protocol: TCP port: 80

QUESTION 2



```
candidate@cli:~$ kubectl config use-context KSSC00401
Switched to context "KSSC00401".
candidate@cli:~$ ssh kssc00401-master
Warning: Permanently added '10.240.86.231' (ECDSA) to the list of known hosts.

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

root@kssc00401-master:~# kubectl get pods -n naboo
NAME          READY   STATUS    RESTARTS   AGE
c-3po         1/1     Running   0           6h48m
chewbacca     1/1     Running   0           6h48m
jawas         1/1     Running   0           6h48m
qui-gon-jinn  1/1     Running   0           6h48m
root@kssc00401-master:~# kubectl get pods -n naboo -o name
pod/c-3po
pod/chewbacca
pod/jawas
pod/qui-gon-jinn
root@kssc00401-master:~# for i in $(kubectl get pods -n naboo -o name)
> do
> kubectl get ${i} -o yaml | grep -i image
> done
Error from server (NotFound): pods "c-3po" not found
Error from server (NotFound): pods "chewbacca" not found
Error from server (NotFound): pods "jawas" not found
Error from server (NotFound): pods "qui-gon-jinn" not found
root@kssc00401-master:~# for i in $(kubectl get pods -n naboo -o name); do kubectl -n naboo
get ${i} -o yaml | grep -i image ; done
  image: centos:centos7.9.2009
  imagePullPolicy: Never
  image: centos:centos7.9.2009
  imageID: docker-pullable://centos@sha256:c73f515d06b0fa07bb18d8202035e739a494ce760aa7312
9f60f4bf2bd22b407
  image: photon:3.0
  imagePullPolicy: Never
  image: photon:3.0
  imageID: docker-pullable://photon@sha256:c48d61f0f3ad19215b75e2087cfbe95d7321abb454e4295
a0e6c38f563ece622
  image: alpine:3.7
  imagePullPolicy: Never
  image: alpine:3.7
  imageID: docker-pullable://alpine@sha256:8421d9a84432575381bfabd248f1eb56f3aa21d9d7cd251
1583c68c9b7511d10
  image: amazonlinux:2
  imagePullPolicy: Never
  image: amazonlinux:2
  imageID: docker-pullable://amazonlinux@sha256:246ef631c75ea83005889621119fd5cc9cbb5500e1
93707c38b6c060d597a146
root@kssc00401-master:~# trivy image centos:centos7.9.2009
2022-05-20T15:39:51.733Z      INFO    Need to update DB
2022-05-20T15:39:51.733Z      INFO    Downloading DB...
27.97 MiB / 27.97 MiB [-----] 100.00% 27.43 MiB p/s 1s
```



```
-----+-----+
root@kssc00401-master:~# for i in $(kubectl get pods -n naboo -o name); do kubectl -n naboo
get ${i} -o yaml | grep -i image ; done
  image: centos:centos7.9.2009
  imagePullPolicy: Never
  image: centos:centos7.9.2009
  imageID: docker-pullable://centos@sha256:c73f515d06b0fa07bb18d8202035e739a494ce760aa7312
9f60f4bf2bd22b407
  image: photon:3.0
  imagePullPolicy: Never
  image: photon:3.0
  imageID: docker-pullable://photon@sha256:c48d61f0f3ad19215b75e2087cfbe95d7321abb454e4295
a0e6c38f563ece622
  image: alpine:3.7
  imagePullPolicy: Never
  image: alpine:3.7
  imageID: docker-pullable://alpine@sha256:8421d9a84432575381bfabd248f1eb56f3aa21d9d7cd251
1583c68c9b7511d10
  image: amazonlinux:2
  imagePullPolicy: Never
  image: amazonlinux:2
  imageID: docker-pullable://amazonlinux@sha256:246ef631c75ea83005889621119fd5cc9cbb5500e1
93707c38b6c060d597a146
root@kssc00401-master:~# trivy image photon:3.0
2022-05-20T15:40:18.003Z      INFO    Detected OS: photon
2022-05-20T15:40:18.003Z      INFO    Detecting Photon Linux vulnerabilities...
2022-05-20T15:40:18.005Z      INFO    Number of language-specific files: 0

photon:3.0 (photon 3.0)
=====
Total: 0 (UNKNOWN: 0, LOW: 0, MEDIUM: 0, HIGH: 0, CRITICAL: 0)
```

```
root@kssc00401-master:~# kubectl get pods -n naboo -o name
pod/c-3po
pod/chewbacca
pod/jawas
pod/qui-gon-jinn
root@kssc00401-master:~# kubectl -n naboo pod/c-3po -o yaml | grep image
Error: flags cannot be placed before plugin name: -n
root@kssc00401-master:~# kubectl -n naboo get pod/c-3po -o yaml | grep image
  image: centos:centos7.9.2009
  imagePullPolicy: Never
  image: centos:centos7.9.2009
  imageID: docker-pullable://centos@sha256:c73f515d06b0fa07bb18d8202035e739a494ce760aa7312
9f60f4bf2bd22b407
root@kssc00401-master:~# kubectl -n naboo delete pod/c-3po
pod "c-3po" deleted
root@kssc00401-master:~# kubectl -n naboo delete pod/jawas
pod "jawas" deleted
```

```
pod "jawas" deleted
root@kssc00401-master:~# history
 1 kubectl get pods -n naboo
 2 kubectl get pods -n naboo -o name
 3 for i in $(kubectl get pods -n naboo -o name); do kubectl get ${i} -o yaml | grep -i
image ; done
 4 for i in $(kubectl get pods -n naboo -o name); do kubectl -n naboo get ${i} -o yaml |
grep -i image ; done
 5 trivy image centos:centos7.9.2009
 6 for i in $(kubectl get pods -n naboo -o name); do kubectl -n naboo get ${i} -o yaml |
grep -i image ; done
 7 trivy image photon:3.0
 8 for i in $(kubectl get pods -n naboo -o name); do kubectl -n naboo get ${i} -o yaml |
grep -i image ; done
 9 trivy image alpine:3.7
10 for i in $(kubectl get pods -n naboo -o name); do kubectl -n naboo get ${i} -o yaml |
grep -i image ; done
11 trivy image amazonlinux:2
12 kubectl get pods -n naboo -o name
13 kubectl -n naboo pod/c-3po -o yaml | grep image
14 kubectl -n naboo get pod/c-3po -o yaml | grep image
15 kubectl -n naboo delete pod/c-3po
16 kubectl -n naboo delete pod/jawas
17 history
root@kssc00401-master:~# █
```



AppArmor is enabled on the cluster's worker node. An AppArmor profile is prepared, but not enforced yet.



You **must** complete this task on the following cluster/nodes:



Cluster	Master node	Worker node
KSSH00401	kssh00401 -master	kssh00401 -worker1

You can switch the cluster/configuration context using the following command:

```
[candidate@cli] $ | kubectl config use-context KSSH00401
```

You may use your browser to open **one additional tab** to access the AppArmor documentation.





Task

On the cluster's worker node, enforce the prepared AppArmor profile located at `/etc/apparmor.d/nginx_apparmor`.

Edit the prepared manifest file located at `/home/candidate/KSSH00401/nginx-pod.yaml` to apply the AppArmor profile.

Finally, apply the manifest file and create the Pod specified in it.

A. See the explanation below

B. Placeholder

Correct Answer: A

QUESTION 3

You must complete this task on the following cluster/nodes: Cluster: immutable-cluster

Master node: master1

Worker node: worker1

You can switch the cluster/configuration context using the following command:

```
[desk@cli] $ kubectl config use-context immutable-cluster
```

Context: It is best practice to design containers to be stateless and immutable.

Task:

Inspect Pods running in namespace `prod` and delete any Pod that is either not stateless or not immutable.

Use the following strict interpretation of stateless and immutable:

1.
Pods being able to store data inside containers must be treated as not stateless.
Note: You don't have to worry whether data is actually stored inside containers or not already.
2.
Pods being configured to be privileged in any way must be treated as potentially not stateless or not immutable.

A. See the explanation below

B. Placeholder

Correct Answer: A



Explanation/Reference:

```
candidate@cli:~$ kubectl config use-context KSRS00501
Switched to context "KSRS00501".
candidate@cli:~$ kubectl get pod -n testing
NAME          READY   STATUS    RESTARTS   AGE
app            1/1     Running   0           6h31m
frontend      1/1     Running   0           6h32m
smtp           1/1     Running   0           6h31m
candidate@cli:~$ kubectl get pod/app -n testing -o yaml
- lastProbeTime: null
  lastTransitionTime: "2022-05-20T08:40:35Z"
  status: "True"
  type: PodScheduled
containerStatuses:
- containerID: docker://11143682c400984c9faf3dff1e056d4b00a7eb1de007fe1834be0a84fal46e18
  image: nginx:latest
  imageID: docker-pullable://nginx@sha256:2d17cc4981bf1e22a87ef3b3dd20fbb72c3868738e3f307662eb40e2630d4320
  lastState: {}
  name: app-container
  ready: true
  restartCount: 0
  started: true
  state:
    running:
      startedAt: "2022-05-20T08:40:37Z"
hostIP: 10.240.86.141
phase: Running
podIP: 10.10.1.3
podIPs:
- ip: 10.10.1.3
qosClass: BestEffort
startTime: "2022-05-20T08:40:35Z"
candidate@cli:~$ kubectl get pod/app -n testing -o yaml | grep -E 'privileged|ReadOnlyFileSy
stem'
  privileged: true
candidate@cli:~$ kubectl get pod/frontend -n testing -o yaml | grep -E 'privileged|ReadOnlyF
ileSystem'
  privileged: false
candidate@cli:~$ kubectl get pod/smtp -n testing -o yaml | grep -E 'privileged|ReadOnlyFileS
ystem'
  privileged: true
candidate@cli:~$ kubectl get pod -n testing -o yaml | grep -i ReadOnly
  readOnlyRootFilesystem: false
  readOnly: true
  readOnlyRootFilesystem: true
  readOnly: true
  readOnlyRootFilesystem: false
  readOnly: true
candidate@cli:~$ kubectl get pod/smtp -n testing -o yaml | grep -E 'privileged|readOnlyRootF
ileSystem'
  privileged: true
candidate@cli:~$ kubectl get pod/app -n testing -o yaml | grep -E 'privileged|readOnlyRootFi
leSystem'
  privileged: true
candidate@cli:~$ kubectl get pod/frontend -n testing -o yaml | grep -E 'privileged|readOnlyR
ootFileSystem'
  privileged: false
candidate@cli:~$ kubectl get pod/frontend -n testing -o yaml | grep -E 'privileged|readOnlyR
ootFileSystem'
  privileged: true
  readOnlyRootFilesystem: false
candidate@cli:~$ kubectl delete pod/app -n testing
pod "app" deleted
candidate@cli:~$ kubectl get pod/smtp -n testing -o yaml | grep -E 'privileged|readOnlyRootF
ilesystem'
  privileged: true
  readOnlyRootFilesystem: false
candidate@cli:~$ kubectl delete pod/smtp -n testing
pod "smtp" deleted
```




QUESTION 4

```
Switched to context "KSSC00202".
candidate@cli:~$ ssh kssc00202-master
Warning: Permanently added '10.177.80.12' (ECDSA) to the list of known hosts.

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

root@kssc00202-master:~# ls /etc/kubernetes/epconfig/
admission_configuration.json  apiserver-client-key.pem  apiserver-client.pem  kubeconfig.yaml  webhook-key.pem  webhook.pem
root@kssc00202-master:~# vim /etc/kubernetes/epconfig/admission_configuration.json
```



```
"imagePolicy": {
  "kubeConfigFile": "/etc/kubernetes/epconfig/kubeconfig.yaml",
  "allowTTL": 50,
  "denyTTL": 50,
  "retryBackoff": 500,
  "defaultAllow": false
```

```
root@kssc00202-master:~# vim /etc/kubernetes/epconfig/admission_configuration.json
root@kssc00202-master:~# vim /etc/kubernetes/epconfig/admission_configuration.json
root@kssc00202-master:~# vim /etc/kubernetes/epconfig/kubeconfig.yaml
```

```
apiVersion: v1
clusters:
- cluster:
    certificate-authority: /etc/kubernetes/epconfig/webhook.pem # CA for verifying the remote service.
    server: https://wakanda.local:8081/image_policy
  name: kubernetes
contexts:
- context:
    cluster: kubernetes
    user: kubernetes-admin
  name: kubernetes-admin@kubernetes
current-context: kubernetes-admin@kubernetes
kind: Config
preferences: {}
users:
- name: kubernetes-admin
  user:
    client-certificate: /etc/kubernetes/epconfig/apiserver-client.pem
    client-key: /etc/kubernetes/epconfig/apiserver-client.pem
```

```
root@kssc00202-master:~# vim /etc/kubernetes/epconfig/admission_configuration.json
root@kssc00202-master:~# vim /etc/kubernetes/epconfig/admission_configuration.json
root@kssc00202-master:~# vim /etc/kubernetes/epconfig/kubeconfig.yaml
root@kssc00202-master:~# vim /etc/kubernetes/manifests/kube-apiserver.yaml p
```

```
apiVersion: v1
kind: Pod
metadata:
  annotations:
    kubeadm.kubernetes.io/kube-apiserver.advertise-address.endpoint: 10.177.80.12:6443
  creationTimestamp: null
  labels:
    component: kube-apiserver
    tier: control-plane
  name: kube-apiserver
  namespace: kube-system
spec:
  containers:
  - command:
    - kube-apiserver
    - --advertise-address=10.177.80.12
    - --allow-privileged=true
    - --authorization-mode=Node,RBAC
    - --client-ca-file=/etc/kubernetes/pki/ca.crt
    - --enable-admission-plugins=NodeRestriction
    - --enable-bootstrap-token-auth=true
    - --etcd-cafile=/etc/kubernetes/pki/etcd/ca.crt
    - --etcd-certfile=/etc/kubernetes/pki/apiserver-etcd-client.crt
    - --etcd-keyfile=/etc/kubernetes/pki/apiserver-etcd-client.key
    - --etcd-servers=https://127.0.0.1:2379
    - --kubelet-client-certificate=/etc/kubernetes/pki/apiserver-kubelet-client.crt
    - --kubelet-client-key=/etc/kubernetes/pki/apiserver-kubelet-client.key
    - --kubelet-preferred-address-types=InternalIP,ExternalIP,Hostname
    - --proxy-client-cert-file=/etc/kubernetes/pki/front-proxy-client.crt
    - --proxy-client-key-file=/etc/kubernetes/pki/front-proxy-client.key
    - --requestheader-allowed-names=front-proxy-client
    - --requestheader-client-ca-file=/etc/kubernetes/pki/front-proxy-ca.crt
    - --requestheader-extra-headers-prefix=X-Remote-Extra-
  "/etc/kubernetes/manifests/kube-apiserver.yaml" 135L, 4626C
```

```
root@kssc00202-master:~# vim /etc/kubernetes/manifests/kube-apiserver.yaml p
2 files to edit
root@kssc00202-master:~# rm -f p
root@kssc00202-master:~# vim /etc/kubernetes/manifests/kube-apiserver.yaml
```



```
apiVersion: v1
kind: Pod
metadata:
  annotations:
    kubeadm.kubernetes.io/kube-apiserver.advertise-address.endpoint: 10.177.80.12:6443
  creationTimestamp: null
  labels:
    component: kube-apiserver
    tier: control-plane
  name: kube-apiserver
  namespace: kube-system
spec:
  containers:
    - command:
        - kube-apiserver
        - --advertise-address=10.177.80.12
        - --allow-privileged=true
        - --authorization-mode=Node,RBAC
        - --client-ca-file=/etc/kubernetes/pki/ca.crt
        - --enable-admission-plugins=NodeRestriction,ImagePolicyWebHook
        - --admission-control-config-file=/etc/kubernetes/epconfig/admin.conf
        - --enable-bootstrap-token-auth=true
        - --etcd-cafile=/etc/kubernetes/pki/etcd/ca.crt
        - --etcd-certfile=/etc/kubernetes/pki/apiserver-etcd-client.crt
        - --etcd-keyfile=/etc/kubernetes/pki/apiserver-etcd-client.key
        - --etcd-servers=https://127.0.0.1:2379
        - --kubelet-client-certificate=/etc/kubernetes/pki/apiserver-kubelet-client.crt
        - --kubelet-client-key=/etc/kubernetes/pki/apiserver-kubelet-client.key
        - --kubelet-preferred-address-types=InternalIP,ExternalIP,Hostname
        - --proxy-client-cert-file=/etc/kubernetes/pki/front-proxy-client.crt
        - --proxy-client-key-file=/etc/kubernetes/pki/front-proxy-client.key
        - --requestheader-allowed-names=front-proxy-client
        - --requestheader-client-ca-file=/etc/kubernetes/pki/front-proxy-ca.crt
```

```
root@kssc00202-master:~# rm -f p
root@kssc00202-master:~# vim /etc/kubernetes/manifests/kube-apiserver.yaml
root@kssc00202-master:~# systemctl daemon-reload
root@kssc00202-master:~#
root@kssc00202-master:~#
root@kssc00202-master:~#
root@kssc00202-master:~# systemctl restart kubelet.service
root@kssc00202-master:~# systemctl enable kubelet.service
root@kssc00202-master:~#
root@kssc00202-master:~#
root@kssc00202-master:~#
root@kssc00202-master:~# ls
KSSC00202 snap
root@kssc00202-master:~# cat KSSC00202/vulnerable-resource.yaml
```

```
KSSC00202 snap
root@kssc00202-master:~# cat KSSC00202/vulnerable-resource.yaml
---
apiVersion: v1
kind: ReplicationController
metadata:
  name: nginx-latest
spec:
  replicas: 1
  selector:
    app: nginx-latest
  template:
    metadata:
      name: nginx-latest
      labels:
        app: nginx-latest
    spec:
      containers:
        - name: nginx-latest
          image: nginx
          ports:
            - containerPort: 80
root@kssc00202-master:~# kubectcl create -f KSSC00202/vulnerable-resource.yaml
```

```
root@kssc00202-master:~# kubectcl create -f KSSC00202/vulnerable-resource.yaml
The connection to the server 10.177.80.12:6443 was refused - did you specify the right host or port?
root@kssc00202-master:~# kubectcl get pods
The connection to the server 10.177.80.12:6443 was refused - did you specify the right host or port?
root@kssc00202-master:~# ls -al .kube/
total 20
drwxr-xr-x 3 root root 4096 Aug 3 04:07 .
drwxr----- 9 root root 4096 Oct 11 15:36 ..
drwxr-xr-x 4 root root 4096 Aug 3 04:07 cache
-rw-r--r-- 1 root root 5636 Aug 3 04:07 config
root@kssc00202-master:~# crictl ps -a
```

```
012ea8587130e a634548d10b03 2 months ago Exited kube-proxy 0 1460a9f
a0f1e8 kube-proxy-cmjb5
405227dfa49d0 a6be758cef4cd 2 months ago Exited etcd 0 cfb6522
e720fb etcd-kssc00202-master
root@kssc00202-master:~# ls -al .kube/ | grep kube-api
root@kssc00202-master:~# crictl ps -a | grep kube-api
WARN[0000] runtime connect using default endpoints: [unix:///var/run/dockerhim.sock unix:///run/containerd/containerd.sock unix:///run/crio.sock unix:///var/run/cri-dockerd.sock]. As the default settings are now deprecated, you should set the endpoint instead.
WARN[0000] unable to determine runtime API version: rpc error: code = Unavailable desc = connection error: desc = "transport: Error wh
ile dialing dial unix /var/run/dockerhim.sock: connect: no such file or directory"
WARN[0000] image connect using default endpoints: [unix:///var/run/dockerhim.sock unix:///run/containerd/containerd.sock unix:///run/crio.sock unix:///var/run/cri-dockerd.sock]. As the default settings are now deprecated, you should set the endpoint instead.
WARN[0000] unable to determine image API version: rpc error: code = Unavailable desc = connection error: desc = "transport: Error whil
e dialing dial unix /var/run/dockerhim.sock: connect: no such file or directory"
a003b3dfb61c d3377fb7177c 30 seconds ago Exited kube-apiserver 3 2dad4e
984a91 kube-apiserver-kssc00202-master
5e70b0a70f9ed d3377fb7177c 7 hours ago Exited kube-apiserver 0 68a9f31
6c2559 kube-apiserver-kssc00202-master
root@kssc00202-master:~#
root@kssc00202-master:~#
root@kssc00202-master:~# exit
logout
Connection to 10.177.80.12 closed.
candidate@cli:~$
```



Cluster: dev Master node: master1 Worker node: worker1 You can switch the cluster/configuration context using the following command: [desk@cli] \$ kubectl config use-context dev Task:

Retrieve the content of the existing secret named adam in the safe namespace.

Store the username field in a file named /home/cert-masters/username.txt, and the password field in a file named /home/cert-masters/password.txt.

1.

You must create both files; they don't exist yet.

2.

Do not use/modify the created files in the following steps, create new temporary files if needed.

Create a new secret named newsecret in the safe namespace, with the following content:

Username: dbadmin Password: moresecurepas

Finally, create a new Pod that has access to the secret newsecret via a volume:

Namespace:safe Pod name:mysecret-pod Container name:db-container Image:redis Volume name:secret-vol Mount path:/etc/mysecret

A. See the explanation below

B. Placeholder

Correct Answer: A

QUESTION 5

A CIS Benchmark tool was run against the kubeadm-created cluster and found multiple issues that must be addressed immediately.



You **must** complete this task on the following cluster/nodes:




Cluster	Master node	Worker node
KSCS00201	kscs00201-master	kscs00201-worker1

You can switch the cluster/configuration context using the following command:

```
[candidate@cli] $ | kubectl config use-context KSCS00201
```

Fix all issues via configuration and restart the affected components to ensure the new settings take effect. Fix all of the following violations that were found against the API server:



1.2.7 Ensure that the
--authorization
-mode
argument is not
set to
AlwaysAllow
FAIL

1.2.8 Ensure that the
--authorization
-mode
argument
includes Node
FAIL

1.2.9 Ensure that the
--authorization
-mode
argument
includes RBAC
FAIL

Fix all of the following violations that were found against the Kubelet: Fix all of the following violations that were found against etcd:



- 4.2.1 Ensure that the `anonymous-auth` argument is set to `false` **FAIL**
- 4.2.2 Ensure that the `--authorization-mode` argument is not set to `AlwaysAllow` **FAIL**

Use `Webhook` authentication/authorization where possible.



2.2 Ensure that the
--client-cert-auth
argument is set
to true

FAIL



A. See explanation below.

B. Placeholder

Correct Answer: A



```
candidate@cli:~$ kubectl delete sa/podrunner -n qa
serviceaccount "podrunner" deleted
candidate@cli:~$ kubectl config use-context KSCS00201
Switched to context "KSCS00201".
candidate@cli:~$ ssh kscs00201-master
Warning: Permanently added '10.240.86.194' (ECDSA) to the list of known hosts.

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

root@kscs00201-master:~# vim /etc/kubernetes/manifests/kube-apiserver.yaml
root@kscs00201-master:~# systemctl daemon-reload
root@kscs00201-master:~# systemctl restart kubelet.service
root@kscs00201-master:~# systemctl enable kubelet.service
root@kscs00201-master:~# systemctl status kubelet.service
● kubelet.service - kubelet: The Kubernetes Node Agent
   Loaded: loaded (/lib/systemd/system/kubelet.service; enabled; vendor preset: enabled)
   Drop-In: /etc/systemd/system/kubelet.service.d
            └─10-kubeadm.conf
   Active: active (running) since Fri 2022-05-20 14:19:31 UTC; 29s ago
     Docs: https://kubernetes.io/docs/home/
    Main PID: 134205 (kubelet)
      Tasks: 16 (limit: 76200)
     Memory: 39.5M
    CGroup: /system.slice/kubelet.service
            └─134205 /usr/bin/kubelet --bootstrap-kubeconfig=/etc/kubernetes/bootstrap-kub

May 20 14:19:35 kscs00201-master kubelet[134205]: I0520 14:19:35.420825 134205 reconciler.>
May 20 14:19:35 kscs00201-master kubelet[134205]: I0520 14:19:35.420863 134205 reconciler.>
May 20 14:19:35 kscs00201-master kubelet[134205]: I0520 14:19:35.420907 134205 reconciler.>
May 20 14:19:35 kscs00201-master kubelet[134205]: I0520 14:19:35.420928 134205 reconciler.>
May 20 14:19:36 kscs00201-master kubelet[134205]: I0520 14:19:36.572353 134205 request.go:>
May 20 14:19:37 kscs00201-master kubelet[134205]: I0520 14:19:37.112347 134205 prober_mana>
May 20 14:19:37 kscs00201-master kubelet[134205]: E0520 14:19:37.185076 134205 kubelet.go:>
May 20 14:19:37 kscs00201-master kubelet[134205]: I0520 14:19:37.645798 134205 kubelet.go:>
May 20 14:19:38 kscs00201-master kubelet[134205]: I0520 14:19:38.184062 134205 kubelet.go:>
May 20 14:19:40 kscs00201-master kubelet[134205]: I0520 14:19:40.036042 134205 prober_mana>
lines 1-22/22 (END)
```



```
de Agent
et.service; enabled; vendor preset: enabled)
ce.d

5-20 14:19:31 UTC; 29s ago

trap-kubeconfig=/etc/kubernetes/bootstrap-kubelet.conf --kubeconfig=/etc/kubernetes/kubelet.
5]: I0520 14:19:35.420825 134205 reconciler.go:221] "operationExecutor.VerifyControllerAtt
5]: I0520 14:19:35.420863 134205 reconciler.go:221] "operationExecutor.VerifyControllerAtt
5]: I0520 14:19:35.420907 134205 reconciler.go:221] "operationExecutor.VerifyControllerAtt
5]: I0520 14:19:35.420928 134205 reconciler.go:157] "Reconciler: start to sync state"
5]: I0520 14:19:36.572353 134205 request.go:665] "Waited for 1.049946364s due to client-side
5]: I0520 14:19:37.112347 134205 prober_manager.go:255] "Failed to trigger a manual run" probe="
5]: E0520 14:19:37.185076 134205 kubelet.go:1711] "Failed creating a mirror pod for" err="p
5]: I0520 14:19:37.645798 134205 kubelet.go:1693] "Trying to delete pod" pod="kube-system/k
5]: I0520 14:19:38.184062 134205 kubelet.go:1698] "Deleted mirror pod because it is outdate
5]: I0520 14:19:40.036042 134205 prober_manager.go:255] "Failed to trigger a manual run" probe="
~
~
lines 1-22/22 (END)
```

```
let.conf --kubeconfig=/etc/kubernetes/kubelet.conf --config=/var/lib/kubelet/config.yaml --
o:221] "operationExecutor.VerifyControllerAttachedVolume started for volume \"kube-proxy\"
o:221] "operationExecutor.VerifyControllerAttachedVolume started for volume \"lib-modules\"
o:221] "operationExecutor.VerifyControllerAttachedVolume started for volume \"flannel-cfg\"
o:157] "Reconciler: start to sync state"
65] "Waited for 1.049946364s due to client-side throttling, not priority and fairness, request
er.go:255] "Failed to trigger a manual run" probe="Readiness"
711] "Failed creating a mirror pod for" err="pods \"kube-apiserver-kscs00201-master\" already
693] "Trying to delete pod" pod="kube-system/kube-apiserver-kscs00201-master" podUID=bb91e1b
698] "Deleted mirror pod because it is outdated" pod="kube-system/kube-apiserver-kscs00201-
er.go:255] "Failed to trigger a manual run" probe="Readiness"
~
~
root@kscs00201-master:~# vim /var/lib/kubelet/config.yaml
```

```
apiVersion: kubelet.config.k8s.io/v1beta1
authentication:
  anonymous:
    enabled: false
  webhook:
    cacheTTL: 0s
    enabled: true
  x509:
    clientCAFile: /etc/kubernetes/pki/ca.crt
authorization:
  mode: Webhook
  webhook:
    cacheAuthorizedTTL: 0s
    cacheUnauthorizedTTL: 0s
cgroupDriver: systemd
clusterDNS:
```

```
~
root@kscs00201-master:~# vim /var/lib/kubelet/config.yaml
root@kscs00201-master:~# vim /var/lib/kubelet/config.yaml
root@kscs00201-master:~# vim /etc/kubernetes/manifests/etcd.yaml
root@kscs00201-master:~# systemctl daemon-reload
root@kscs00201-master:~# systemctl restart kubelet.service
root@kscs00201-master:~# systemctl status kubelet.service
```




```
● kubelet.service - kubelet: The Kubernetes Node Agent
   Loaded: loaded (/lib/systemd/system/kubelet.service; enabled; vendor preset: enabled)
   Drop-In: /etc/systemd/system/kubelet.service.d
            └─10-kubeadm.conf
   Active: active (running) since Fri 2022-05-20 14:22:29 UTC; 4s ago
     Docs: https://kubernetes.io/docs/home/
   Main PID: 135849 (kubelet)
    Tasks: 17 (limit: 76200)
   Memory: 38.0M
   CGroup: /system.slice/kubelet.service
            └─135849 /usr/bin/kubelet --bootstrap-kubeconfig=/etc/kubernetes/bootstrap-kub>

May 20 14:22:30 kscs00201-master kubelet[135849]: I0520 14:22:30.330232 135849 reconciler.>
May 20 14:22:30 kscs00201-master kubelet[135849]: I0520 14:22:30.330259 135849 reconciler.>
May 20 14:22:30 kscs00201-master kubelet[135849]: I0520 14:22:30.330304 135849 reconciler.>
May 20 14:22:30 kscs00201-master kubelet[135849]: I0520 14:22:30.330354 135849 reconciler.>
May 20 14:22:30 kscs00201-master kubelet[135849]: I0520 14:22:30.330378 135849 reconciler.>
May 20 14:22:30 kscs00201-master kubelet[135849]: I0520 14:22:30.330397 135849 reconciler.>
May 20 14:22:30 kscs00201-master kubelet[135849]: I0520 14:22:30.330415 135849 reconciler.>
May 20 14:22:30 kscs00201-master kubelet[135849]: I0520 14:22:30.330433 135849 reconciler.>
May 20 14:22:30 kscs00201-master kubelet[135849]: I0520 14:22:30.330452 135849 reconciler.>
May 20 14:22:30 kscs00201-master kubelet[135849]: I0520 14:22:30.330463 135849 reconciler.>
lines 1-22/22 (END)

May 20 14:22:30 kscs00201-master kubelet[135849]: I0520 14:22:30.330463 135849 reconciler.>
root@kscs00201-master:~#
root@kscs00201-master:~#
root@kscs00201-master:~#
root@kscs00201-master:~# exit
logout
Connection to 10.240.86.194 closed.
candidate@cli:~$
```

[Latest CKS Dumps](#)[CKS Study Guide](#)[CKS Braindumps](#)