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.pass4itsure.com/cks.html

100% Passing Guarantee
100% Money Back Assurance

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

QUESTION 1

```
candidate@cli:~$ kubectl config use-context KSSH00301
Switched to context "KSSH00301".
candidate@cli:~$
candidate@cli:~$
candidate@cli:~$ kubectl get ns dev-team --show-labels
NAME       STATUS   AGE     LABELS
dev-team   Active   6h39m   environment=dev,kubernetes.io/metadata.name=dev-team
candidate@cli:~$ kubectl get pods -n dev-team --show-labels
NAME            READY   STATUS    RESTARTS   AGE     LABELS
users-service   1/1     Running   0          6h40m   environment=dev
candidate@cli:~$ ls
KSCH00301  KSMV00102  KSSC00301  KSSH00401     test-secret-pod.yaml
KSCS00101  KSMV00301  KSSH00301  password.txt  username.txt
candidate@cli:~$ vim np.yaml
```

```
apiVersion: networking.k8s.io/v1
kind: NetworkPolicy
metadata:
  name: pod-access
  namespace: dev-team
spec:
  podSelector:
    matchLabels:
      environment: dev
  policyTypes:
    - Ingress
  ingress:
    - from:
        - namespaceSelector:
            matchLabels:
              environment: dev
        - podSelector:
            matchLabels:
              environment: testing
```

```
candidate@cli:~$ vim np.yaml
candidate@cli:~$ cat np.yaml
apiVersion: networking.k8s.io/v1
kind: NetworkPolicy
metadata:
  name: pod-access
  namespace: dev-team
spec:
  podSelector:
    matchLabels:
      environment: dev
  policyTypes:
    - Ingress
  ingress:
    - from:
        - namespaceSelector:
            matchLabels:
              environment: dev
        - podSelector:
            matchLabels:
              environment: testing
candidate@cli:~$
candidate@cli:~$
candidate@cli:~$ kubectl create -f np.yaml -n dev-team
networkpolicy.networking.k8s.io/pod-access created
candidate@cli:~$ kubectl describe netpol -n dev-team
Name:         pod-access
Namespace:    dev-team
Created on:   2022-05-20 15:35:33 +0000 UTC
Labels:       <none>
Annotations:  <none>
Spec:
  PodSelector:     environment=dev
  Allowing ingress traffic:
    To Port: <any> (traffic allowed to all ports)
    From:
      NamespaceSelector: environment=dev
    From:
      PodSelector: environment=testing
  Not affecting egress traffic
  Policy Types: Ingress
candidate@cli:~$ cat KSSH00301/network-policy.yaml
---
apiVersion: networking.k8s.io/v1
kind: NetworkPolicy
metadata:
  name: ""
  namespace: ""
spec:
  podSelector: {}
  policyTypes:
    - Ingress
  ingress:
    - from: []
    - from: []
candidate@cli:~$ cp np.yaml KSSH00301/network-policy.yaml
candidate@cli:~$ cat KSSH00301/network-policy.yaml
```

```
candidate@cli:~$ cat KSSH00301/network-policy.yaml
apiVersion: networking.k8s.io/v1
kind: NetworkPolicy
metadata:
  name: pod-access
  namespace: dev-team
spec:
  podSelector:
    matchLabels:
      environment: dev
  policyTypes:
    - Ingress
  ingress:
    - from:
        - namespaceSelector:
            matchLabels:
              environment: dev
        - podSelector:
            matchLabels:
              environment: testing
candidate@cli:~$
```

1.

Retrieve the content of the existing secret named default-token-xxxxx in the testing namespace.

Store the value of the token in the token.txt

2.

Create a new secret named test-db-secret in the DB namespace with the following content:

username: mysql password: password@123

Create the Pod name test-db-pod of image nginx in the namespace db that can access test-db-secret via a volume at path /etc/mysql-credentials

A. See the explanation below:

B. PlaceHolder

Correct Answer: A

To add a Kubernetes cluster to your project, group, or instance:

1.

Navigate to your:

2.

Click Add Kubernetes cluster.

3.

Click the Add existing cluster tab and fill in the details:

Get the API URL by running this command:

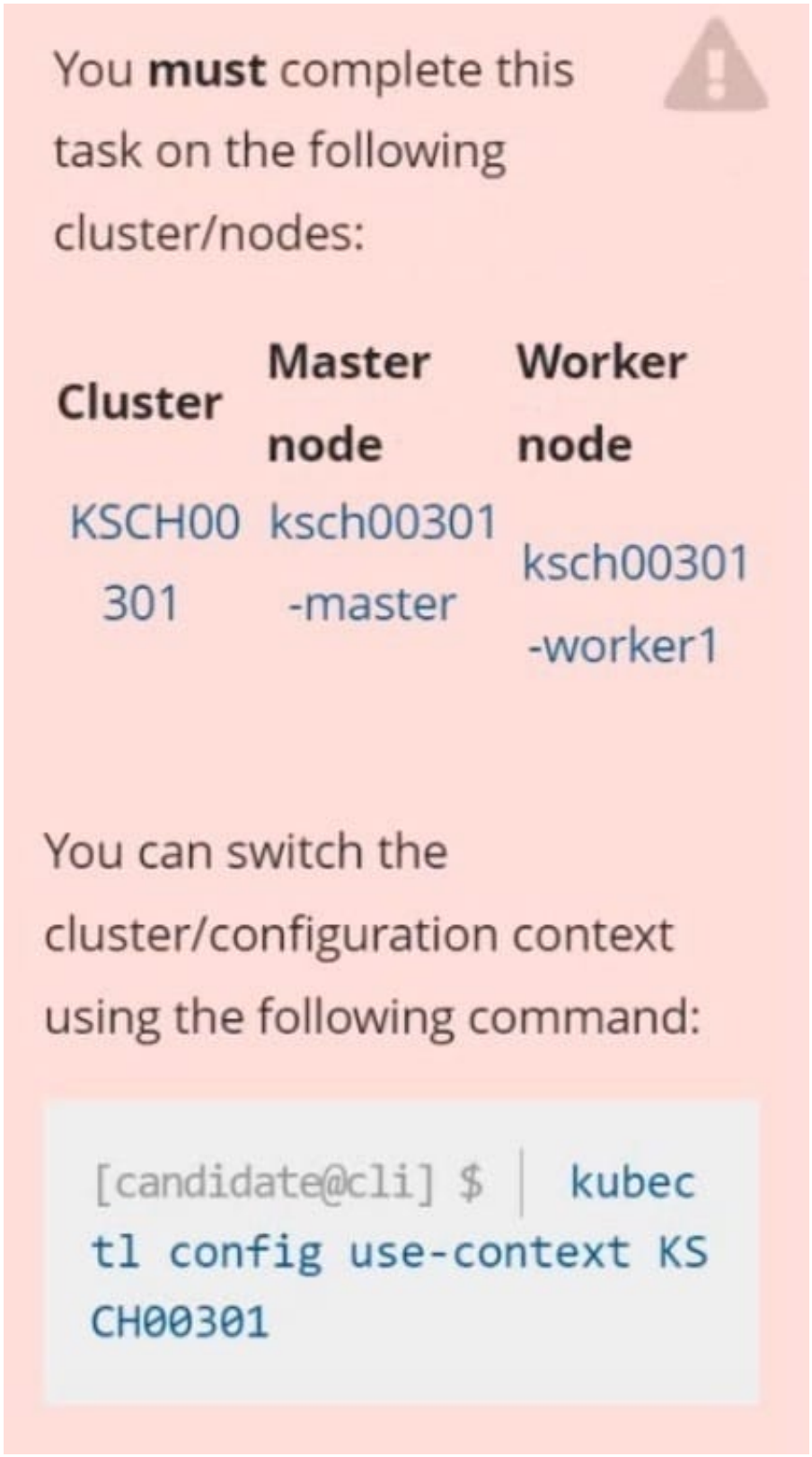
kubectl cluster-info | grep -E \\'Kubernetes master|Kubernetes control plane\\' | awk \\'/http/ {print $NF}\\'

uk.co.certification.simulator.questionpool.PList@113e1f90

kubectl get secret -o jsonpath="{[\\'data\\'][\\'ca\.crt\\']}"

QUESTION 2

CORRECT TEXT Your organization\\'s security policy includes:



1.

ServiceAccounts must not automount API credentials

2.

ServiceAccount names must end in "-sa"

The Pod specified in the manifest file /home/candidate/KSCH00301 /pod-m

nifest.yaml fails to schedule because of an incorrectly specified ServiceAccount.

Complete the following tasks:

Task

1.

Create a new ServiceAccount named frontend-sa in the existing namespace qa. Ensure the ServiceAccount does not automount API credentials.

2.

Using the manifest file at /home/candidate/KSCH00301 /pod-manifest.yaml, create the Pod.

3.

Finally, clean up any unused ServiceAccounts in namespace qa.

A. See the explanation below

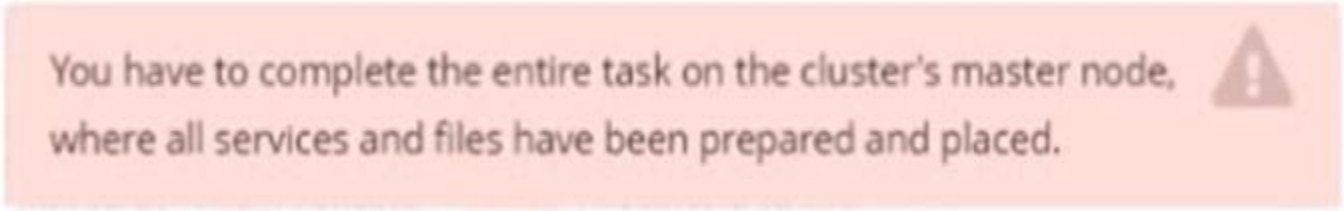
B. PlaceHolder

Correct Answer: A

QUESTION 3

CORRECT TEXT

A container image scanner is set up on the cluster, but it\\'s not yet fully integrated into the cluster s configuration. When complete, the container image scanner shall scan for and reject the use of vulnerable images.



Given an incomplete configuration in directory /etc/kubernetes/epconfig and a functional container image scanner with HTTPS endpoint https://wakanda.local:8081 /image_policy:

1.

Enable the necessary plugins to create an image policy

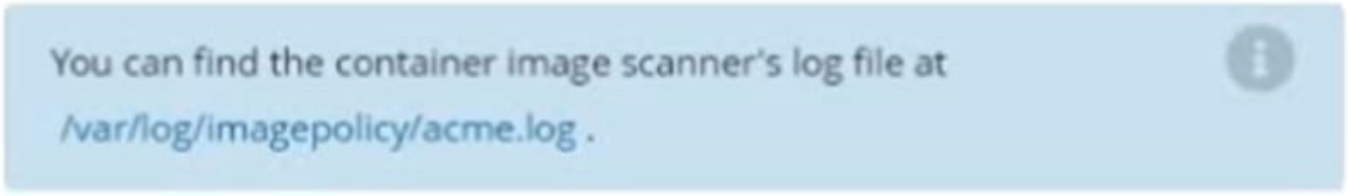
2.

Validate the control configuration and change it to an implicit deny

3.

Edit the configuration to point to the provided HTTPS endpoint correctly

Finally, test if the configuration is working by trying to deploy the vulnerable resource /root/KSSC00202/vulnerable-resource.yml.



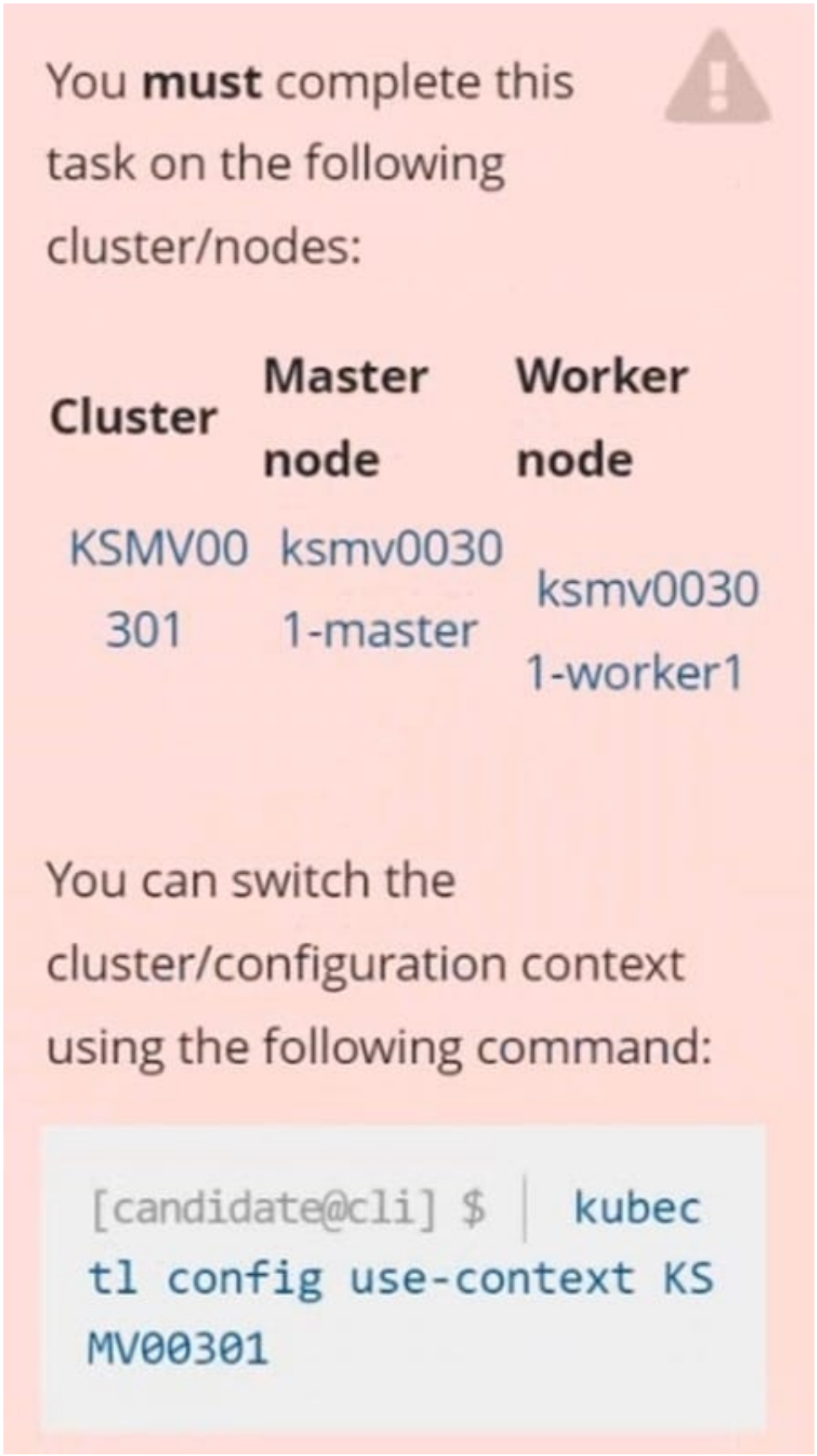
A. See the explanation below

B. PlaceHolder

Correct Answer: A

QUESTION 4

CORRECT TEXT


You must complete this task on the following cluster/nodes:
Cluster
Master node
Worker node
KSMV00301
ksmv0030 1-master
ksmv0030 1-worker1
You can switch the cluster/configuration context using the following command:
[candidate@cli] $ kubectl config use-context KSMV00301

Context

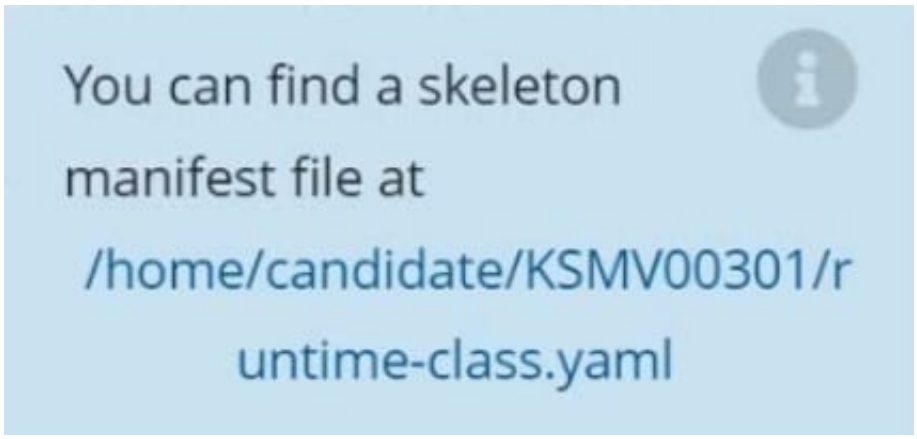
This cluster uses containerd as CRI runtime.

Containerd\\'s default runtime handler is runc. Containerd has been prepared to support an additional runtime handler, runsc (gVisor).

Task

Create a RuntimeClass named sandboxed using the prepared runtime handler named runsc.

Update all Pods in the namespace server to run on gVisor.



A. See the explanation below

B. PlaceHolder

Correct Answer: A

```
candidate@cli:~$ kubectl config use-context KSMV00301
Switched to context "KSMV00301".
candidate@cli:~$ cat /home/candidate/KSMV00301/runtime-class.yaml
---
apiVersion: node.k8s.io/v1
kind: RuntimeClass
metadata:
  name: ""
handler: ""
candidate@cli:~$ vim /home/candidate/KSMV00301/runtime-class.yaml
```

```
---
apiVersion: node.k8s.io/v1
kind: RuntimeClass
metadata:
  name: "sandboxed"
handler: "runsc"
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
:wq!
```

```
candidate@cli:~$ kubectl config use-context KSMV00301
Switched to context "KSMV00301".
candidate@cli:~$ cat /home/candidate/KSMV00301/runtime-class.yaml
---
apiVersion: node.k8s.io/v1
kind: RuntimeClass
metadata:
  name: ""
handler: ""
candidate@cli:~$ vim /home/candidate/KSMV00301/runtime-class.yaml
candidate@cli:~$ cat /home/candidate/KSMV00301/runtime-class.yaml
---
apiVersion: node.k8s.io/v1
kind: RuntimeClass
metadata:
  name: "sandboxed"
handler: "runsc"
candidate@cli:~$ kubectl get deployments.apps -n server
NAME        READY   UP-TO-DATE   AVAILABLE   AGE
workload1   1/1     1            1           5h43m
workload2   1/1     1            1           5h43m
workload3   1/1     1            1           5h43m
candidate@cli:~$ kubectl get pods -n server
NAME                         READY   STATUS    RESTARTS   AGE
workload1-6869857dd7-s45rc   1/1     Running   0          5h43m
workload2-d4bd497d5-h44df    1/1     Running   0          5h43m
workload3-8587774495-chm56   1/1     Running   0          5h43m
candidate@cli:~$ kubectl -n server edit deployments.apps workload1
```

```
  template:
    metadata:
      creationTimestamp: null
      labels:
        app: nginx
      name: workload1
    spec:
      runtimeClassName: sandboxed
      containers:
      - image: nginx:1.14.2
        imagePullPolicy: IfNotPresent
        name: workload1
        ports:
        - containerPort: 80
          protocol: TCP
        resources: {}
        terminationMessagePath: /dev/termination-log
        terminationMessagePolicy: File
      dnsPolicy: ClusterFirst
      restartPolicy: Always
      schedulerName: default-scheduler
      securityContext: {}
      terminationGracePeriodSeconds: 30
status:
"/tmp/kubectl-edit-3385772700.yaml"
```

```
NAME                         READY   STATUS    RESTARTS   AGE
workload1-6869857dd7-s45rc   1/1     Running   0          5h44m
workload2-d4bd497d5-h44df    1/1     Running   0          5h44m
workload3-8587774495-chm56   1/1     Running   0          5h44m
candidate@cli:~$ kubectl -n server edit deployments.apps workload1
Edit cancelled, no changes made.
candidate@cli:~$ kubectl get pods -n server
NAME                         READY   STATUS    RESTARTS   AGE
workload1-6869857dd7-s45rc   1/1     Running   0          5h45m
workload2-d4bd497d5-h44df    1/1     Running   0          5h44m
workload3-8587774495-chm56   1/1     Running   0          5h44m
candidate@cli:~$ kubectl -n server edit deployments.apps workload2
Edit cancelled, no changes made.
candidate@cli:~$ kubectl create -f /home/candidate/KSMV00301/runtime-class.yaml
runtimeclass.node.k8s.io/sandboxed created
candidate@cli:~$ kubectl get pods -n server
NAME                         READY   STATUS    RESTARTS   AGE
workload1-6869857dd7-s45rc   1/1     Running   0          5h45m
workload2-d4bd497d5-h44df    1/1     Running   0          5h45m
workload3-8587774495-chm56   1/1     Running   0          5h45m
candidate@cli:~$ kubectl -n server edit deployments.apps workload2
```

```
  strategy:
    rollingUpdate:
      maxSurge: 25%
      maxUnavailable: 25%
    type: RollingUpdate
  template:
    metadata:
      creationTimestamp: null
      labels:
        app: nginx
      name: workload2
    spec:
      runtimeClassName: sandboxed
```

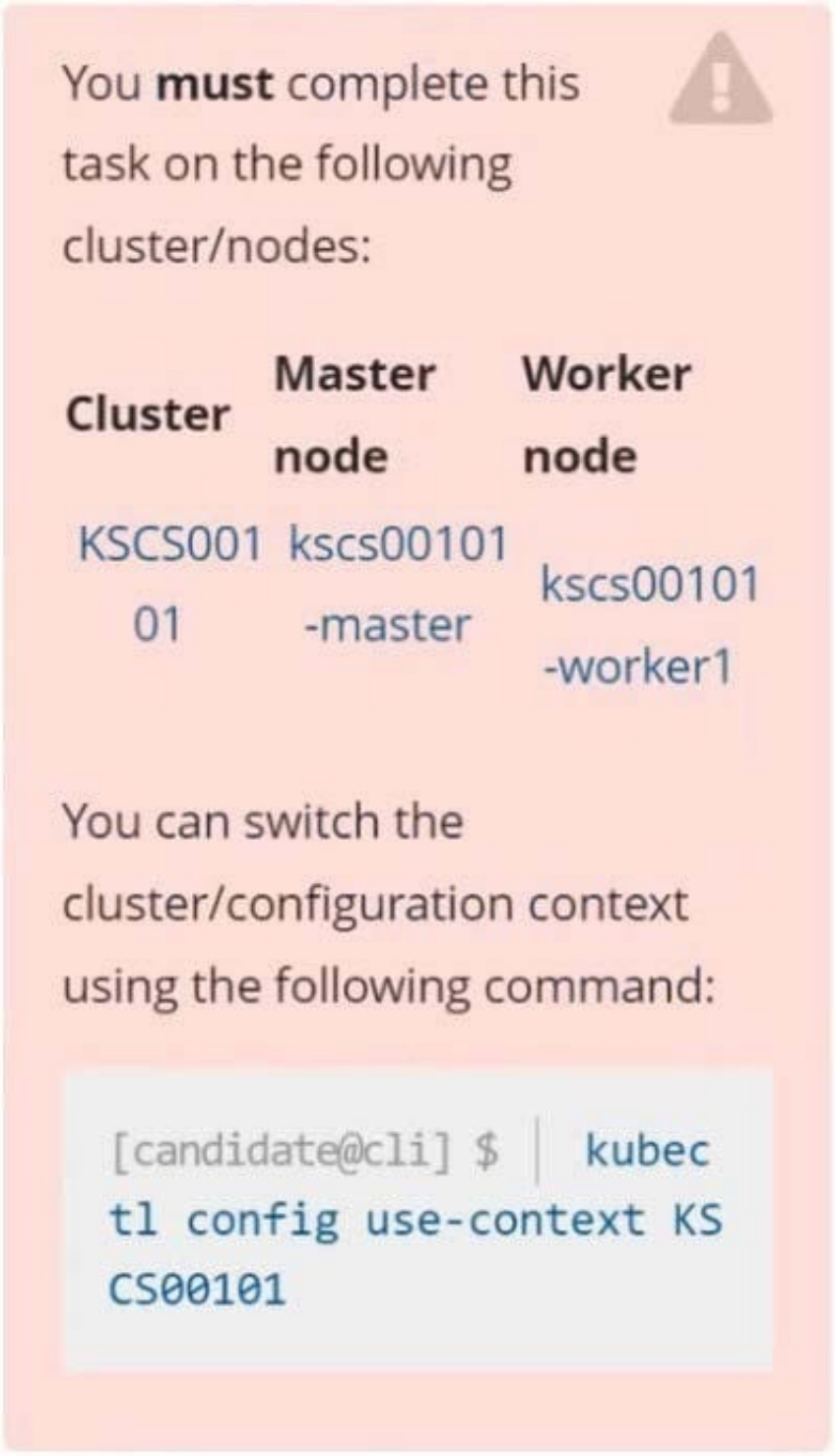
```
NAME                         READY   STATUS    RESTARTS   AGE
workload1-6869857dd7-s45rc   1/1     Running   0          5h45m
workload2-d4bd497d5-h44df    1/1     Running   0          5h45m
workload3-8587774495-chm56   1/1     Running   0          5h45m
candidate@cli:~$ kubectl -n server edit deployments.apps workload2
deployment.apps/workload2 edited
candidate@cli:~$ kubectl get pods -n server
NAME                         READY   STATUS    RESTARTS   AGE
workload1-8d8649ff6-wvjtg    1/1     Running   0          15s
workload2-765bdb98c8-wd8cm   1/1     Running   0          4s
workload3-8587774495-chm56   1/1     Running   0          5h45m
candidate@cli:~$ kubectl -n server edit deployments.apps workload3
```

```
    app: nginx
  name: workload3
spec:
  runtimeClassName: sandboxed
  containers:
  - image: nginx:1.14.2
    imagePullPolicy: IfNotPresent
    name: workload3
    ports:
```

```
candidate@cli:~$ kubectl -n server edit deployments.apps workload3
deployment.apps/workload3 edited
candidate@cli:~$ kubectl get pods -n server
NAME                         READY   STATUS    RESTARTS   AGE
workload1-8d8649ff6-wvjtg    1/1     Running   0          58s
workload2-765bdb98c8-wd8cm   1/1     Running   0          47s
workload3-76c994bb4d-s6k85   1/1     Running   0          4s
candidate@cli:~$
```

QUESTION 5

CORRECT TEXT Context



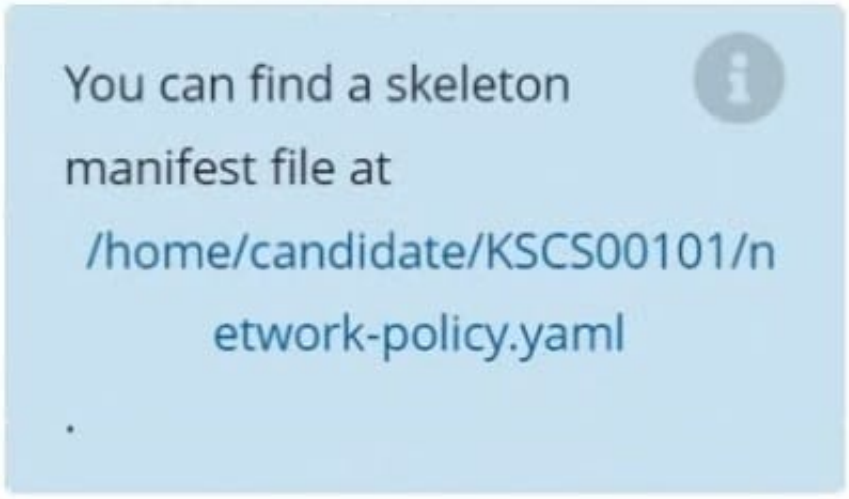
A default-deny NetworkPolicy avoids to accidentally expose a Pod in a namespace that doesn\\'t have any other NetworkPolicy defined.

Task

Create a new default-deny NetworkPolicy named defaultdeny in the namespace testing for all traffic of type Egress.

The new NetworkPolicy must deny all Egress traffic in the namespace testing.

Apply the newly created default-deny NetworkPolicy to all Pods running in namespace testing.



A. See explanation below.

B. PlaceHolder

Correct Answer: A