

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.pass2lead.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

CORRECT TEXT Context

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

Cluster	Master node	Worker node
KSCS001 01	kscs00101 -master	kscs00101 -worker1

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

```
[candidate@cli] $ | kubectl config use-context KSCS00101
```

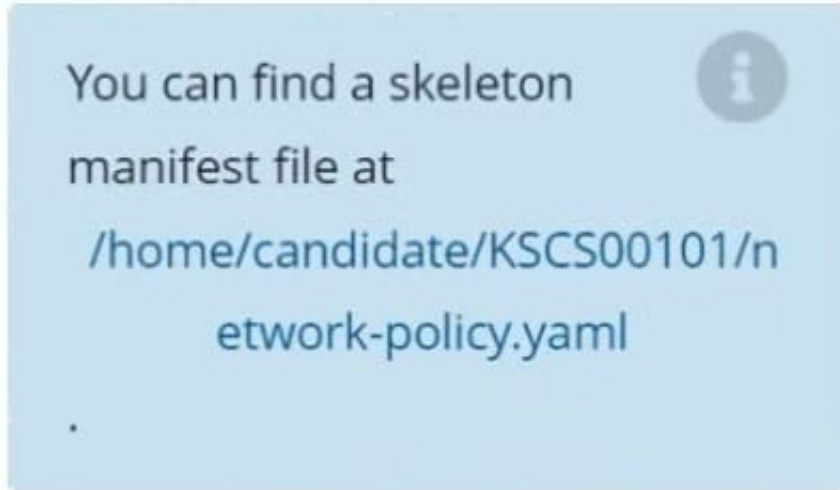
A default-deny NetworkPolicy avoids to accidentally expose a Pod in a namespace that doesn't have any other NetworkPolicy defined.

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

QUESTION 2

```
candidate@cli:~$ kubectl config use-context KSCS00101
Switched to context "KSCS00101".
candidate@cli:~$ cat /home/candidate/KSCS00101/network-policy.yaml
---
apiVersion: networking.k8s.io/v1
kind: NetworkPolicy
metadata:
  name: ""
  namespace: ""
spec:
  podSelector: {}
  policyTypes: []
candidate@cli:~$ vim /home/candidate/KSCS00101/network-policy.yaml
candidate@cli:~$ █
```

```
apiVersion: networking.k8s.io/v1
kind: NetworkPolicy
metadata:
  name: "defaultdeny"
  namespace: "testing"
spec:
  podSelector: {}
  policyTypes:
  - Egress
  egress:
  - to:
    - podSelector: {}
      namespaceSelector:
        matchLabels:
          access: testingproject
```

```
candidate@cli:~$ vim /home/candidate/KSCS00101/network-policy.yaml
candidate@cli:~$ vim /home/candidate/KSCS00101/network-policy.yaml
candidate@cli:~$ kubectl label ns testing access=testingproject
namespace/testing labeled
candidate@cli:~$ cat /home/candidate/KSCS00101/network-policy.yaml
---
apiVersion: networking.k8s.io/v1
kind: NetworkPolicy
metadata:
  name: "defaultdeny"
  namespace: "testing"
spec:
  podSelector: {}
  policyTypes:
  - Egress
  egress:
  - to:
    - podSelector: {}
      namespaceSelector:
        matchLabels:
          access: testingproject
candidate@cli:~$ kubectl create -f /home/candidate/KSCS00101/network-policy.yaml
networkpolicy.networking.k8s.io/defaultdeny created
candidate@cli:~$ kubectl -n testing describe networkpolicy
Name:          defaultdeny
Namespace:     testing
Created on:    2022-05-20 14:28:27 +0000 UTC
Labels:        <none>
Annotations:   <none>
Spec:
  PodSelector: <none> (Allowing the specific traffic to all pods in this namespace)
  Not affecting ingress traffic
  Allowing egress traffic:
    To Port: <any> (traffic allowed to all ports)
    To:
      NamespaceSelector: access=testingproject
      PodSelector: <none>
  Policy Types: Egress
candidate@cli:~$ █
```

Create a RuntimeClass named gvisor-rc using the prepared runtime handler named runsc.

Create a Pods of image Nginx in the Namespace server to run on the gVisor runtime class

A. See the explanation below:

B. Placeholder

Correct Answer: A

Install the Runtime Class for gVisor

{ # Step 1: Install a RuntimeClass cat