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.leads4pass.com/cks.html

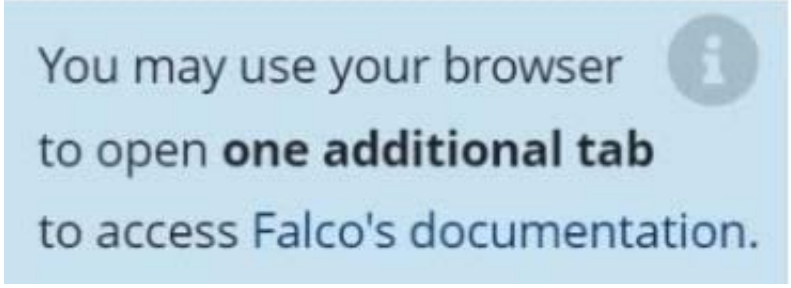
100% Passing Guarantee
100% Money Back Assurance

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

QUESTION 1

CORRECT TEXT


You must complete this task on the following cluster/nodes:
Cluster
Master node
Worker node
KSRS00101
ksrs00101-master
ksrs00101-worker1
You can switch the cluster/configuration context using the following command:
[candidate@cli] $ kubectl config use-context KSRS00101


You may use your browser to open one additional tab to access Falco's documentation.

Two tools are pre-installed on the cluster\\'s worker node:

1.

sysdig

2.

falco

Using the tool of your choice (including any non pre-installed tool), analyze the container\\'s behavior for at least 30 seconds, using filters that detect newly spawning and executing processes. Store an incident file at /opt/KSRS00101/alerts/

details, containing the detected incidents, one per line, in the following format:

```
timestamp,uid/username,proce
ssName
```

The following example shows a properly formatted incident file:

```
01:40:19.601363716,root,init
01:40:20.606013716,nobody,ba
sh
01:40:21.137163716,1000,tar
```

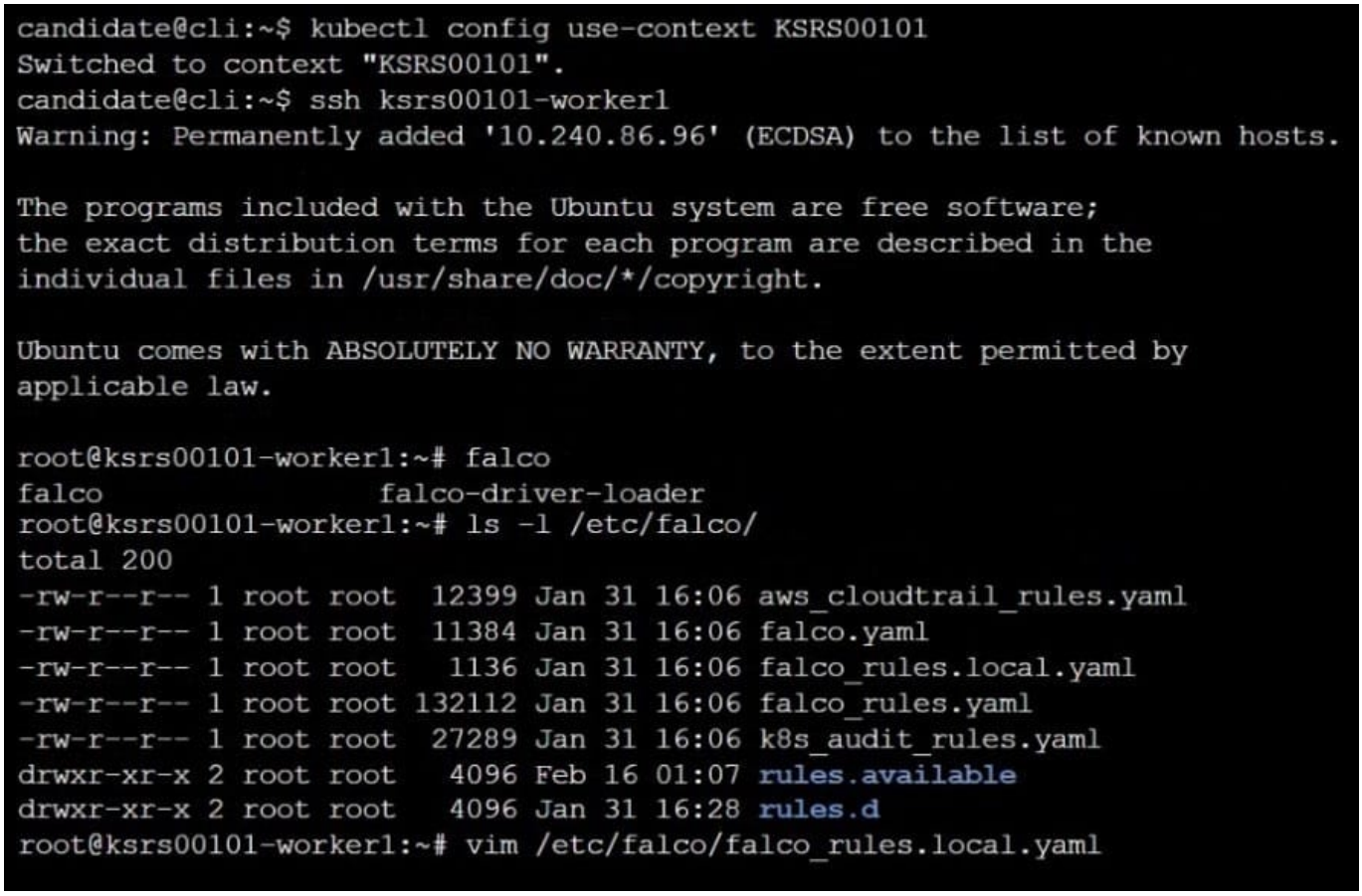
Keep the tool's original timestamp-format as-is.

Make sure to store the incident file on the cluster's worker node.

A. See the explanation below:

B. PlaceHolder

Correct Answer: A



```
# Copyright (C) 2019 The Falco Authors.
#
#
# Licensed under the Apache License, Version 2.0 (the "License");
# you may not use this file except in compliance with the License.
# You may obtain a copy of the License at
#
#     http://www.apache.org/licenses/LICENSE-2.0
#
# Unless required by applicable law or agreed to in writing, software
# distributed under the License is distributed on an "AS IS" BASIS,
# WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
# See the License for the specific language governing permissions and
# limitations under the License.
#

####################
# Your custom rules!
####################

# Add new rules, like this one
# - rule: The program "sudo" is run in a container
#   desc: An event will trigger every time you run sudo in a container
#   condition: evt.type = execve and evt.dir=< and container.id != host and proc.name = sudo
#   output: "Sudo run in container (user=%user.name %container.info parent=%proc.pname cmdli
ne=%proc.cmdline)"
#   priority: ERROR
#   tags: [users, container]

# Or override/append to any rule, macro, or list from the Default Rules
- rule: Container Drift Detected (chmod)
  desc: New executable created in a container due to chmod
  condition: >
    evt.type in (open,openat,create) and
    evt.is_open_exec=true and
    container and
    not runc_writing_exec_fifo and
    not runc_writing_var_lib_docker and
    not user_known_container_drift_activities and
    evt.rawres>=0
  output:
     %evt.time,%user.uid,%proc.name
  priority: ERROR
```

Text

```
root@ksrs00101-worker1:~# vim /etc/falco/falco_rules.local.yaml
root@ksrs00101-worker1:~# systemctl status falco.service
● falco.service - Falco Runtime Security
     Loaded: loaded (/lib/systemd/system/falco.service; disabled; vendor preset: enabled)
     Active: inactive (dead)
root@ksrs00101-worker1:~# systemctl enable falco.service
Created symlink /etc/systemd/system/multi-user.target.wants/falco.service → /lib/systemd/sys
tem/falco.service.
root@ksrs00101-worker1:~# systemctl start falco.service
root@ksrs00101-worker1:~# exit
logout
Connection to 10.240.86.96 closed.
candidate@cli:~$ ssh ksrs00101-worker1
Last login: Fri May 20 15:59:48 2022 from 10.240.86.88
root@ksrs00101-worker1:~# vim /etc/falco/falco.yaml
```

```
# When using json output, whether or not to include the "tags" property
# itself in the json output. If set to true, outputs caused by rules
# with no tags will have a "tags" field set to an empty array. If set to
# false, the "tags" field will not be included in the json output at all.
json_include_tags_property: true

# Send information logs to stderr and/or syslog Note these are *not* security
# notification logs! These are just Falco lifecycle (and possibly error) logs.
log_stderr: true
log_syslog: true
log_file: /opt/KSRS00101/alerts/details

# Minimum log level to include in logs. Note: these levels are
# separate from the priority field of rules. This refers only to the
# log level of falco's internal logging. Can be one of "emergency",
# "alert", "critical", "error", "warning", "notice", "info", "debug".
log_level: info
```

Text

```
root@ksrs00101-worker1:~# vim /etc/falco/falco.yaml
root@ksrs00101-worker1:~# grep log /etc/falco/falco.yaml
# cloudtrail log files.
# If true, the times displayed in log messages and output messages
# Send information logs to stderr and/or syslog Note these are *not* security
# notification logs! These are just Falco lifecycle (and possibly error) logs.
log_stderr: true
log_syslog: true
log_file: /opt/KSRS00101/alerts/details
# Minimum log level to include in logs. Note: these levels are
# log level of falco's internal logging. Can be one of "emergency",
log_level: info
#    - log: log a DEBUG message noting that the buffer was full
# Notice it is not possible to ignore and log/alert messages at the same time.
# The rate at which log/alert messages are emitted is governed by a
    - log
# The timeout error will be reported to the log according to the above log_* settings.
syslog_output:
#    - logging (alternate method than syslog):
#         program: logger -t falco-test
# this information will be logged, however the main Falco daemon will not be stopped.
root@ksrs00101-worker1:~# systemctl restart falco.service
root@ksrs00101-worker1:~# exit
logout
Connection to 10.240.86.96 closed.
candidate@cli:~$
```

QUESTION 2

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

[desk@cli] $ kubectl config use-context qa

Context:

A pod fails to run because of an incorrectly specified ServiceAccount

Task:

Create a new service account named backend-qa in an existing namespace qa, which must not have access to any secret.

Edit the frontend pod yaml to use backend-qa service account

Note: You can find the frontend pod yaml at /home/cert_masters/frontend-pod.yaml

A. See the explanation below

B. PlaceHolder

Correct Answer: A

[desk@cli] $ k create sa backend-qa -n qasa/backend-qa created [desk@cli] $ k get role,rolebinding -n qaNo resources found in qa namespace. [desk@cli] $ k create role backend -n qa --resource pods,namespaces,configmaps --verb list#

No access to secret [desk@cli] $ k create rolebinding backend -n qa --role backend --serviceaccount qa:backend-qa [desk@cli] $ vim /home/ cert_masters/frontend-pod.yaml uk.co.certification.simulator.questionpool.PList@120e0660 [desk@cli] $ k apply -f /home/cert_masters/frontend-pod.yamlpod created [desk@cli] $ k create sa backend-qa -n qaserviceaccount/backend-qa created [desk@cli] $ k get role,rolebinding -n qaNo resources found in qa namespace. [desk@cli] $ k create role backend -n qa --resource pods,namespaces,configmaps --verb listrole.rbac.authorization.k8s.io/backend created [desk@cli] $ k create rolebinding backend -n qa --role backend --serviceaccount qa:backendqarolebinding.rbac.authorization.k8s.io/backend created [desk@cli] $ vim /home/cert_masters/frontend-pod.yaml apiVersion: v1 kind: Pod metadata: name: frontend spec: serviceAccountName: backend-qa # Add this image: nginx name: frontend [desk@cli] $ k apply -f /home/cert_masters/frontend-pod.yamlpod/frontend createdhttps://kubernetes.io/docs/tasks/configure-pod-container/configure-service-account/

QUESTION 3

Cluster: qa-cluster

Master node: master Worker node: worker1 You can switch the cluster/configuration context using the following command: [desk@cli] $ kubectl config use-context qa-cluster

Task:

Create a NetworkPolicy named restricted-policy to restrict access to Pod product running in namespace dev.

Only allow the following Pods to connect to Pod products-service:

1.

Pods in the namespace qa

2.

Pods with label environment: stage, in any namespace

A. See the below.

B. PlaceHolder

Correct Answer: A

QUESTION 4

```
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