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.geekcert.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

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

Cluster: trace Master node: master Worker node: worker1

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

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

Given: You may use Sysdig or Falco documentation.

Task:

Use detection tools to detect anomalies like processes spawning and executing something weird frequently in the single container belonging to Pod tomcat.

Two tools are available to use:

1.

falco

2.

sysdig

Tools are pre-installed on the worker1 node only.

Analyse the container\\'s behaviour for at least 40 seconds, using filters that detect newly spawning and executing processes.

Store an incident file at /home/cert_masters/report, in the following format:

[timestamp],[uid],[processName]

Note: Make sure to store incident file on the cluster\\'s worker node, don\\'t move it to master node.

A. See the explanation below

B. PlaceHolder

Correct Answer: A

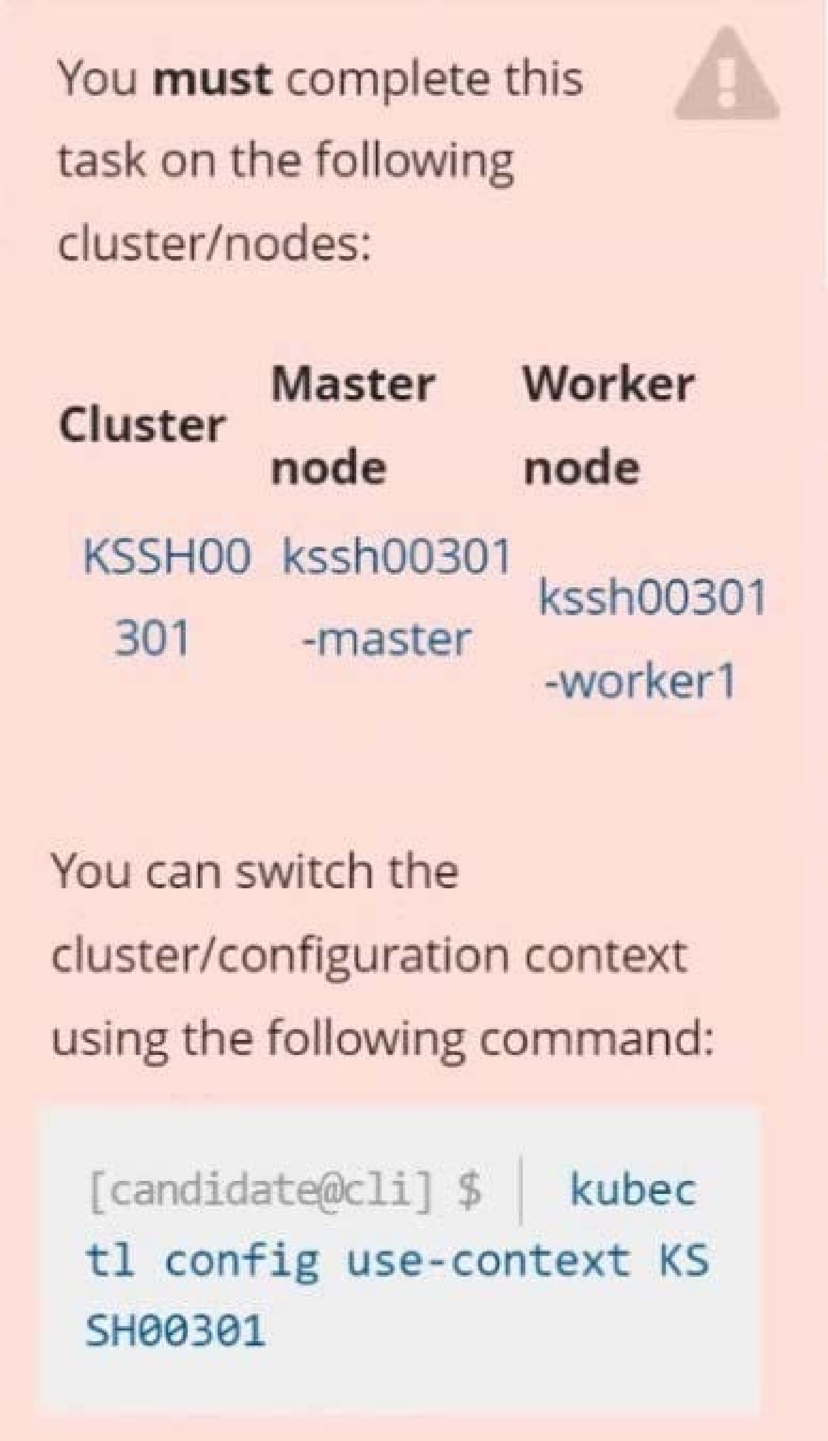
$vim /etc/falco/falco_rules.local.yaml uk.co.certification.simulator.questionpool.PList@120e24d0 $kill -1 Explanation[desk@cli] $ ssh node01[node01@cli] $ vim /etc/falco/falco_rules.yamlsearch for Container Drift Detected and paste in falco_rules.local.yaml[node01@cli] $ vim /etc/falco/falco_rules.local.yaml

-rule: Container Drift Detected (open+create) desc: New executable created in a container due to open+create condition: > evt.type in (open,openat,creat) 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 # Add this/Refer falco documentation priority: ERROR [node01@cli] $ vim /etc/falco/falco.yaml

QUESTION 2

CORRECT TEXT

Task



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

Cluster	Master node	Worker node
KSSH00301	kssh00301-master	kssh00301-worker1

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

```
[candidate@cli] $ kubectl config use-context KSSH00301
```

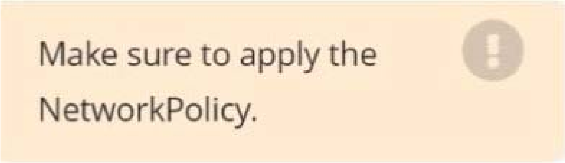
Create a NetworkPolicy named pod-access to restrict access to Pod users-service running in namespace dev-team. Only allow the following Pods to connect to Pod users-service:

1.

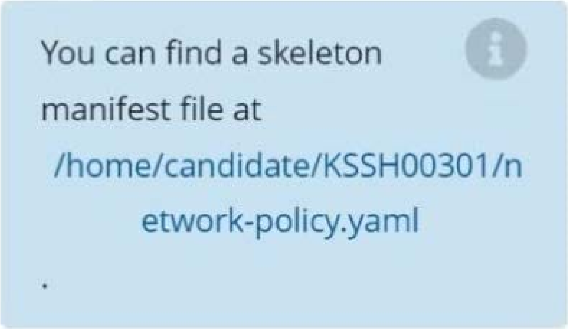
Pods in the namespace qa

2.

Pods with label environment: testing, in any namespace



Make sure to apply the NetworkPolicy.



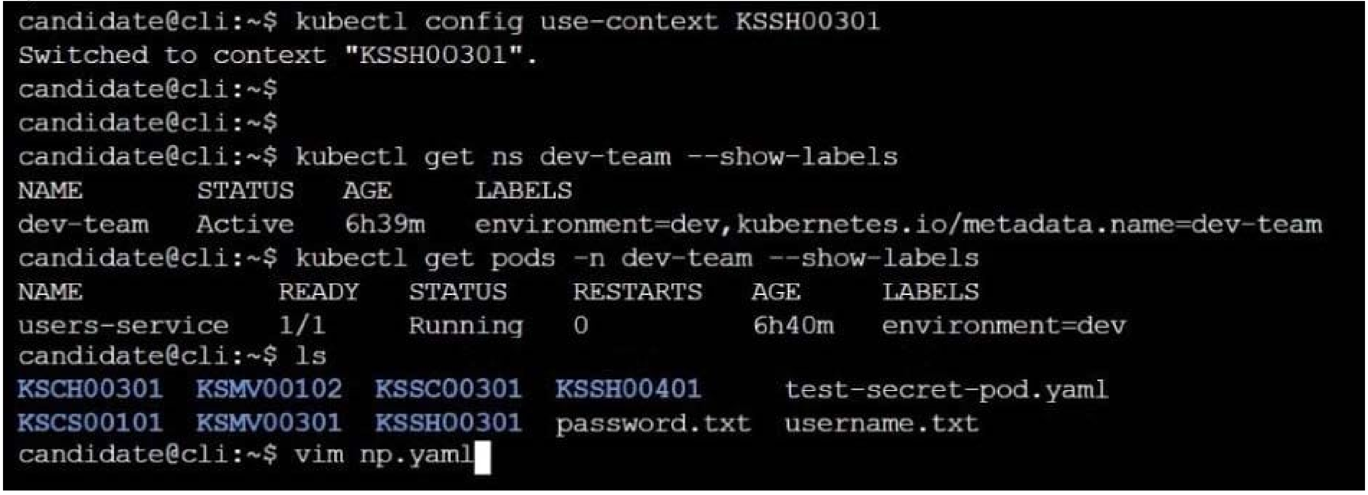
You can find a skeleton manifest file at /home/candidate/KSSH00301/network-policy.yaml

.

A. See explanation below.
B. PlaceHolder

Correct Answer: A
Explanation

Explanation/Reference:



```
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
```

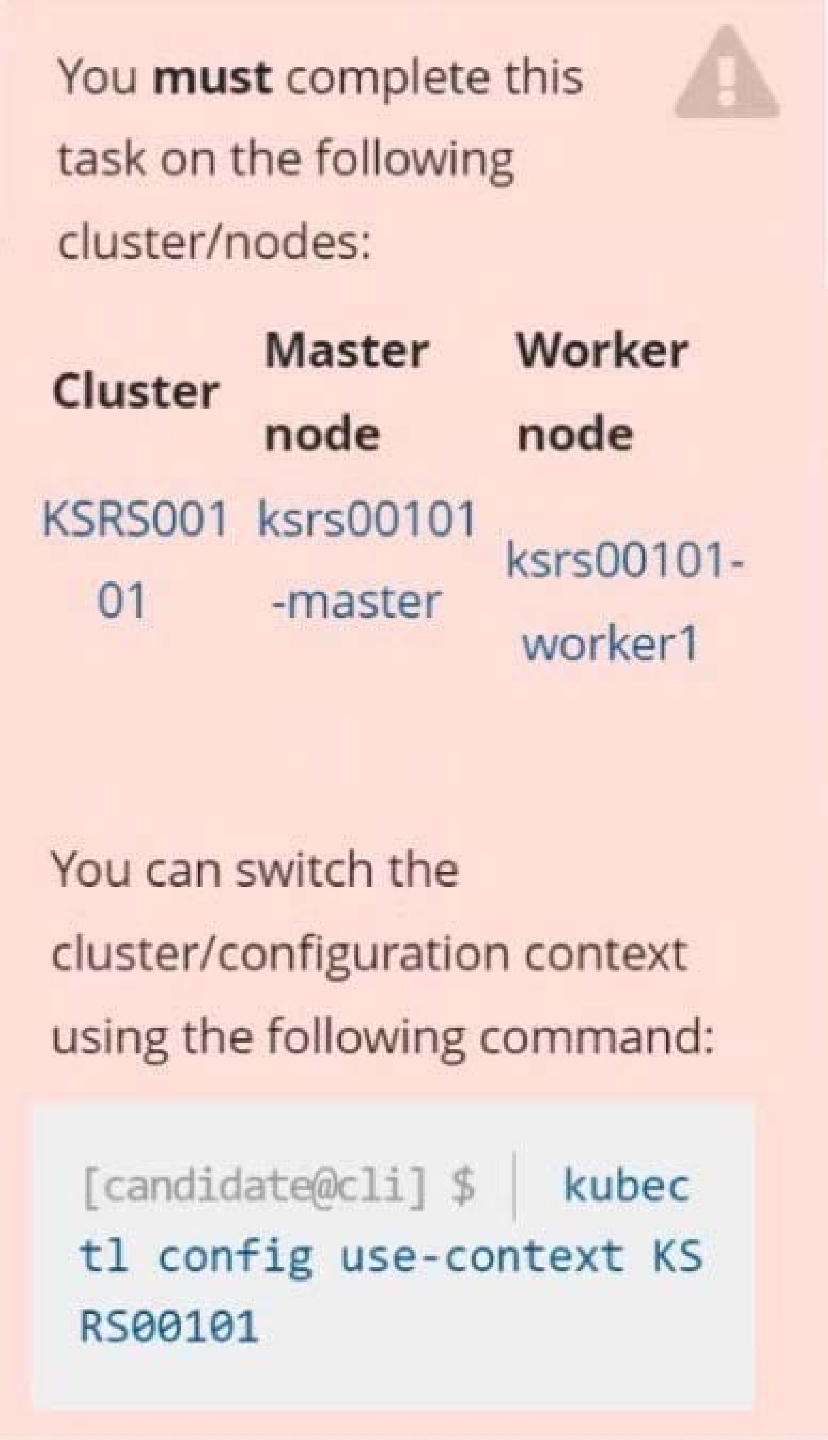
A. See explanation below.

B. PlaceHolder

Correct Answer: A

QUESTION 3

CORRECT TEXT

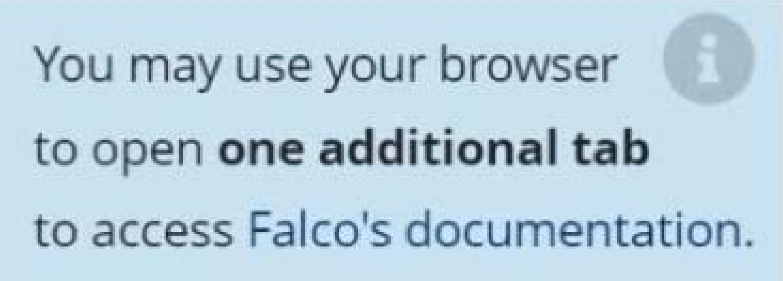


You **must** complete this task on the following cluster/nodes:

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.

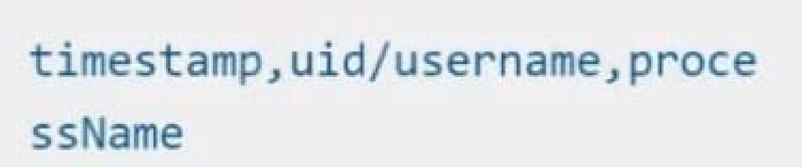
sysdig

2.

falco

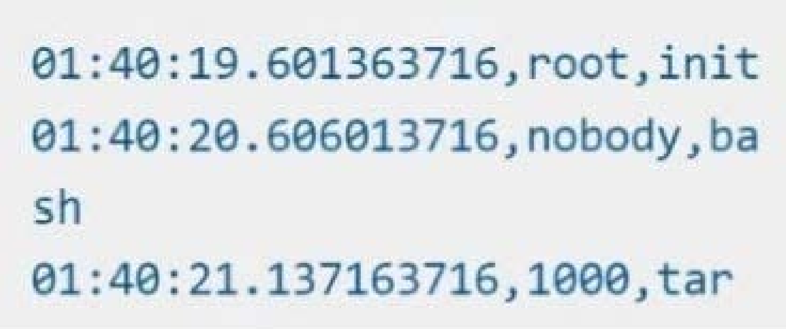
Using the tool of your choice (including any non pre-installed tool), analyze the container\\'s behavior for at least 30 seconds, using filters that detect newly spawning and executing processes. Store an incident file at /opt/KSRS00101/alerts/

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

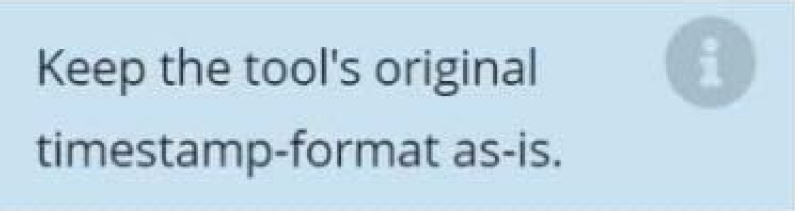


```
timestamp,uid/username,processName
```

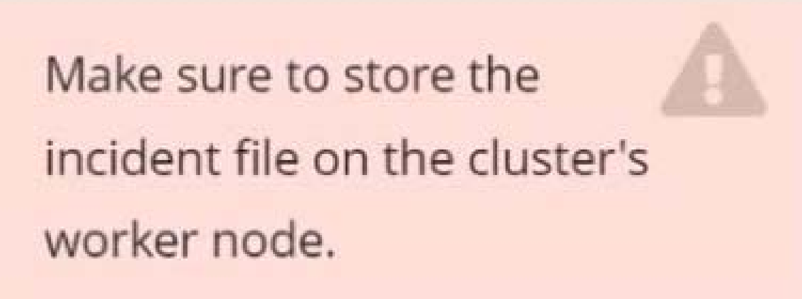
The following example shows a properly formatted incident file:



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



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

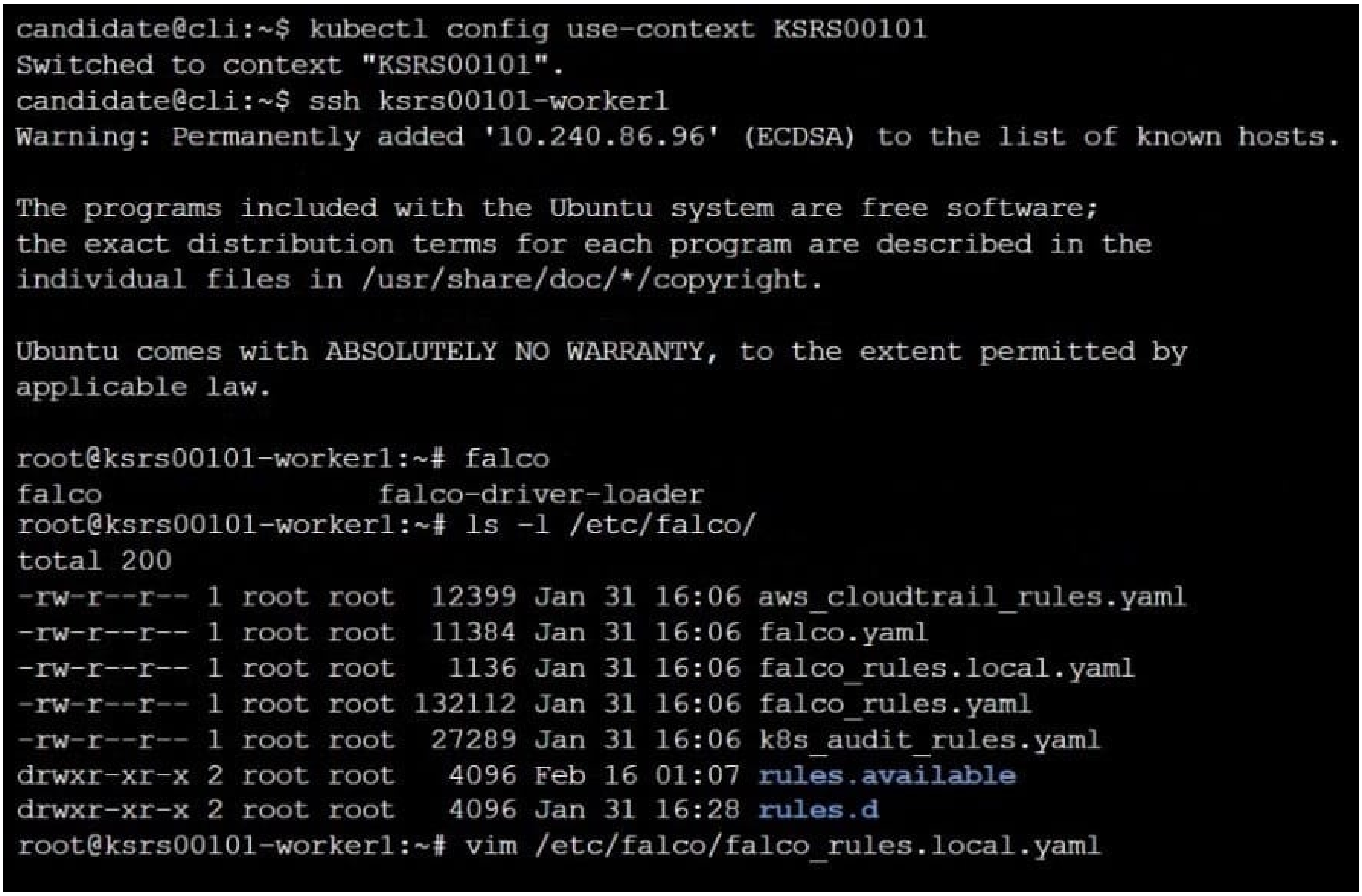


Make sure to store the incident file on the cluster's worker node.

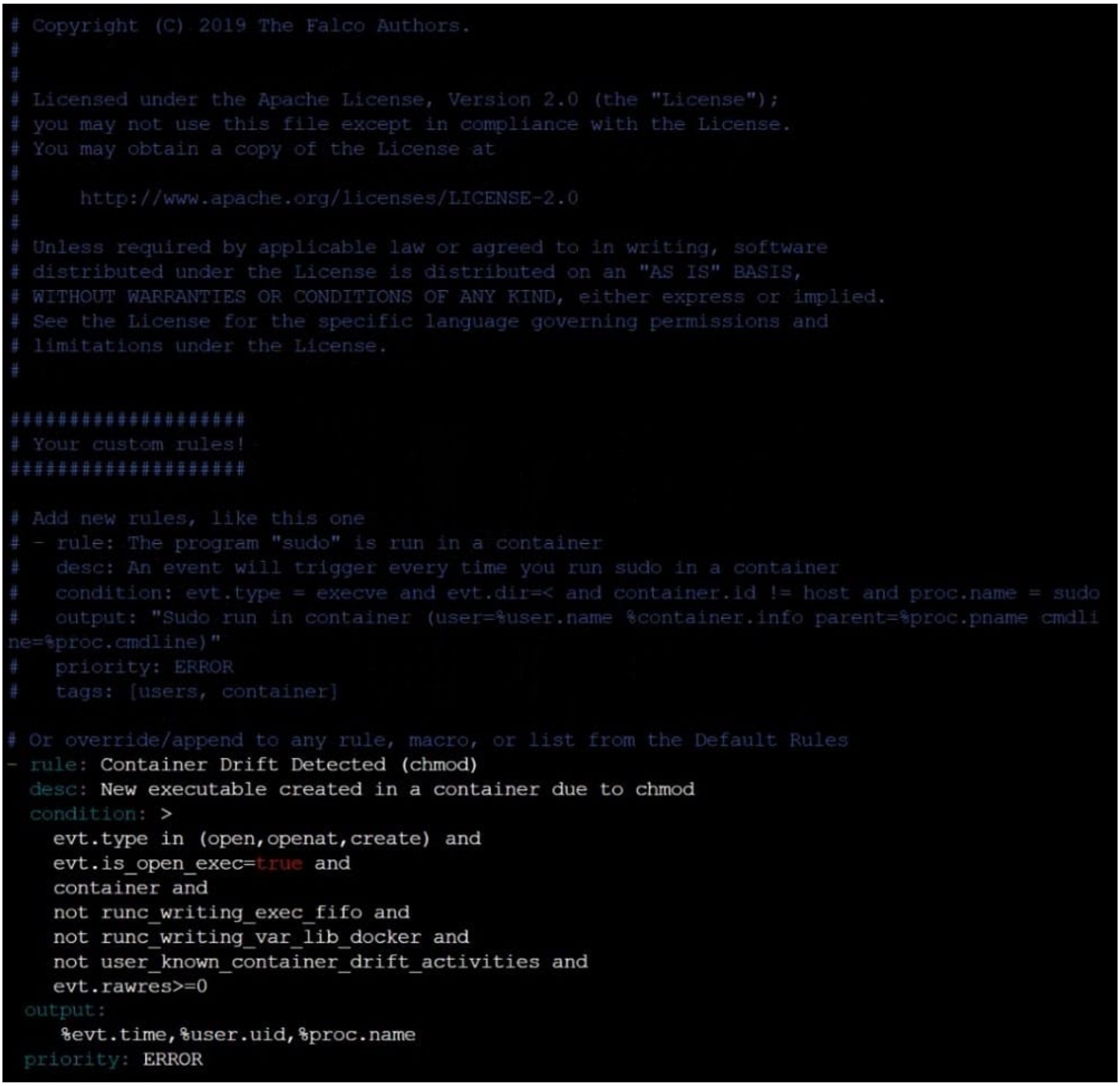
A. See the explanation below:

B. PlaceHolder

Correct Answer: A

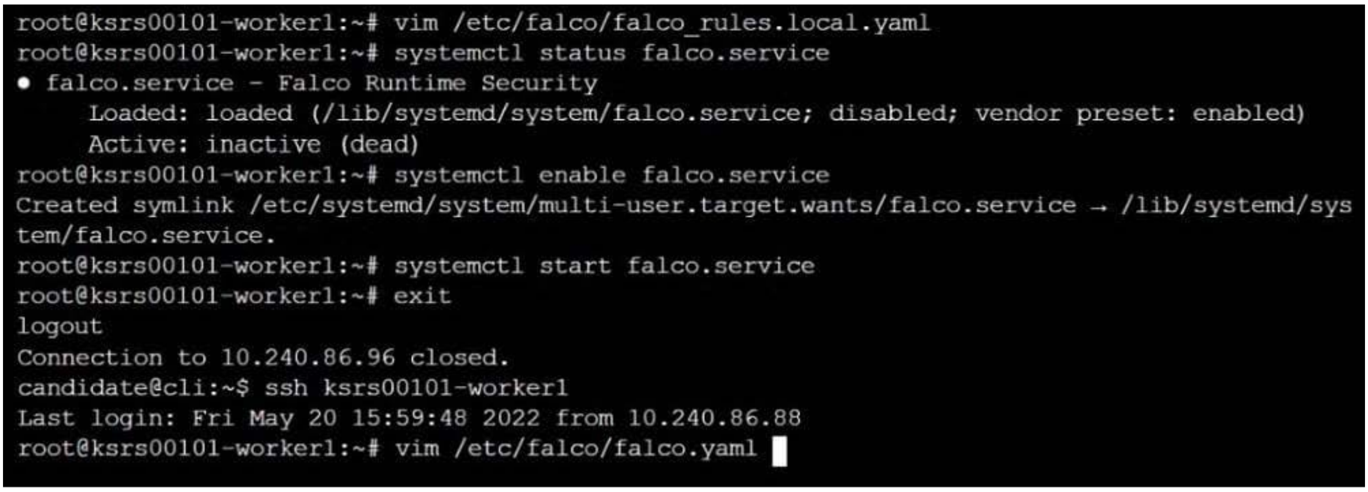


```
candidate@cli:~$ kubectl config use-context KSRS00101
Switched to context "KSRS00101".
candidate@cli:~$ ssh ksrs00101-worker1
Warning: Permanently added '10.240.86.96' (ECDSA) to the list of known hosts.

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@ksrs00101-worker1:~# falco
falco                    falco-driver-loader
root@ksrs00101-worker1:~# ls -l /etc/falco/
total 200
-rw-r--r-- 1 root root  12399 Jan 31 16:06 aws_cloudtrail_rules.yaml
-rw-r--r-- 1 root root  11384 Jan 31 16:06 falco.yaml
-rw-r--r-- 1 root root   1136 Jan 31 16:06 falco_rules.local.yaml
-rw-r--r-- 1 root root 132112 Jan 31 16:06 falco_rules.yaml
-rw-r--r-- 1 root root  27289 Jan 31 16:06 k8s_audit_rules.yaml
drwxr-xr-x 2 root root   4096 Feb 16 01:07 rules.available
drwxr-xr-x 2 root root   4096 Jan 31 16:28 rules.d
root@ksrs00101-worker1:~# vim /etc/falco/falco_rules.local.yaml
```

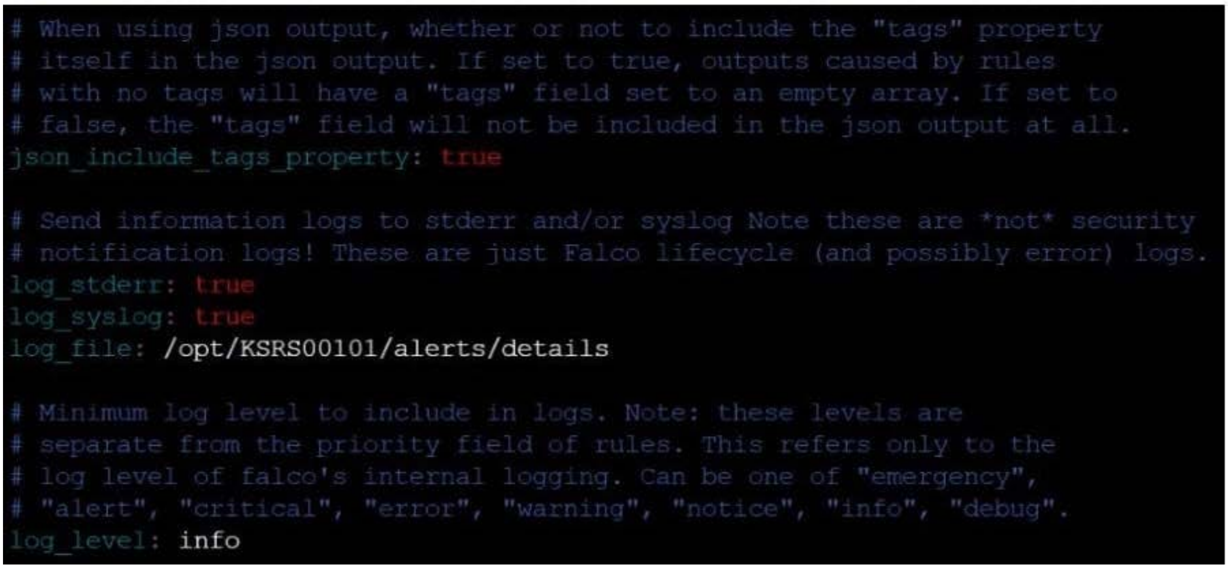


```
# Copyright (C) 2019 The Falco Authors.
#
#
# Licensed under the Apache License, Version 2.0 (the "License");
# you may not use this file except in compliance with the License.
# You may obtain a copy of the License at
#
#     http://www.apache.org/licenses/LICENSE-2.0
#
# Unless required by applicable 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

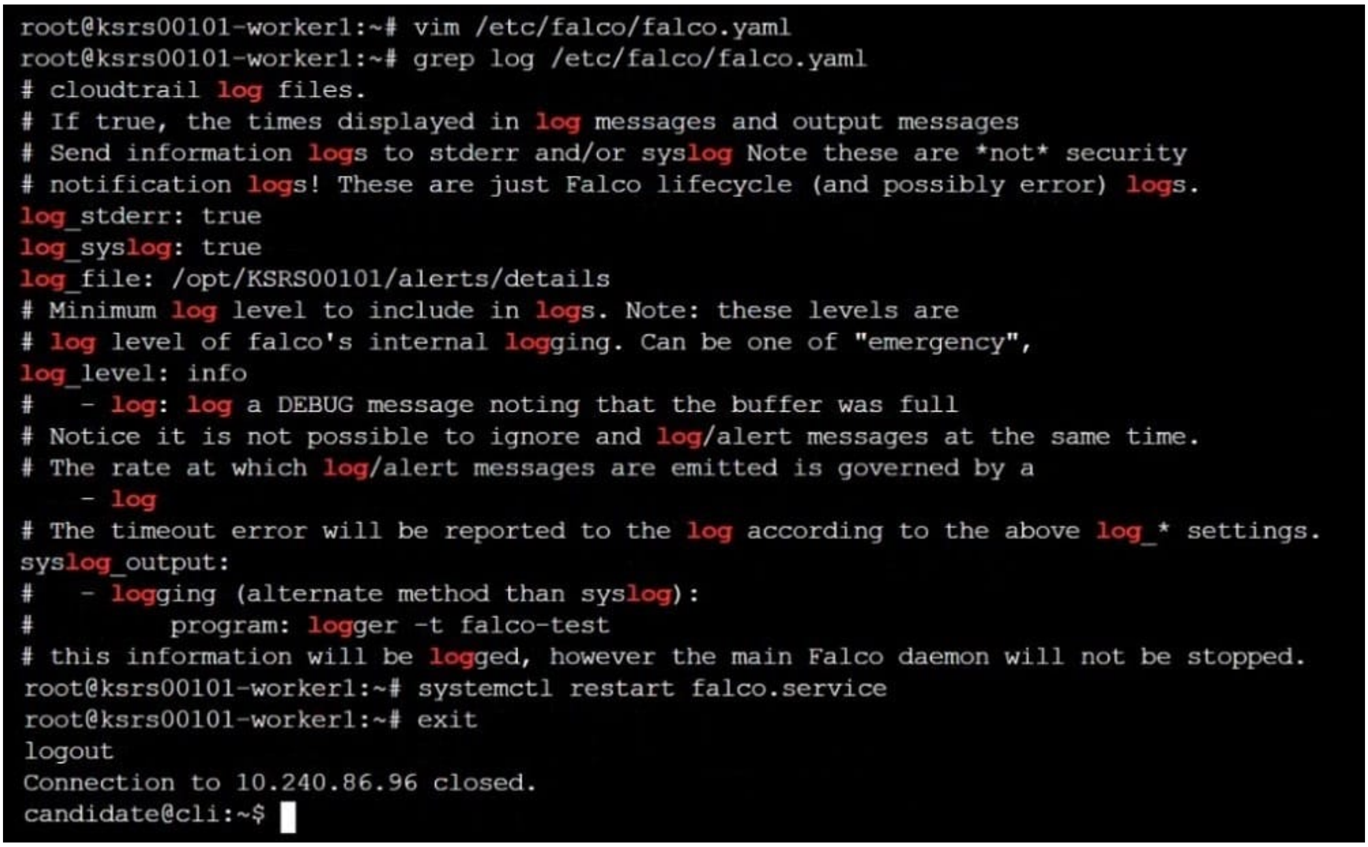


```
root@ksrs00101-worker1:~# vim /etc/falco/falco_rules.local.yaml
root@ksrs00101-worker1:~# systemctl status falco.service
● falco.service - Falco Runtime Security
     Loaded: loaded (/lib/systemd/system/falco.service; disabled; vendor preset: enabled)
     Active: inactive (dead)
root@ksrs00101-worker1:~# 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 4

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 5

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

Latest CKS Dumps CKS Practice Test CKS Study Guide