Free Questions for CKS by dumpshq

Shared by Shaw on 24-05-2024

For More Free Questions and Preparation Resources

Check the Links on Last Page

Question 1

Question Type: MultipleChoice

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

Cluster:apparmor

Master node:master

Worker node:worker1

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

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

Given: AppArmor is enabled on the worker1 node.

Task:

On the worker1 node,

1. Enforce the prepared AppArmor profile located at:/etc/apparmor.d/nginx

2. Edit the prepared manifest file located at/home/cert_masters/nginx.yamlto apply the apparmor profile

3. Create the Pod using this manifest

Options:

A) Explanation:

[desk@cli] $ssh worker1

[worker1@cli] $apparmor_parser -q /etc/apparmor.d/nginx

[worker1@cli] $aa-status | grep nginx

nginx-profile-1

[worker1@cli] $logout

[desk@cli] $vim nginx-deploy.yaml

Add these lines under metadata:

annotations: # Add this line

container.apparmor.security.beta.kubernetes.io/<container-name>: localhost/nginx-profile-1

[desk@cli] $kubectl apply -f nginx-deploy.yaml

Explanation

[desk@cli] $ssh worker1

[worker1@cli] $apparmor_parser -q /etc/apparmor.d/nginx

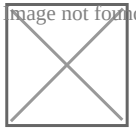
[worker1@cli] $aa-status | grep nginx

nginx-profile-1

[worker1@cli] $logout

[desk@cli] $vim nginx-deploy.yaml

Image not found or type unknown



[desk@cli] $kubectl apply -f nginx-deploy.yaml

pod/nginx-deploy created

Reference:https://kubernetes.io/docs/tutorials/clusters/apparmor/
pod/nginx-deploy created
[desk@cli] $kubectl apply -f nginx-deploy.yaml
pod/nginx-deploy created

Reference:https://kubernetes.io/docs/tutorials/clusters/apparmor/

Answer:

A

Question 2

Question Type: **MultipleChoice**

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

[desk@cli] $kubectl config use-context test-account

Task:Enable audit logs in the cluster.

To do so, enable the log backend, and ensure that:

1. logs are stored at/var/log/Kubernetes/logs.txt

2. log files are retained for5days

3. at maximum, a number of10old audit log files are retained

A basic policy is provided at/etc/Kubernetes/logpolicy/audit-policy.yaml. It only specifies what not to log.

Note: The base policy is located on the cluster's master node.

Edit and extend the basic policy to log:

1.Nodeschanges atRequestResponselevel

2. The request body ofpersistentvolumeschanges in the namespacefrontend

3.ConfigMapandSecretchanges in all namespaces at theMetadatalevel

Also, add a catch-all rule to log all other requests at theMetadatalevel

Note:Don't forget to apply the modified policy.

Options:

A) Explanation:
$vim /etc/kubernetes/log-policy/audit-policy.yaml
- level: RequestResponse
userGroups: ['system:nodes']
- level: Request
resources:

- group: '' # core API group
resources: ['persistentvolumes']
namespaces: ['frontend']
- level: Metadata
resources:
- group: ''
resources: ['configmaps', 'secrets']
- level: Metadata
$vim /etc/kubernetes/manifests/kube-apiserver.yaml
Add these
- --audit-policy-file=/etc/kubernetes/log-policy/audit-policy.yaml
- --audit-log-path=/var/log/kubernetes/logs.txt
- --audit-log-maxage=5
- --audit-log-maxbackup=10
Explanation
[desk@cli] $ssh master1
[master1@cli] $vim /etc/kubernetes/log-policy/audit-policy.yaml
apiVersion: audit.k8s.io/v1 # This is required.
kind: Policy
Don't generate audit events for all requests in RequestReceived stage.
omitStages:
- 'RequestReceived'
rules:
Don't log watch requests by the 'system:kube-proxy' on endpoints or services
- level: None
users: ['system:kube-proxy']

verbs: ['watch']
resources:
- group: '' # core API group
resources: ['endpoints', 'services']
Don't log authenticated requests to certain non-resource URL paths.
- level: None
userGroups: ['system:authenticated']
nonResourceURLs:
- '/api*' # Wildcard matching.
- '/version'
Add your changes below
- level: RequestResponse
userGroups: ['system:nodes'] # Block for nodes
- level: Request
resources:
- group: '' # core API group
resources: ['persistentvolumes'] # Block for persistentvolumes
namespaces: ['frontend'] # Block for persistentvolumes of frontend ns
- level: Metadata
resources:
- group: '' # core API group
resources: ['configmaps', 'secrets'] # Block for configmaps & secrets
- level: Metadata # Block for everything else
[master1@cli] $vim /etc/kubernetes/manifests/kube-apiserver.yaml
apiVersion: v1
kind: Pod

metadata:
annotations:
kubeadm.kubernetes.io/kube-apiserver.advertise-address.endpoint: 10.0.0.5:6443
labels:
component: kube-apiserver
tier: control-plane
name: kube-apiserver
namespace: kube-system
spec:
containers:
- command:
- kube-apiserver
- --advertise-address=10.0.0.5
- --allow-privileged=true
- --authorization-mode=Node,RBAC
- --audit-policy-file=/etc/kubernetes/log-policy/audit-policy.yaml #Add this
- --audit-log-path=/var/log/kubernetes/logs.txt #Add this
- --audit-log-maxage=5 #Add this
- --audit-log-maxbackup=10 #Add this
...
output truncated
Note: log volume & policy volume is already mounted invim /etc/kubernetes/manifests/kube-apiserver.yamlso no need to mount it.

Reference:https://kubernetes.io/docs/tasks/debug-application-cluster/audit/
Note: log volume & policy volume is already mounted invim /etc/kubernetes/manifests/kube-apiserver.yamlso no need to mount it.

Reference:https://kubernetes.io/docs/tasks/debug-application-cluster/audit/

Answer:

A

Question 3

Question Type: **MultipleChoice**

Context:

Cluster:prod

Master node:master1

Worker node:worker1

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

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

Task:

Analyse and edit the given Dockerfile (based on theubuntu:18:04image)

/home/cert_masters/Dockerfilefixing two instructions present in the file being prominent security/best-practice issues.

Analyse and edit the given manifest file

/home/cert_masters/mydeployment.yamlfixing two fields present in the file being prominent security/best-practice issues.

Note:Don't add or remove configuration settings; only modify the existing configuration settings, so that two configuration settings each are no longer security/best-practice concerns.

Should you need an unprivileged user for any of the tasks, use usernobodywith user id65535

Options:

A) Explanation:
1. For Dockerfile:Fix the image version & user name in Dockerfile
2. For mydeployment.yaml : Fix security contexts
Explanation
[desk@cli] $vim /home/cert_masters/Dockerfile
FROM ubuntu:latest # Remove this
FROM ubuntu:18.04 # Add this
USER root # Remove this
USER nobody # Add this
RUN apt get install -y lsof=4.72 wget=1.17.1 nginx=4.2
ENV ENVIRONMENT=testing
USER root # Remove this
USER nobody # Add this

CMD ['nginx -d']

Image not found or type unknown

[desk@cli] $vim/home/cert_masters/mydeployment.yaml

apiVersion: apps/v1

kind: Deployment

metadata:

creationTimestamp: null

labels:

app: kafka

name: kafka

spec:

replicas: 1

selector:

matchLabels:

app: kafka

strategy: {}

template:

metadata:

creationTimestamp: null

labels:

app: kafka

spec:

containers:

- image: bitnami/kafka
name: kafka
volumeMounts:
- name: kafka-vol
mountPath: /var/lib/kafka
securityContext:
{'capabilities':{'add':['NET_ADMIN'],'drop':['all']},'privileged': True,'readOnlyRootFilesystem': False, 'runAsUser': 65535} # Delete This
{'capabilities':{'add':['NET_ADMIN'],'drop':['all']},'privileged': False,'readOnlyRootFilesystem': True, 'runAsUser': 65535} # Add This
resources: {}
volumes:
- name: kafka-vol
emptyDir: {}
status: {}
Pictorial View:
[desk@cli] $vim/home/cert_masters/mydeployment.yaml

Answer:

A

Question 4

Question Type: MultipleChoice

Given an existing Pod named test-web-pod running in the namespace test-system

Edit the existing Role bound to the Pod's Service Account named sa-backend to only allow performing get operations on endpoints.

Create a new Role named test-system-role-2 in the namespace test-system, which can perform patch operations, on resources of type statefulsets.

Options:

A) Create a new RoleBinding named test-system-role-2-binding binding the newly created Role to the Pod's ServiceAccount sa-backend.

Answer:

A

Question 5

Question Type: MultipleChoice

Create a network policy named restrict-np to restrict to pod nginx-test running in namespace testing.

Only allow the following Pods to connect to Pod nginx-test:-

1. pods in the namespace default

2. pods with label version:v1 in any namespace.

Make sure to apply the network policy.

Options:

A) Explanation:

Answer:

A

Question 6

Question Type: **MultipleChoice**

Create a User named john, create the CSR Request, fetch the certificate of the user after approving it.

Create a Role name john-role to list secrets, pods in namespace john

Finally, Create a RoleBinding named john-role-binding to attach the newly created role john-role to the user john in the namespace john.

To Verify:Use the kubectl auth CLI command to verify the permissions.

Options:

A) Explanation:
se kubectl to create a CSR and approve it.
Get the list of CSRs:
kubectl get csr
Approve the CSR:
kubectl certificate approve myuser
Get the certificate
Retrieve the certificate from the CSR:
kubectl get csr/myuser -o yaml
here are the role and role-binding to give john permission to create NEW_CRD resource:
kubectl apply -f roleBindingJohn.yaml --as=john
rolebinding.rbac.authorization.k8s.io/john_external-rosource-rb created
kind: RoleBinding
apiVersion: rbac.authorization.k8s.io/v1
metadata:
name: john_crd
namespace: development-john

subjects:
- kind: User
name: john
apiGroup: rbac.authorization.k8s.io
roleRef:
kind: ClusterRole
name: crd-creation
kind: ClusterRole
apiVersion: rbac.authorization.k8s.io/v1
metadata:
name: crd-creation
rules:
- apiGroups: ['kubernetes-client.io/v1']
resources: ['NEW_CRD']
verbs: ['create, list, get']

Answer:

A

Question 7

Question Type: **MultipleChoice**

Using the runtime detection tool Falco, Analyse the container behavior for at least 30 seconds, using filters that detect newly spawning and executing processes

Options:

A) store the incident file art /opt/falco-incident.txt, containing the detected incidents. one per line, in the format [timestamp],[uid],[user-name],[processName]

Answer:

A

Question 8

Question Type: **MultipleChoice**

use the Trivy to scan the following images,

Options:

A) 1. amazonlinux:1

2. k8s.gcr.io/kube-controller-manager:v1.18.6

Look for images with HIGH or CRITICAL severity vulnerabilities and store the output of the same in /opt/trivy-vulnerable.txt

Answer:

A

Question 9

Question Type: MultipleChoice

Enable audit logs in the cluster, To Do so, enable the log backend, and ensure that

1. logs are stored at /var/log/kubernetes/kubernetes-logs.txt.

2. Log files are retained for 5 days.

3. at maximum, a number of 10 old audit logs files are retained.

Edit and extend the basic policy to log:

Options:

A) 1. Cronjobs changes at RequestResponse

2. Log the request body of deployments changes in the namespace kube-system.
3. Log all other resources in core and extensions at the Request level.
4. Don't log watch requests by the 'system:kube-proxy' on endpoints or

Answer:

A