Free Questions for KCNA by go4braindumps

Shared by Weber on 09-08-2024

For More Free Questions and Preparation Resources

Check the Links on Last Page

Question 1

Question Type: MultipleChoice

Which of the following best describes the way kubernetes Role-based access control (RBAC) works?

Options:

A- Kubernetes does not do RBAC

B- Kubernetes RBAC states which users can perform which actions against which re-source

C- Kubernetes RBAC lists which operations on which resources are denied to users

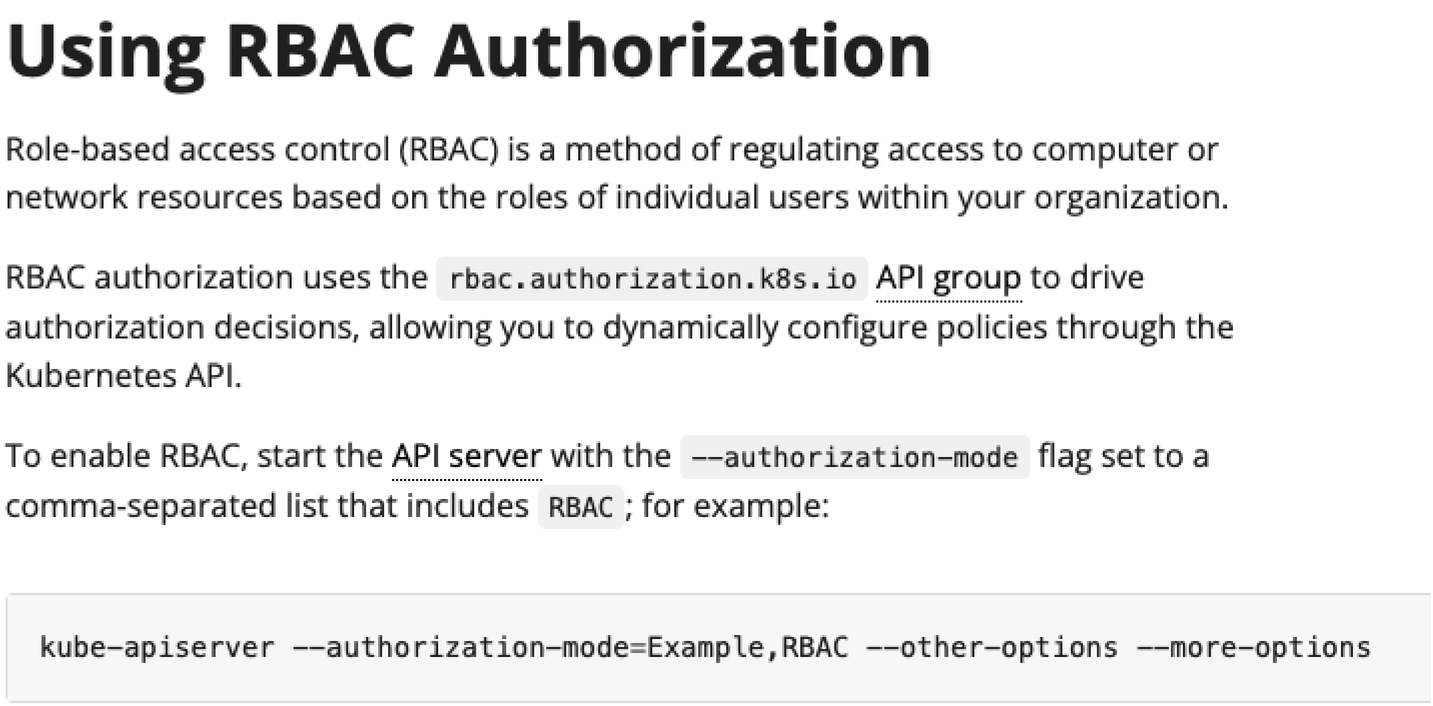
D- Kubernetes RBAC is responsible for authenticating subjects such as users and groups

Answer:

B

Explanation:

https://kubernetes.io/docs/reference/access-authn-authz/rbac/



Using RBAC Authorization

Role-based access control (RBAC) is a method of regulating access to computer or network resources based on the roles of individual users within your organization.

RBAC authorization uses the `rbac.authorization.k8s.io` API group to drive authorization decisions, allowing you to dynamically configure policies through the Kubernetes API.

To enable RBAC, start the API server with the `--authorization-mode` flag set to a comma-separated list that includes `RBAC`; for example:

```
kube-apiserver --authorization-mode=Example,RBAC --other-options --more-options
```

Question 2

Question Type: **MultipleChoice**

What is the name of the Kubernetes agent that runs on each worker nodes?

Options:

A- kubelet

B- systemd

C- kube-proxy

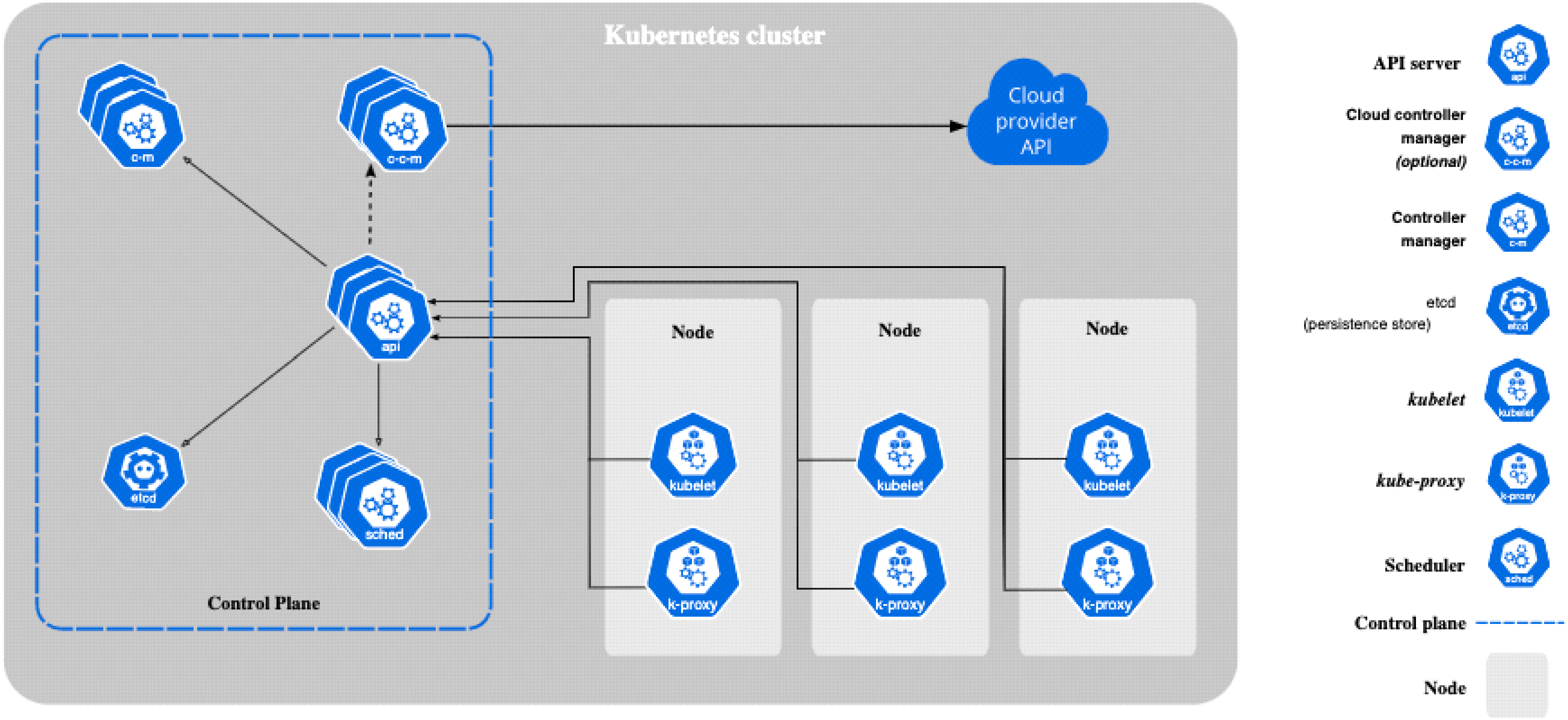
D- pod

Answer:

A

Explanation:

https://kubernetes.io/docs/concepts/overview/components/



Question 3

Question Type: MultipleChoice

The Kubernetes API provides an interface for storing objects. Which of the following describes the type of objects stored by the Kubernetes API?

Options:

A- Containers

B- REST

C- YAML

D- ETCD

Answer:

B

Explanation:

Kubernetes objects are RESTful objects.

Question 4

Question Type: **MultipleChoice**

Which of the following best describes the way K8S Role-based access control (RBAC) works?

Options:

A- K8S does not do RBAC or Cluster role

B- RBAC lists which operations are denied to users

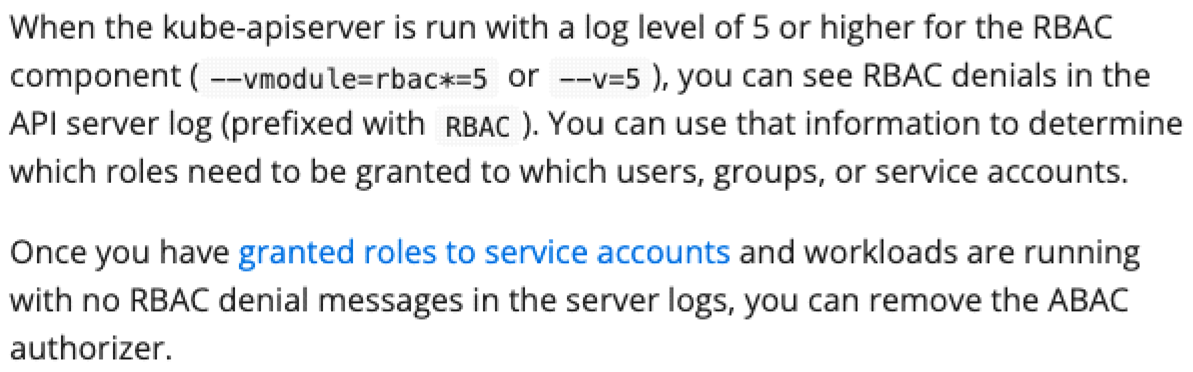
C- States which users can perform which actions against the resources.

Answer:

C

Explanation:

https://kubernetes.io/docs/reference/access-authn-authz/rbac/



When the kube-apiserver is run with a log level of 5 or higher for the RBAC component (`--vmodule=rbac*=5` or `--v=5`), you can see RBAC denials in the API server log (prefixed with `RBAC`). You can use that information to determine which roles need to be granted to which users, groups, or service accounts.

Once you have granted roles to service accounts and workloads are running with no RBAC denial messages in the server logs, you can remove the ABAC authorizer.

Question 5

Question Type: **MultipleChoice**

To run a startup task before a Pod's container starts up. What Kubernetes feature can help you ac-complish this?

Options:

A- Init container

B- Sidecar container

C- Startup probe

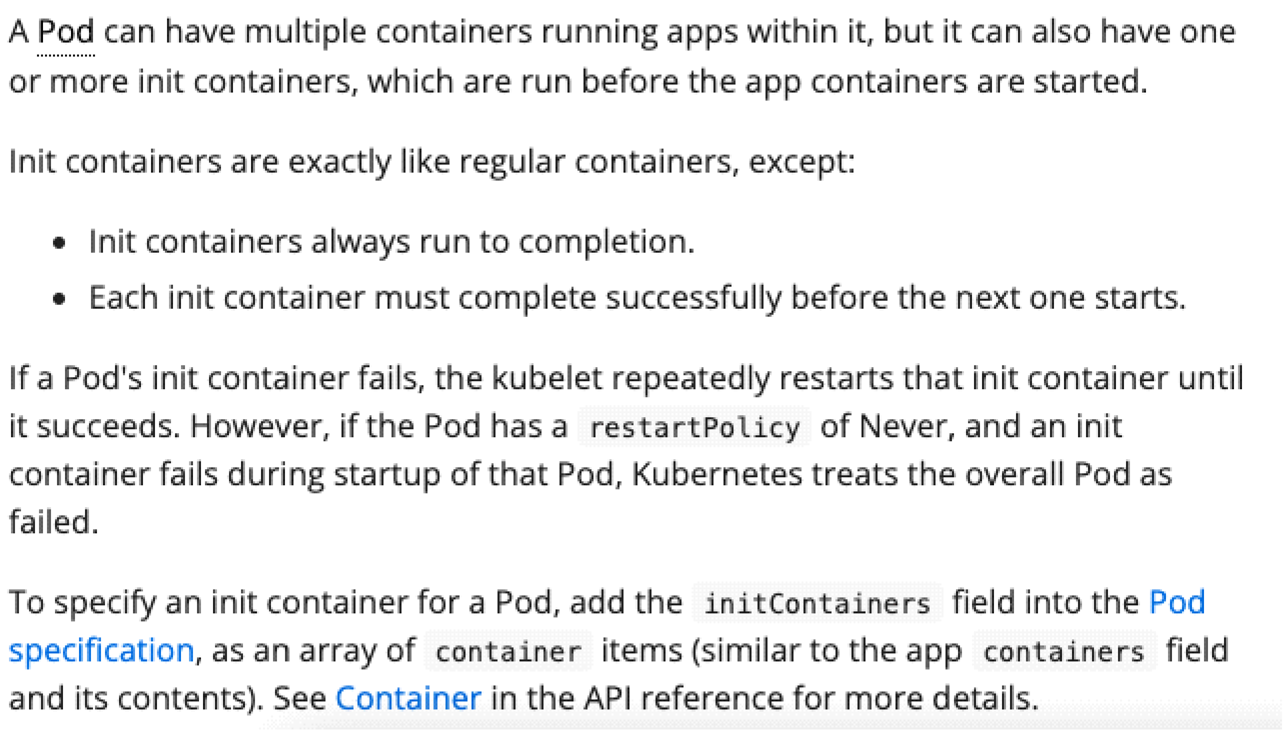
D- DaemonSet

Answer:

A

Explanation:

https://kubernetes.io/docs/concepts/workloads/pods/init-containers/



A Pod can have multiple containers running apps within it, but it can also have one or more init containers, which are run before the app containers are started.

Init containers are exactly like regular containers, except:

- Init containers always run to completion.
- Each init container must complete successfully before the next one starts.

If a Pod's init container fails, the kubelet repeatedly restarts that init container until it succeeds. However, if the Pod has a `restartPolicy` of Never, and an init container fails during startup of that Pod, Kubernetes treats the overall Pod as failed.

To specify an init container for a Pod, add the `initContainers` field into the Pod specification, as an array of `container` items (similar to the app `containers` field and its contents). See Container in the API reference for more details.

Question 6

Question Type: MultipleChoice

What is a benefits of Kubernetes federation?

Options:

A- Avoids scalability limits on pods and nodes

B- Creates highly available clusters in different regions

C- Low latency

Answer:

A, B, C

Question 7

Question Type: **MultipleChoice**

Which control plane component is responsible for scheduling pods?

Options:

A- kube-proxy

B- kube scheduler

C- kubelet

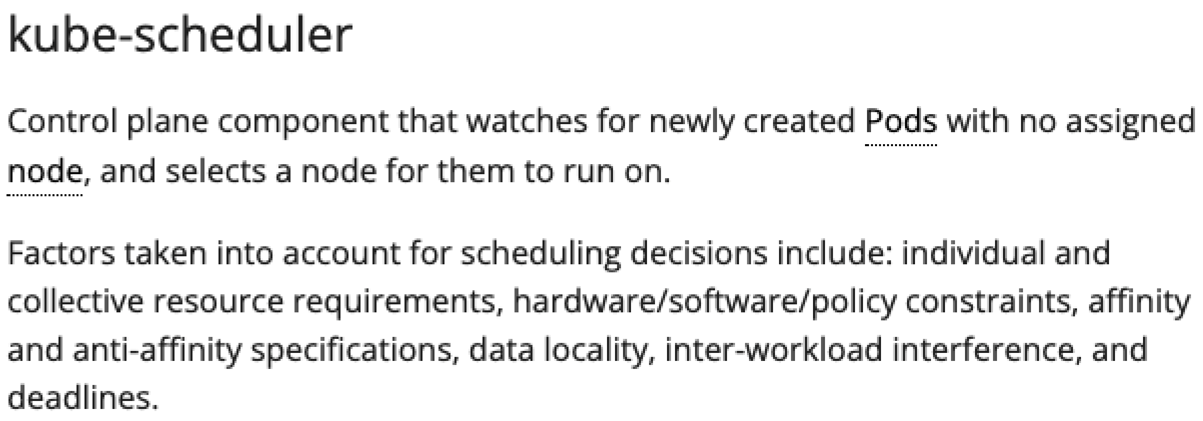
D- kube api-server

Answer:

B

Explanation:

https://kubernetes.io/docs/concepts/overview/components/



kube-scheduler

Control plane component that watches for newly created Pods with no assigned node, and selects a node for them to run on.

Factors taken into account for scheduling decisions include: individual and collective resource requirements, hardware/software/policy constraints, affinity and anti-affinity specifications, data locality, inter-workload interference, and deadlines.

Question 8

Question Type: **MultipleChoice**

Which part of a Kubernetes cluster is responsible for running container workloads?

Options:

A- Worker Node

B- kube-proxy

C- Control plane

E- etcd

Answer:

A

Explanation:

Worker Nodes are responsible for executing containerized workloads.

Question 9

Question Type: **MultipleChoice**

Which of the following is NOT a Kubernetes component?

Options:

A- Scheduler

B- Docker

C- Cloud Controller manager

D- Kube-proxy

Answer:

B

Explanation:

Docker is not a Kubernetes component.

Question 10

Question Type: MultipleChoice

Which style of operations are preferred for K8S and cloud native applications?

Options:

A- JSON

B- Declarative

C- Imperative

Answer:

B

Explanation:

https://kubernetes.io/docs/tasks/manage-kubernetes-objects/declarative-config/#trade-offs

Question 11

Question Type: **MultipleChoice**

Which of the following container runtime is planned to be deprecated in Kubernetes 1.20 and high-er?

Options:

A- cri-o

B- None of the options

C- docker

D- podman

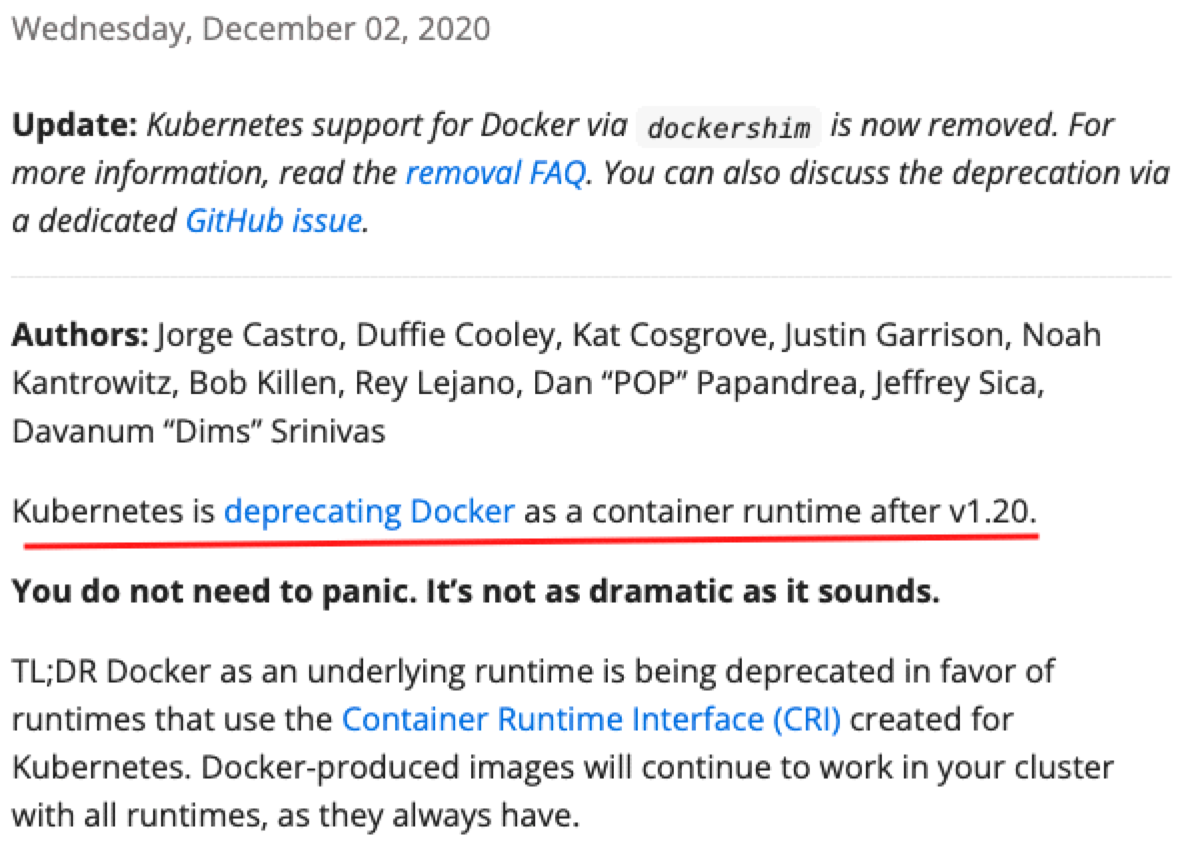
E- containerd

Answer:

C

Explanation:

https://kubernetes.io/blog/2020/12/02/dont-panic-kubernetes-and-docker/



Wednesday, December 02, 2020

Update: *Kubernetes support for Docker via* `dockershim` *is now removed. For more information, read the removal FAQ. You can also discuss the deprecation via a dedicated GitHub issue.*

Authors: Jorge Castro, Duffie Cooley, Kat Cosgrove, Justin Garrison, Noah Kantrowitz, Bob Killen, Rey Lejano, Dan "POP" Papandrea, Jeffrey Sica, Davanum "Dims" Srinivas

Kubernetes is deprecating Docker as a container runtime after v1.20.

You do not need to panic. It's not as dramatic as it sounds.

TL;DR Docker as an underlying runtime is being deprecated in favor of runtimes that use the Container Runtime Interface (CRI) created for Kubernetes. Docker-produced images will continue to work in your cluster with all runtimes, as they always have.

Question 12

Question Type: MultipleChoice

How to create deployment name app-dep, image=nginx, and replicas 5 using imperative command?

Options:

A- kubectl create app-dep deployment --image=nginx --replicas=5

B- kubectl create deployment app-dep --image=nginx --replicas=5

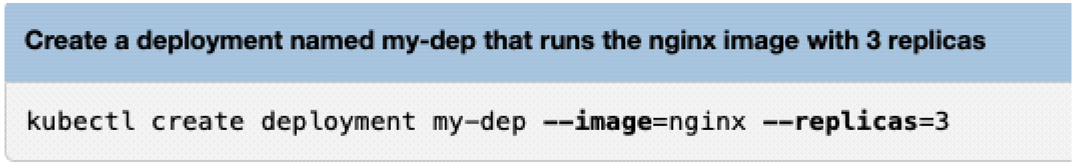
C- kubectl create app-dep deployment --replicas=5 --image=nginx

Answer:

B

Explanation:

https://kubernetes.io/docs/reference/generated/kubectl/kubectl-commands#-em-deployment-em-



Create a deployment named my-dep that runs the nginx image with 3 replicas

```
kubectl create deployment my-dep --image=nginx --replicas=3
```

To Get Premium Files for KCNA Visit

https://www.p2pexams.com/products/kcna

For More Free Questions Visit

https://www.p2pexams.com/linux-foundation/pdf/kcna