



Free Questions for **KCNA** by **braindumpscollection**

Shared by **Britt** on **29-01-2024**

For More Free Questions and Preparation Resources

Check the Links on Last Page

Question 1

Question Type: MultipleChoice

Notary and the update framework leading security projects in CNCF

Options:

A- TRUE

B- FALSE

Answer:

A

Explanation:

<https://github.com/cncf/landscape#trail-map>



CLOUD NATIVE TRAIL MAP

The Cloud Native Landscape [/cncf.io](https://cncf.io) has a large number of options. This Cloud Native Trail Map is a recommended process for leveraging open source, cloud native technologies. At each step, you can choose a vendor-supported offering or do it yourself, and everything after step #3 is optional based on your circumstances.

HELP ALONG THE WAY

A. Training and Certification

Consider training offerings from CNCF and then take the exam to become a Certified Kubernetes Administrator or a Certified Kubernetes Application Developer [/cncf.io/training](https://cncf.io/training)

B. Consulting Help

If you want assistance with Kubernetes and the surrounding ecosystem, consider leveraging a Kubernetes Certified

1. CONTAINERIZATION

- Commonly done with Docker containers
- Any size application and dependencies (even PDP-11 code running on an emulator) can be containerized
- Over time, you should aspire towards splitting suitable applications and writing future functionality as microservices



3. ORCHESTRATION & APPLICATION DEFINITION

- Kubernetes is the market-leading orchestration solution
- You should select a Certified Kubernetes Distribution, Hosted Platform, or Installer: cncf.io/ck
- Helm Charts help you define, install, and upgrade even the most complex Kubernetes application



5. SERVICE PROXY, DISCOVERY, & MESH

- CoreDNS is a fast and flexible tool that

2. CI/CD

- Setup CI/CD that can handle container events
- Setup a CI/CD pipeline
- Argo is a declarative workflow paradigm for progressing

4. MONITORING

- Prometheus
- Grafana

Question 2

Question Type: MultipleChoice

Which command-line tool is used to interact with the Kubernetes cluster?

Options:

A- kube-api

B- kubectl

C- kube-scheduler

Answer:

B

Explanation:

Command line tool (kubectl)

Kubernetes provides a command line tool for communicating with a Kubernetes cluster's control plane, using the Kubernetes API.

This tool is named `kubectl`.

For configuration, `kubectl` looks for a file named `config` in the `$HOME/.kube` directory. You can specify other `kubeconfig` files by setting the `KUBECONFIG` environment variable or by setting the `--kubeconfig` flag.

This overview covers `kubectl` syntax, describes the command operations, and provides common examples. For details about each command, including all the supported flags and subcommands, see the [kubectl](#) reference documentation.

For installation instructions, see [Installing kubectl](#); for a quick guide, see the [cheat sheet](#). If you're used to using the `docker` command-line tool, [kubectl for Docker Users](#) explains some equivalent commands for Kubernetes.

Question 3

Question Type: MultipleChoice

In Kubernetes, what is considered the primary cluster data source?

Options:

- A- etcd (pronounce: esty-d)
- B- api server
- C- kubelet
- D- scheduler

Answer:

A

Explanation:

<https://kubernetes.io/docs/concepts/overview/components/#etcd>

etcd

Consistent and highly-available key value store used as Kubernetes' backing store for all cluster data.

If your Kubernetes cluster uses etcd as its backing store, make sure you have a [back up](#) plan for those data.

You can find in-depth information about etcd in the official [documentation](#).

Question 4

Question Type: MultipleChoice

Observability and monitoring are not the same?

Options:

A- True

B- False

Answer:

A

Question 5

Question Type: MultipleChoice

What is the default service type in Kubernetes?

Options:

A- ClusterIP

B- NodePort

C- serviceType

D- loadBalancer

Answer:

A

Explanation:

<https://kubernetes.io/docs/concepts/services-networking/service/#publishing-services-service-types>

Kubernetes `ServiceTypes` allow you to specify what kind of Service you want. The default is `ClusterIP`.

`Type` values and their behaviors are:

- `ClusterIP`: Exposes the Service on a cluster-internal IP. Choosing this value makes the Service only reachable from within the cluster. This is the default `ServiceType`.
- `NodePort`: Exposes the Service on each Node's IP at a static port (the `NodePort`). A `ClusterIP` Service, to which the `NodePort` Service routes, is automatically created. You'll be able to contact the `NodePort` Service, from outside the cluster, by requesting `<NodeIP>:<NodePort>`.
- `LoadBalancer`: Exposes the Service externally using a cloud provider's load balancer. `NodePort` and `ClusterIP` Services, to which the external load balancer routes, are automatically created.
- `ExternalName`: Maps the Service to the contents of the `externalName` field (e.g. `foo.bar.example.com`), by returning a `CNAME` record with its value. No proxying of any kind is set up.

Question 6

Question Type: MultipleChoice

What are default kubernetes namespaces?

Options:

- A-** default, kube-public, kube-system, kube-node-lease
- B-** kube-default, kube-public, kube-system, kube-node-lease
- C-** default, kube-public, kube-systems, kube-node-lease
- D-** default, kube-public, kube-system, kube-node-leases

Answer:

A

Explanation:

<https://kubernetes.io/docs/concepts/overview/working-with-objects/namespaces/>

You can list the current namespaces in a cluster using:

```
kubectl get namespace
```

NAME	STATUS	AGE
default	Active	1d
kube-node-lease	Active	1d
kube-public	Active	1d
kube-system	Active	1d

Kubernetes starts with four initial namespaces:

- `default` The default namespace for objects with no other namespace
- `kube-system` The namespace for objects created by the Kubernetes system
- `kube-public` This namespace is created automatically and is readable by all users (including those not authenticated). This namespace is mostly reserved for cluster usage, in case that some resources should be visible and readable publicly throughout the whole cluster. The public aspect of this namespace is only a convention, not a requirement.
- `kube-node-lease` This namespace holds [Lease](#) objects associated with each node. Node leases allow the kubelet to send [heartbeats](#) so that the control plane can detect node failure.

Question 7

Question Type: MultipleChoice

Fluentd is the leading project in the CNCF space for logging?

Options:

A- TRUE

B- FALSE

Answer:

A

Explanation:

<https://github.com/cncf/landscape#trail-map>



CLOUD NATIVE TRAIL MAP

The Cloud Native Landscape [/cncf.io](https://cncf.io) has a large number of options. This Cloud Native Trail Map is a recommended process for leveraging open source, cloud native technologies. At each step, you can choose a vendor-supported offering or do it yourself, and everything after step #3 is optional based on your circumstances.

HELP ALONG THE WAY

A. Training and Certification

Consider training offerings from CNCF and then take the exam to become a Certified Kubernetes Administrator or a Certified Kubernetes Application Developer [/cncf.io/training](https://cncf.io/training)

B. Consulting Help

If you want assistance with Kubernetes and the surrounding ecosystem, consider leveraging a Kubernetes Certified

1. CONTAINERIZATION

- Commonly done with Docker containers
- Any size application and dependencies (even PDP-11 code running on an emulator) can be containerized
- Over time, you should aspire towards splitting suitable applications and writing future functionality as microservices



3. ORCHESTRATION & APPLICATION DEFINITION

- Kubernetes is the market-leading orchestration solution
- You should select a Certified Kubernetes Distribution, Hosted Platform, or Installer: cncf.io/ck
- Helm Charts help you define, install, and upgrade even the most complex Kubernetes application



5. SERVICE PROXY, DISCOVERY, & MESH

- CoreDNS is a fast and flexible tool that

2. CI/CD

- Setup CI/CD that can handle container events
- Setup a CI/CD pipeline
- Argo is a declarative workflow paradigm for progressing

4. MONITORING

- Prometheus is a monitoring system and time series database
- Grafana is a multi-platform dashboard for Prometheus

Question 8

Question Type: MultipleChoice

The Kubernetes rolling update is used for ___.

Options:

- A- Updating a service
- B- Scaling an application
- C- Updating a deployment

Answer:

C

Explanation:

<https://kubernetes.io/docs/tutorials/kubernetes-basics/update/update-intro/>

Performing a Rolling Update

Objectives

- Perform a rolling update using kubectl.

Updating an application

Users expect applications to be available all the time and developers are expected to deploy new versions of them several times a day. In Kubernetes this is done with rolling updates. **Rolling updates** allow Deployments' update to take place with zero downtime by incrementally updating Pods instances with new ones. The new Pods will be scheduled on Nodes with available resources.

In the previous module we scaled our application to run multiple instances. This is a requirement for performing updates without affecting application availability. By default, the maximum number of Pods that can be unavailable during the update and the maximum number of new Pods that can be created, is one. Both options can be configured to either numbers or percentages (of Pods). In Kubernetes, updates are versioned and any Deployment update can be reverted to a previous (stable) version.

Summary:

- Updating an app

Rolling updates allow Deployments' update to take place with zero downtime by incrementally updating Pods instances with new ones.

Question 9

Question Type: MultipleChoice

A _____ is an application running on kubernetes.

Options:

A- node

B- pod

C- workload

D- container

Answer:

C

Explanation:

<https://kubernetes.io/docs/concepts/workloads/>

Workloads

A workload is an application running on Kubernetes. Whether your workload is a single component or several that work together, on Kubernetes you run it inside a set of *Pods*. In Kubernetes, a Pod represents a set of running containers on your cluster.

Kubernetes pods have a *defined lifecycle*. For example, once a pod is running in your cluster then a critical fault on the node where that pod is running means that all the pods on that node fail. Kubernetes treats this level of failure as final: you would need to create a new Pod to recover, even if the node later becomes healthy.

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>

