

Linux-Foundation

Exam Questions KCNA

Kubernetes and Cloud Native Associate (KCNA)



NEW QUESTION 1

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

- 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-> Text Description automatically generated with medium confidence

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

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

NEW QUESTION 2

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

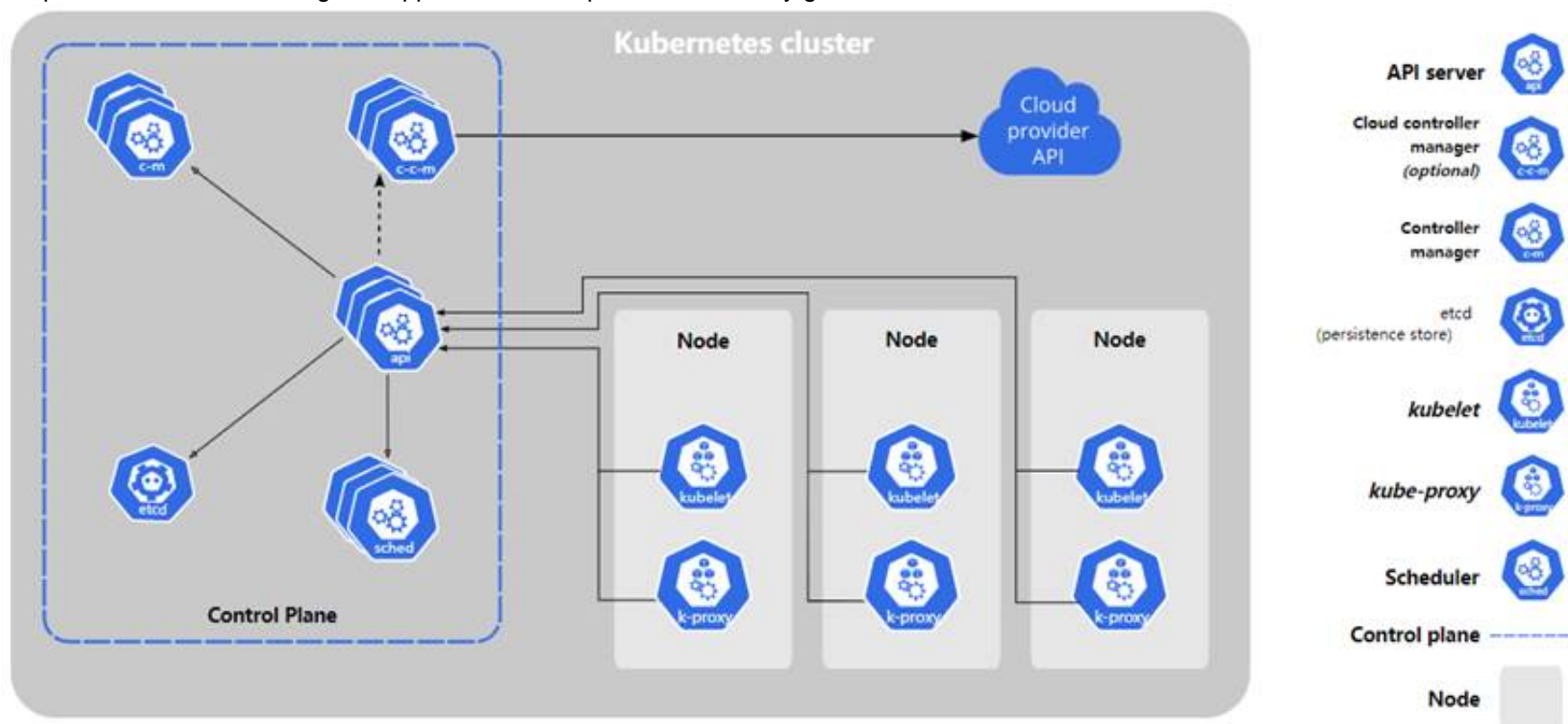
- A. kubelet
- B. systemd
- C. kube-proxy
- D. pod

Answer: A

Explanation:

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

Graphical user interface, diagram, application Description automatically generated



NEW QUESTION 3

What is the primary interface for Kubernetes cluster?

- A. Kubernetes Api
- B. Kubelet
- C. YAML
- D. Control Plane
- E. JSON

Answer: A

Explanation:

<https://kubernetes.io/docs/concepts/overview/components/#kube-apiserver>

A screenshot of a computer Description automatically generated with medium confidence

kube-apiserver

The API server is a component of the Kubernetes control plane that exposes the Kubernetes API. The API server is the front end for the Kubernetes control plane.

The main implementation of a Kubernetes API server is [kube-apiserver](#). kube-apiserver is designed to scale horizontally—that is, it scales by deploying more instances. You can run several instances of kube-apiserver and balance traffic between those instances.

NEW QUESTION 4

What can you use to add new resource types to your cluster?

- A. start container
- B. CustomResourceDefinitions
- C. init container
- D. Flux
- E. CRI-O

Answer: B

Explanation:

<https://kubernetes.io/docs/concepts/extend-kubernetes/api-extension/custom-resources/> Graphical user interface, text, application Description automatically generated

CustomResourceDefinitions

The [CustomResourceDefinition](#) API resource allows you to define custom resources. Defining a CRD object creates a new custom resource with a name and schema that you specify. The Kubernetes API serves and handles the storage of your custom resource. The name of a CRD object must be a valid [DNS subdomain name](#).

This frees you from writing your own API server to handle the custom resource, but the generic nature of the implementation means you have less flexibility than with [API server aggregation](#).

Refer to the [custom controller example](#) for an example of how to register a new custom resource, work with instances of your new resource type, and use a controller to handle events.

NEW QUESTION 5

The Kubernetes rolling update is used for _____ .

- 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/> Graphical user interface, text Description automatically generated with medium confidence

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.

NEW QUESTION 6

Which project in this list is a leading project in the observability space?

- A. Jaeger
- B. Vitess
- C. Argo
- D. Kubernetes

Answer: A

Explanation:

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



CLOUD NATIVE TRAIL MAP

The Cloud Native Landscape (cncl.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

B. Consulting Help

If you want assistance with Kubernetes and the surrounding ecosystem, consider leveraging a Kubernetes Certified Service Provider. cncf.io/csp

C. Join CNCF's End User Community

For companies that don't offer cloud native services externally. cncf.io/enduser

WHAT IS CLOUD NATIVE?

Cloud native technologies empower organizations to build and run scalable applications in modern, dynamic environments such as public, private, and hybrid clouds. Containers, service meshes, microservices, immutable infrastructure, and declarative APIs exemplify this approach.

These techniques enable loosely coupled systems that are resilient, manageable, and observable. Combined with robust automation, they allow engineers to make high-impact changes frequently and predictably with minimal toil.

The Cloud Native Computing Foundation seeks to drive adoption of this paradigm by fostering and sustaining an ecosystem of open source, vendor-neutral projects. We democratize state-of-the-art patterns to make these innovations accessible for everyone.

cncl.io

v20200501



NEW QUESTION 7

A new Pod is created. Then, the Pod is assigned to a Node. Which Kubernetes component was re-sponsible for determining which Node to assign the Pod to?

- A. kubelet
- B. Scheduler
- C. API Server
- D. Controller manager

Answer: B

Explanation:

<https://kubernetes.io/docs/reference/command-line-tools-reference/kube-scheduler/> Graphical user interface, text, application Description automatically generated

The Kubernetes scheduler is a control plane process which assigns Pods to Nodes. The scheduler determines which Nodes are valid placements for each Pod in the scheduling queue according to constraints and available resources. The scheduler then ranks each valid Node and binds the Pod to a suitable Node. Multiple different schedulers may be used within a cluster; kube-scheduler is the reference implementation. See [scheduling](#) for more information about scheduling and the kube-scheduler component.

```
kube-scheduler [flags]
```

NEW QUESTION 8

Which prometheus metric type represents a single number value that can increase and decrease over time?

- A. Gauge

- B. Histogram
- C. Summary
- D. Counter

Answer: A

Explanation:

https://prometheus.io/docs/concepts/metric_types/#gauge Graphical user interface, text Description automatically generated

Gauge

A *gauge* is a metric that represents a single numerical value that can arbitrarily go up and down.

Gauges are typically used for measured values like temperatures or current memory usage, but also "counts" that can go up and down, like the number of concurrent requests.

NEW QUESTION 9

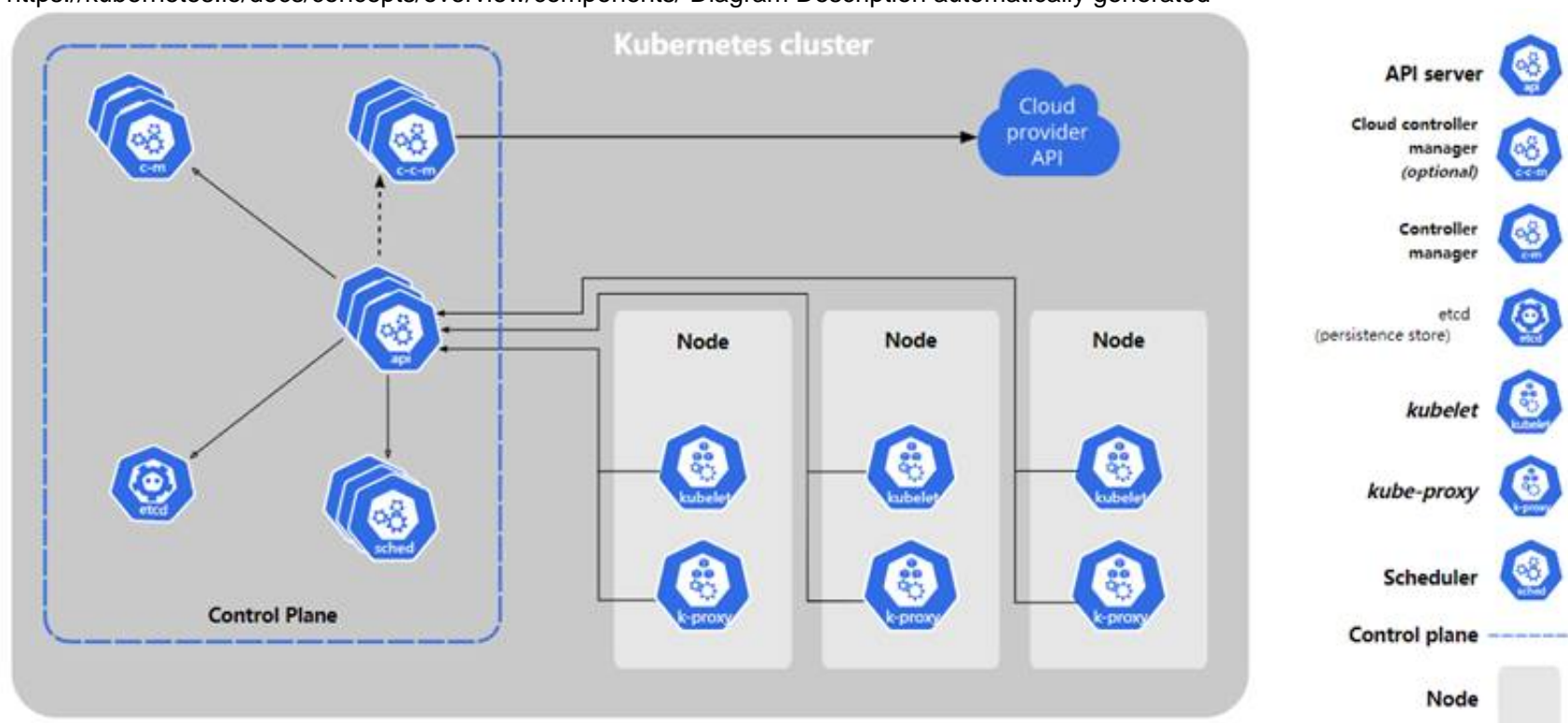
Which of the following components is part of the Kubernetes control panel

- A. kubectl
- B. kube-proxy
- C. Service Mesh
- D. kubelet
- E. Cloud control manager

Answer: E

Explanation:

<https://kubernetes.io/docs/concepts/overview/components/> Diagram Description automatically generated



NEW QUESTION 10

What is the command used to scale the application?

- A. kubectl run
- B. kubectl explain
- C. kubectl scale

Answer: C

Explanation:

<https://kubernetes.io/docs/reference/generated/kubectl/kubectl-commands#scale> Graphical user interface, text, application, email Description automatically generated

scale

Set a new size for a deployment, replica set, replication controller, or stateful set.

Scale also allows users to specify one or more preconditions for the scale action.

If `--current-replicas` or `--resource-version` is specified, it is validated before the scale is attempted, and it is guaranteed that the precondition holds true when the scale is sent to the server.

Usage

```
$ kubectl scale [--resource-version=version] [--current-replicas=count] --replicas=COUNT (-f
FILENAME | TYPE NAME)
```



NEW QUESTION 10

Which of the following is not the part of Kubernetes Control Plane?

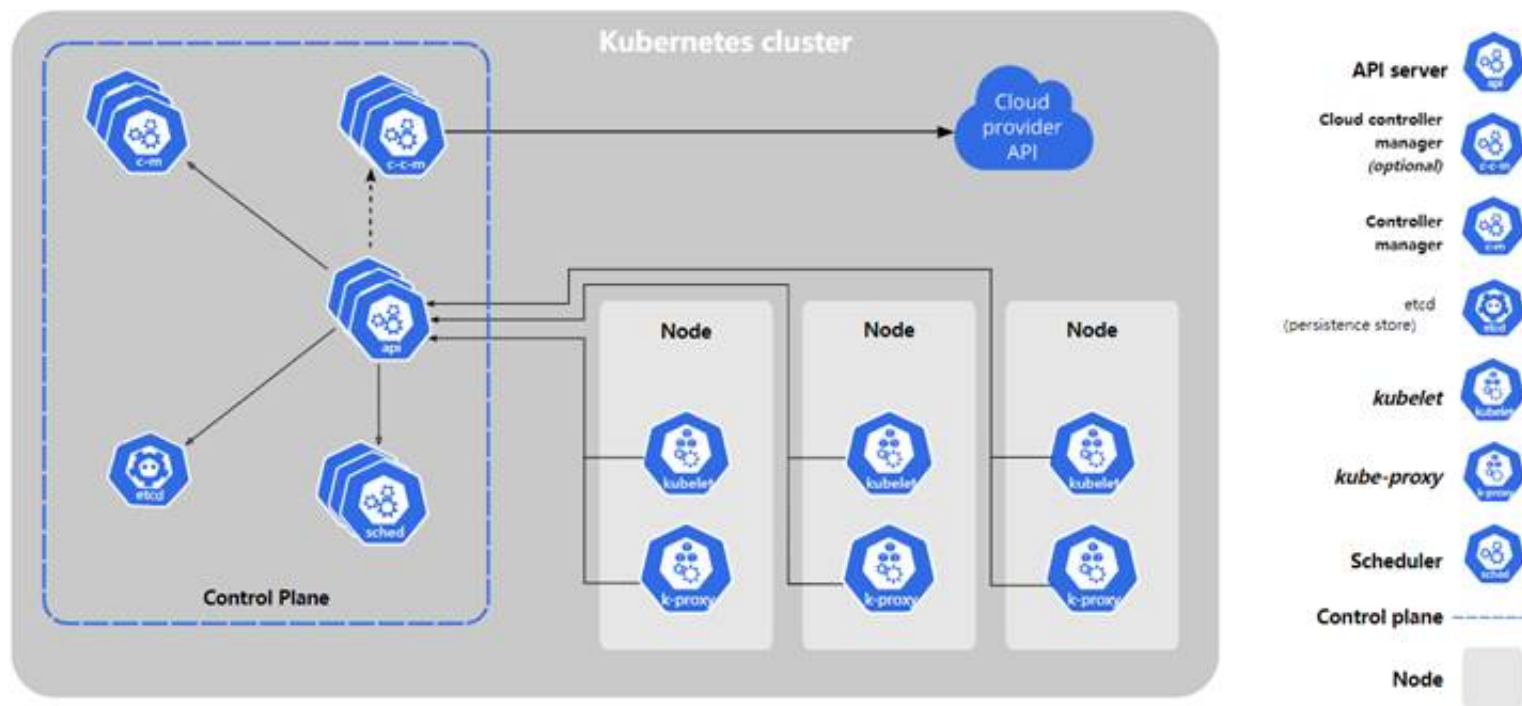
- A. kube scheduler
- B. etcd (pronounce: esty-d)
- C. kube api-server
- D. kube-proxy

Answer: D

Explanation:

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

Graphical user interface, diagram, application Description automatically generated



NEW QUESTION 11

What do control groups provide when it come to containers

- A. Permission
- B. Image Storage
- C. Isolation
- D. Logging

Answer: C

Explanation:

Text Description automatically generated

What is the use of kernel control groups in container technology?

A control group (cgroup) is a Linux kernel feature that limits, accounts for, and isolates the resource usage (CPU, memory, disk I/O, network, and so on) of a collection of processes. Jul 21, 2021

NEW QUESTION 16

Fluentd is the only way to export logs from Kubernetes cluster or applications running in cluster

- A. True

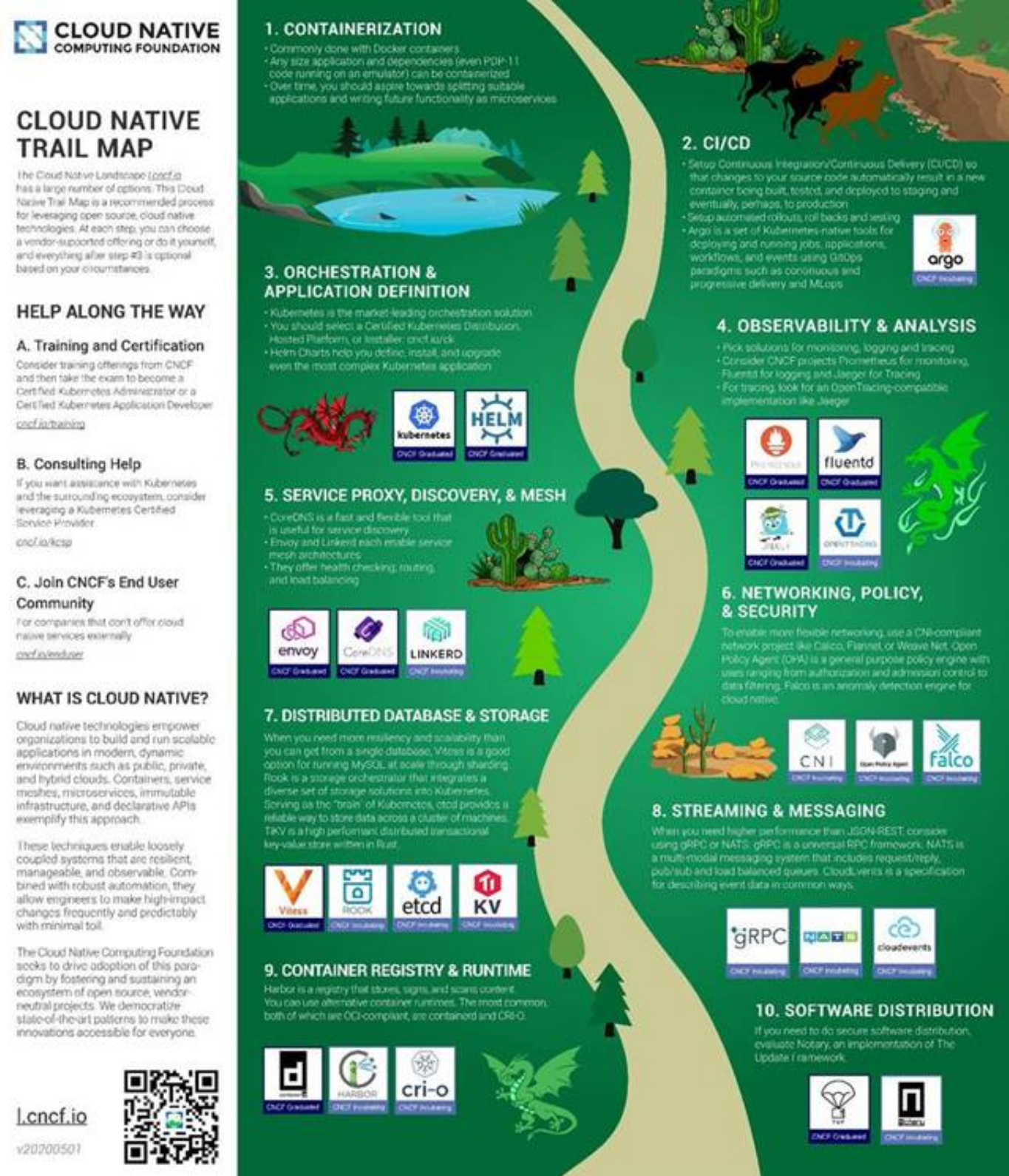
B. False

Answer: B

Explanation:

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

A picture containing timeline Description automatically generated



NEW QUESTION 17

What is OPA?

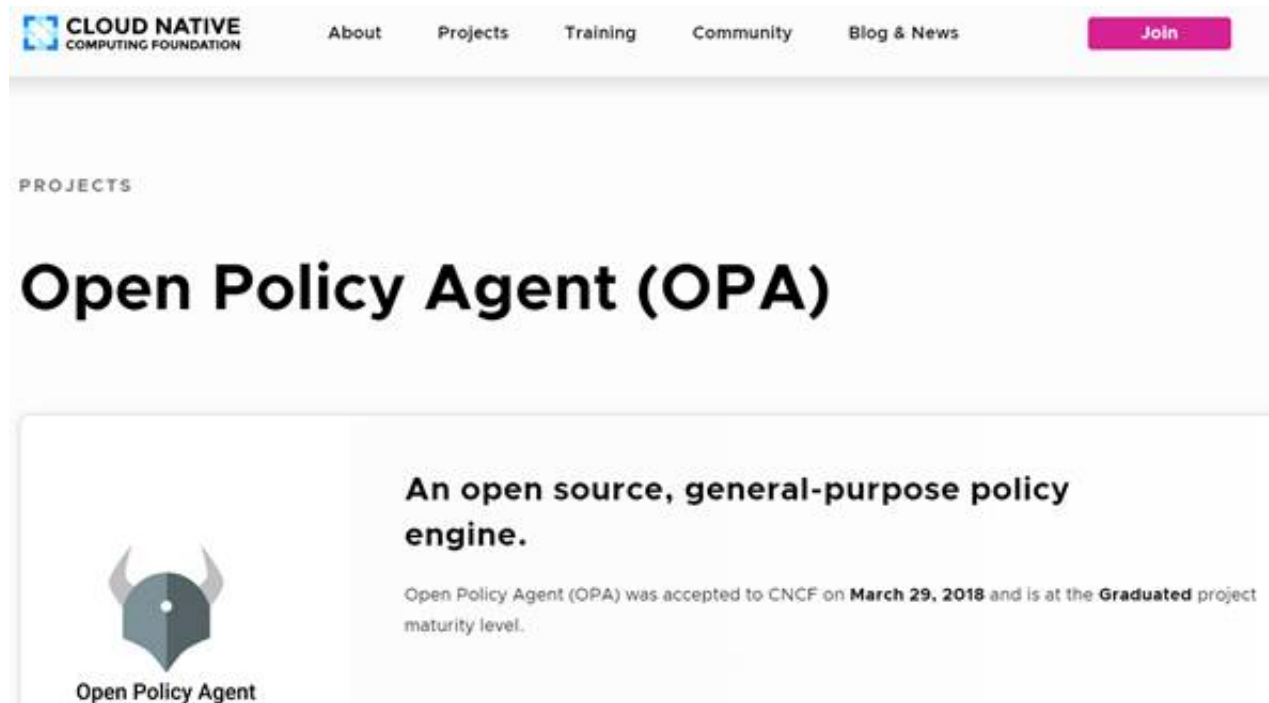
- A. Open Permission Agent
- B. Online Policy Audit
- C. Open Policy Agent
- D. Offline Policy Accessor

Answer: C

Explanation:

<https://www.cncf.io/projects/open-policy-agent-opa/>

Graphical user interface, text, application, email Description automatically generated



NEW QUESTION 22

What is Open Container Initiative 'OCI'?

- A. A protocol for communicating with the kubernetes api
- B. The governing body of the Cloud Native Computing Foundation 'CNCF'
- C. An open standard for managing service mesh in kubernetes
- D. An organization that creates open standards for containers

Answer: D

Explanation:

<https://opencontainers.org/>

Text Description automatically generated

Open Container Initiative

The **Open Container Initiative** is an open governance structure for the express purpose of creating open industry standards around container formats and runtimes.

Established in June 2015 by Docker and other leaders in the container industry, the OCI currently contains three specifications: the Runtime Specification (runtime-spec), the Image Specification (image-spec) and the Distribution Specification (distribution-spec). The Runtime Specification outlines how to run a "filesystem bundle" that is unpacked on disk. At a high-level an OCI implementation would download an OCI Image then unpack that image into an OCI Runtime filesystem bundle. At this point the OCI Runtime Bundle would be run by an OCI Runtime.

NEW QUESTION 24

What is the command to list all the available objects in your Kubernetes cluster?

- A. kubectl get all
- B. kubectl get api-resources
- C. kubectl api-resources
- D. kubectl get pods

Answer: C

Explanation:

<https://kubernetes.io/docs/reference/kubectl/cheatsheet/>

Graphical user interface, text, application, email Description automatically generated

Resource types [↗](#)

List all supported resource types along with their shortnames, [API group](#), whether they are [namespaced](#), and [Kind](#):

```
kubectl api-resources
```

NEW QUESTION 28

Which of the following command is used to get detailed information about the pod?

- A. kubectl info
- B. kubectl get
- C. kubectl describe
- D. kubectl explain

Answer: C

Explanation:

<https://kubernetes.io/docs/reference/generated/kubectl/kubectl-commands#describe> Graphical user interface, application Description automatically generated

Describe a pod

`kubectl describe pods/nginx`

Describe a pod identified by type and name in "pod.json"

`kubectl describe -f pod.json`

Describe all pods

`kubectl describe pods`

NEW QUESTION 30

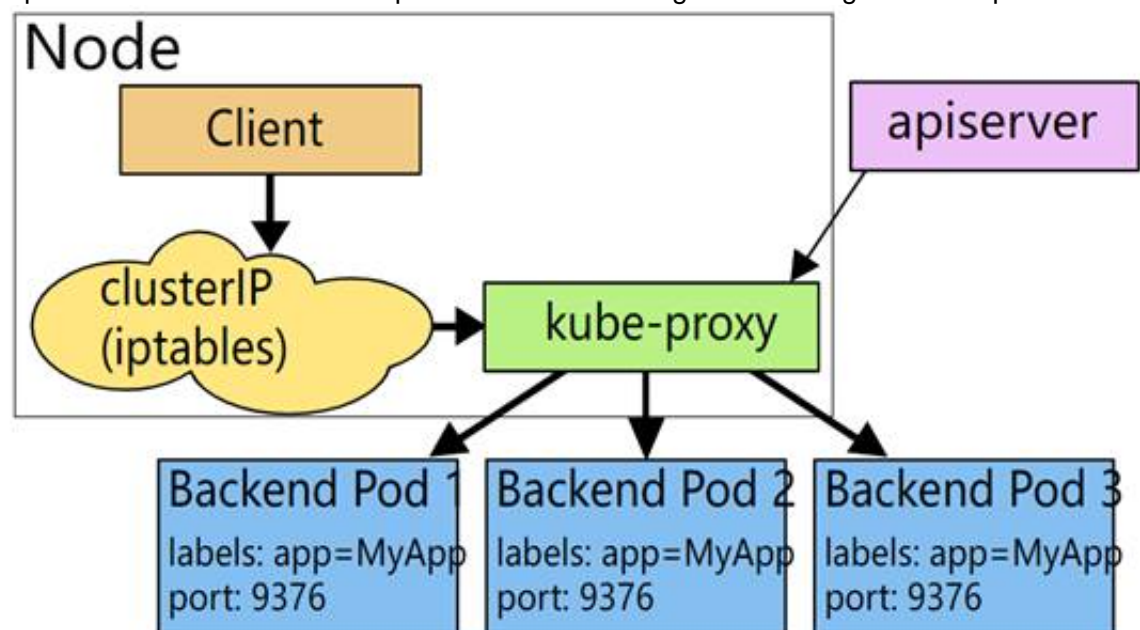
How does service logical group set of pods?

- A. Using hostname
- B. Using label and selectors
- C. Using IP address

Answer: B

Explanation:

<https://kubernetes.io/docs/concepts/services-networking/service/> Diagram Description automatically generated



NEW QUESTION 31

Which role is responsible of creating service level indicator 'SLI', service level objective 'SLO', & Service Level Agreements 'SLA'?

- A. Site reliability engineer 'SRE'
- B. DevOps
- C. GitOps
- D. Security and compliance engineer
- E. Developer

Answer: A

Explanation:

<https://www.atlassian.com/incident-management/kpis/sla-vs-slo-vs-sli> Text Description automatically generated

How does this impact SREs?

For those of you following Google's model and using [Site Reliability Engineering \(SRE\) teams](#) to bridge the gap between development and operations, SLAs, SLOs, and SLIs are foundational to success. SLAs help teams set boundaries and error budgets. SLOs help prioritize work. And SLIs tell SREs when they need to freeze all launches to save an endangered error budget—and when they can loosen up the reins.

NEW QUESTION 35

What makes cloud native technology so important?

- A. It makes data centric
- B. It strengthens team
- C. It removes roadblocks to innovation
- D. It helps gather software requirements
- E. It makes operational centric

Answer: C

Explanation:

<https://github.com/cncf/foundation/blob/main/charter.md>

Graphical user interface, text, application Description automatically generated

Cloud native technologies empower organizations to build and run scalable applications in modern, dynamic environments such as public, private, and hybrid clouds. Containers, service meshes, microservices, immutable infrastructure, and declarative APIs exemplify this approach.

These techniques enable loosely coupled systems that are resilient, manageable, and observable. Combined with robust automation, they allow engineers to make high-impact changes frequently and predictably with minimal toil.

NEW QUESTION 39

Which of the following is an example of vertical scaling?

- A. Using cluster autoscaler
- B. Adding more resources (memory and/or cpu) to a kubernetes node
- C. Adding more nodes to kubernetes cluster
- D. Adding more replica pods to a deployment

Answer: B

Explanation:

<https://kubernetes.io/docs/tasks/run-application/horizontal-pod-autoscale/> Text Description automatically generated

Horizontal scaling means that the response to increased load is to deploy more Pods. This is different from *vertical* scaling, which for Kubernetes would mean assigning more resources (for example: memory or CPU) to the Pods that are already running for the workload.

NEW QUESTION 42

Which is not a service type in Kubernetes?

- A. ClusterIP
- B. NodePort
- C. Ingress
- D. LoadBalancer
- E. ExternalName

Answer: C

Explanation:

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

without a Service. Services allow your applications to receive traffic. Services can be exposed in different ways by specifying a type in the ServiceSpec:

- *ClusterIP* (default) - Exposes the Service on an internal IP in the cluster. This type makes the Service only reachable from within the cluster.
- *NodePort* - Exposes the Service on the same port of each selected Node in the cluster using NAT. Makes a Service accessible from outside the cluster using `<NodeIP>: <NodePort>` . Superset of ClusterIP.
- *LoadBalancer* - Creates an external load balancer in the current cloud (if supported) and assigns a fixed, external IP to the Service. Superset of NodePort.
- *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. This type requires v1.7 or higher of `kube-dns` , or CoreDNS version 0.0.8 or higher.

More information about the different types of Services can be found in the [Using Source IP](#) tutorial. Also see [Connecting Applications with Services](#).

Text Description automatically generated

NEW QUESTION 47

What does the 'kops' acronym means?

- A. Kubernetes Open Platform Specification
- B. Kubernetes Operations
- C. Kubernetes Operators
- D. Kubernetes Operation Policy Specification

Answer: B

Explanation:

<https://github.com/kubernetes/kops>

Graphical user interface, text, application, email Description automatically generated

kOps - Kubernetes Operations

go report A+ reference

The easiest way to get a production grade Kubernetes cluster up and running.

What is kOps?

We like to think of it as `kubect1` for clusters.

`kops` will not only help you create, destroy, upgrade and maintain production-grade, highly available, Kubernetes cluster, but it will also provision the necessary cloud infrastructure.

AWS (Amazon Web Services) and GCE (Google Cloud Platform) are currently officially supported, with DigitalOcean, Hetzner and OpenStack in beta support, and Azure in alpha.

NEW QUESTION 51

What is the default service type in Kubernetes?

- A. CusterIP
- B. NodePort
- C. serviceType
- D. loadBalancer

Answer: A

Explanation:

<https://kubernetes.io/docs/concepts/services-networking/service/#publishing-services-service-types> Graphical user interface, text, application, email Description automatically generated

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.

NEW QUESTION 52

Which of the following is not a stop on the cloud native trailmap?

- A. Microservices
- B. CI/CD
- C. Containerization
- D. Software distribution

Answer: A

Explanation:

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

NEW QUESTION 57

What is etcd used for in Kubernetes?

- A. Integration with cloud platforms

- B. Network routing for the cluster
- C. Kubernetes API security
- D. Backend object storage for the Kubernetes API

Answer: D

Explanation:

etcd serves as a distributed object store that backs the Kubernetes API.

NEW QUESTION 60

You might need to run a stateless application in kubernetes, and you want to be able to scale easily and perform rolling updates. What kubernetes resource type can you use to do this

- A. Dameon set
- B. Replica set
- C. Deployment
- D. pod
- E. service
- F. Stateful set

Answer: C

Explanation:

<https://kubernetes.io/docs/concepts/workloads/controllers/deployment/> Graphical user interface, text, application Description automatically generated

Deployments

A *Deployment* provides declarative updates for Pods and ReplicaSets.

You describe a *desired state* in a Deployment, and the Deployment Controller changes the actual state to the desired state at a controlled rate. You can define Deployments to create new ReplicaSets, or to remove existing Deployments and adopt all their resources with new Deployments.

Note: Do not manage ReplicaSets owned by a Deployment. Consider opening an issue in the main Kubernetes repository if your use case is not covered below.

NEW QUESTION 64

What command to view the kube config?

- A. kubectl view config
- B. kubectl config view
- C. kubectl get kubeconfig

Answer: B

Explanation:

<https://kubernetes.io/docs/reference/generated/kubectl/kubectl-commands#-em-view-em-> Graphical user interface, text, application Description automatically generated

view

Display merged kubeconfig settings or a specified kubeconfig file.

You can use --output jsonpath={...} to extract specific values using a jsonpath expression.

Usage

```
$ kubectl config view
```



NEW QUESTION 69

What are default kubernetes namespaces?

- 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/>

Graphical user interface, text Description automatically generated with medium confidence

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.

NEW QUESTION 72

Notary and the update framework leading security projects in CNCF

- A. TRUE
- B. FALSE

Answer: A

Explanation:

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

A picture containing timeline Description automatically generated



CLOUD NATIVE TRAIL MAP

The Cloud Native Landscape (cncl.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

B. Consulting Help

If you want assistance with Kubernetes and the surrounding ecosystem, consider leveraging a Kubernetes Certified Service Provider. cncf.io/csp

C. Join CNCF's End User Community

For companies that don't offer cloud native services externally. cncf.io/enduser

WHAT IS CLOUD NATIVE?

Cloud native technologies empower organizations to build and run scalable applications in modern, dynamic environments such as public, private, and hybrid clouds. Containers, service meshes, microservices, immutable infrastructure, and declarative APIs exemplify this approach.

These techniques enable loosely coupled systems that are resilient, manageable, and observable. Combined with robust automation, they allow engineers to make high-impact changes frequently and predictably with minimal toil.

The Cloud Native Computing Foundation seeks to drive adoption of this paradigm by fostering and sustaining an ecosystem of open source, vendor-neutral projects. We democratize state-of-the-art patterns to make these innovations accessible for everyone.

cnf.io

v20200501



NEW QUESTION 73

What is horizontal scaling?

- A. Creating a Deployment
- B. Adding resources to existing apps and servers
- C. Moving workloads from one server to another
- D. Adding additional replicas of apps and servers

Answer: D

Explanation:

<https://kubernetes.io/docs/tasks/run-application/horizontal-pod-autoscale/> Text, letter Description automatically generated

In Kubernetes, a *HorizontalPodAutoscaler* automatically updates a workload resource (such as a Deployment or StatefulSet), with the aim of automatically scaling the workload to match demand.

Horizontal scaling means that the response to increased load is to deploy more Pods. This is different from *vertical* scaling, which for Kubernetes would mean assigning more resources (for example: memory or CPU) to the Pods that are already running for the workload.

If the load decreases, and the number of Pods is above the configured minimum, the HorizontalPodAutoscaler instructs the workload resource (the Deployment, StatefulSet, or other similar resource) to scale back down.

Horizontal pod autoscaling does not apply to objects that can't be scaled (for example: a DaemonSet.)

The HorizontalPodAutoscaler is implemented as a Kubernetes API resource and a controller. The resource determines the behavior of the controller. The horizontal pod autoscaling controller, running within the Kubernetes control plane, periodically adjusts the desired scale of its target (for example, a Deployment) to match observed metrics such as average CPU utilization, average memory utilization, or any other custom metric you specify.

NEW QUESTION 77

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

- 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/>

Graphical user interface, text, application, email Description automatically generated

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.

NEW QUESTION 81

Which organizational persona creates Service Level Agreements 'SLA', Service Level Objectives 'SLO', and Service Level Indicator 'SLI'?

- A. Developer
- B. DevSecOps
- C. Site Reliability Engineer (SRE)
- D. Security and Compliance Engineer
- E. DevOps

Answer: C

Explanation:

SREs create SLAs, SLOs, and SLIs to define and implement standards for application and infra-structure reliability.

NEW QUESTION 82

Which access control component of Kubernetes is responsible for authorization and decides what requestor is allowed to do?

- A. Service Account
- B. Role-based access control 'RBAC'
- C. Deployment

Answer: B

Explanation:

<https://kubernetes.io/docs/reference/access-authn-authz/authorization/> Text, letter Description automatically generated

Authorization Modes

The Kubernetes API server may authorize a request using one of several authorization modes:

- **Node** - A special-purpose authorization mode that grants permissions to kubelets based on the pods they are scheduled to run. To learn more about using the Node authorization mode, see [Node Authorization](#).
- **ABAC** - Attribute-based access control (ABAC) defines an access control paradigm whereby access rights are granted to users through the use of policies which combine attributes together. The policies can use any type of attributes (user attributes, resource attributes, object, environment attributes, etc). To learn more about using the ABAC mode, see [ABAC Mode](#).
- **RBAC** - Role-based access control (RBAC) is a method of regulating access to computer or network resources based on the roles of individual users within an enterprise. In this context, access is the ability of an individual user to perform a specific task, such as view, create, or modify a file. To learn more about using the RBAC mode, see [RBAC Mode](#)
 - When specified RBAC (Role-Based Access Control) uses the `rbac.authorization.k8s.io` API group to drive authorization decisions, allowing admins to dynamically configure permission policies through the Kubernetes API.
 - To enable RBAC, start the apiserver with `--authorization-mode=RBAC`.

NEW QUESTION 87

Which command is used to expose Kubernetes service

- A. `kubectl expose`
- B. `kubectl create`
- C. `kubectl run`

Answer: A

Explanation:

<https://kubernetes.io/docs/reference/generated/kubectl/kubectl-commands#expose> Text Description automatically generated

Create a service for a replicated nginx, which serves on port 80 and connects to the containers on port 8000

```
kubectl expose rc nginx --port=80 --target-port=8000
```

NEW QUESTION 89

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

- A. kube-api
- B. kubectl

C. kube-scheduler

Answer: B

Explanation:

<https://kubernetes.io/docs/reference/kubectl/>

Graphical user interface, text, application, email Description automatically generated

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.

NEW QUESTION 91

.....

Thank You for Trying Our Product

We offer two products:

1st - We have Practice Tests Software with Actual Exam Questions

2nd - Questions and Answers in PDF Format

KCNA Practice Exam Features:

- * KCNA Questions and Answers Updated Frequently
- * KCNA Practice Questions Verified by Expert Senior Certified Staff
- * KCNA Most Realistic Questions that Guarantee you a Pass on Your First Try
- * KCNA Practice Test Questions in Multiple Choice Formats and Updates for 1 Year

100% Actual & Verified — Instant Download, Please Click
[Order The KCNA Practice Test Here](#)