

KCNA Dumps

Kubernetes and Cloud Native Associate (KCNA)

<https://www.certleader.com/KCNA-dumps.html>



NEW QUESTION 1

kubeadm is an administrative dashboard for kubernetes

- A. False
- B. True

Answer: A

Explanation:

<https://kubernetes.io/docs/reference/setup-tools/kubeadm/>
Graphical user interface, text, application Description automatically generated

Kubeadm

Kubeadm is a tool built to provide `kubeadm init` and `kubeadm join` as best-practice "fast paths" for creating Kubernetes clusters.

kubeadm performs the actions necessary to get a minimum viable cluster up and running. By design, it cares only about bootstrapping, not about provisioning machines. Likewise, installing various nice-to-have addons, like the Kubernetes Dashboard, monitoring solutions, and cloud-specific addons, is not in scope.



Instead, we expect higher-level and more tailored tooling to be built on top of kubeadm, and ideally, using kubeadm as the basis of all deployments will make it easier to create conformant clusters.

NEW QUESTION 2

What is the most common way to scale the application in the cloud environment?

- A. Parallel Scaling
- B. Horizontal Scaling
- C. Vertical Scaling

Answer: B

Explanation:

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

NEW QUESTION 3

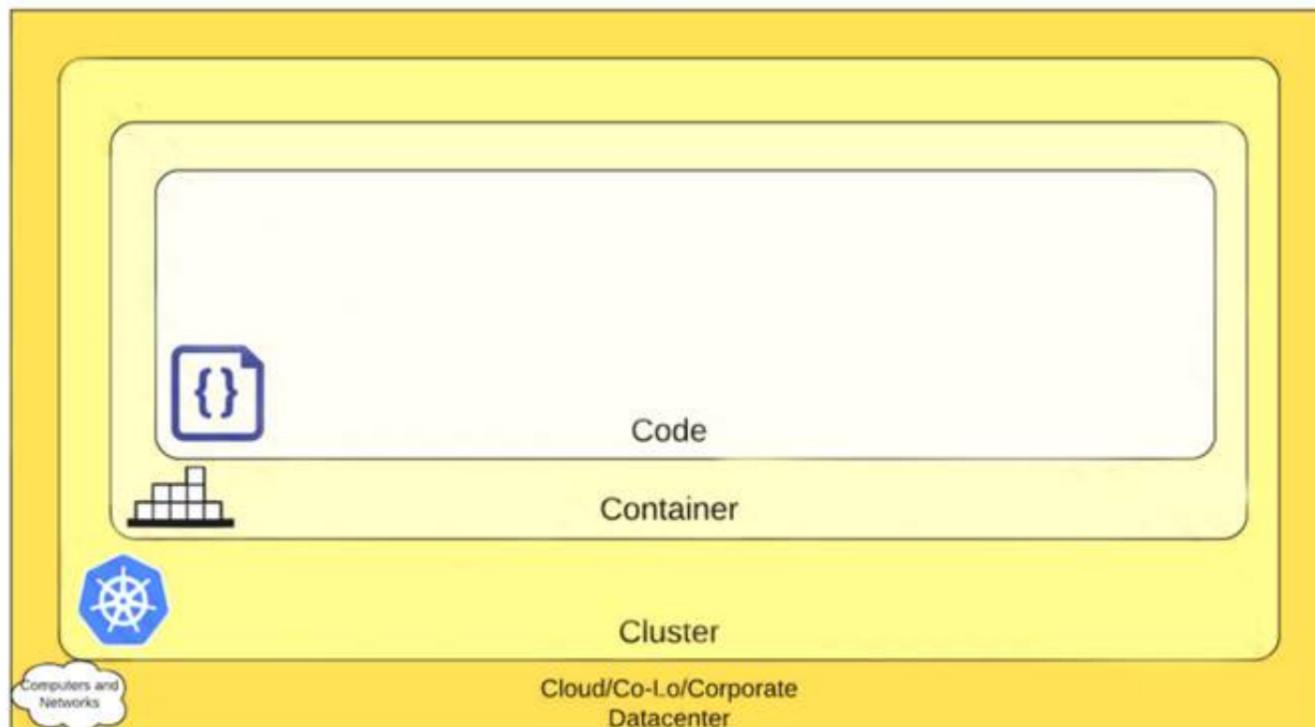
There are three Nodes in a cluster, and want to run exactly one replica of a Pod on each Node. Prefer to automatically create a replica on any new Nodes when they are added. Which Kubernetes re-source should you use?

- A. DaemonSet
- B. ReplicaSet
- C. NodeSet
- D. StatefulSet
- E. Deployment

Answer: A

Explanation:

<https://kubernetes.io/docs/concepts/workloads/controllers/daemonset/>
A DaemonSet runs replicas on all (or just some) Nodes in the cluster.
Table Description automatically generated with medium confidence



NEW QUESTION 4

What tool allows us to build useful visual representations of prometheus data?

- A. Grafana
- B. kubectl
- C. Distributed system tracing
- D. Rook
- E. Kibana

Answer: A

Explanation:

<https://prometheus.io/>
Graphical user interface, text, application Description automatically generated

 Great visualization

Prometheus has multiple modes for visualizing data: a built-in expression browser, Grafana integration, and a console template language.

NEW QUESTION 5

What is the main difference between Argo vs. Flux CD?

- A. Argo is pull-based, and Flux is push-based
- B. No difference; both are pull-based
- C. Argo is push-based, and Flux is pull-based
- D. No difference; both are push-based

Answer: C

Explanation:

ArgoCD: <https://argo-cd.readthedocs.io/en/stable/developer-guide/ci/#can-i-retrigger-the-checks-without-pushing-a-new-c>
FluxCD: <https://fluxcd.io/>

NEW QUESTION 6

Which of the following is used to request storage in Kubernetes?

- A. PersistentVolume 'PV'
- B. PersistentVolumeClaim 'PVC'
- C. Container Storage Interface 'CSI'
- D. StorageClasses

Answer: B

Explanation:

<https://kubernetes.io/docs/concepts/storage/persistent-volumes/>

NEW QUESTION 7

A _____ is an application running on kubernetes.

- A. node
- B. pod
- C. workload
- D. container

Answer: C

Explanation:

<https://kubernetes.io/docs/concepts/workloads/> Text Description automatically generated

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 that level of failure as final: you would need to create a new Pod to recover, even if the node later becomes healthy.

NEW QUESTION 8

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 9

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 COMPUTING FOUNDATION

CLOUD NATIVE TRAIL MAP

The Cloud Native Landscape (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://www.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](https://www.cncf.io/csp)

C. Join CNCF's End User Community
For companies that don't offer cloud native services externally.
[cncf.io/enduser](https://www.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.

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

2. CI/CD
• Setup Continuous Integration/Continuous Delivery (CI/CD) so that changes to your source code automatically result in a new container being built, tested, and deployed to staging and eventually, perhaps, to production
• Setup automated rollouts, rollbacks and testing
• Argo is a set of Kubernetes-native tools for deploying and running jobs, applications, workflows, and events using GitOps paradigms such as continuous and progressive delivery and MLOps

3. ORCHESTRATION & APPLICATION DEFINITION
• Kubernetes is the market-leading orchestration solution
• You should select a Certified Kubernetes Distribution, Hosted Platform, or Installer: [cncf/arkade](#)
• Helm Charts help you define, install, and upgrade even the most complex Kubernetes application

4. OBSERVABILITY & ANALYSIS
• Pick solutions for monitoring, logging and tracing
• Consider CNCF projects Prometheus for monitoring, Fluentd for logging and Jaeger for tracing
• For tracing, look for an OpenTracing-compatible implementation like Jaeger

5. SERVICE PROXY, DISCOVERY, & MESH
• CoreDNS is a fast and flexible tool that is useful for service discovery
• Envoy and Linkerd each enable service mesh architectures
• They offer health checking, routing, and load balancing

6. NETWORKING, POLICY, & SECURITY
To enable more flexible networking, use a CNF-compliant network project like Calico, Flannel, or Weave Net. Open Policy Agent (OPA) is a general purpose policy engine with uses ranging from authorization and admission control to data filtering. Falco is an anomaly detection engine for cloud native.

7. DISTRIBUTED DATABASE & STORAGE
When you need more resiliency and scalability than you can get from a single database, Vitess is a good option for running MySQL at scale through sharding. Rook is a storage orchestrator that integrates a diverse set of storage solutions into Kubernetes. Serving as the "brain" of Kubernetes, etcd provides a reliable way to store data across a cluster of machines. TiKV is a high performance distributed transactional key-value store written in Rust.

8. STREAMING & MESSAGING
When you need higher performance than JSON-RPC, consider using gRPC or NATS. gRPC is a universal RPC framework. NATS is a multi-modal messaging system that includes request/reply, pub/sub and load balanced queues. CloudEvents is a specification for describing event data in common ways.

9. CONTAINER REGISTRY & RUNTIME
Harbor is a registry that stores, signs, and scans content. You can use alternative container runtimes. The most common, both of which are OCI-compliant, are containerd and CRI-O.

10. SOFTWARE DISTRIBUTION
If you need to do secure software distribution, evaluate Notary, an implementation of The Update Framework.

[cncf.io](https://www.cncf.io)
v20200501

NEW QUESTION 10

Which project is not a dominant CNCF project in the storage landscape?

- A. Envoy
- B. Vitess
- C. Rook
- D. TiKV

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 (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

B. Consulting Help

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

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.

cncf.io

v20200501



NEW QUESTION 10

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 12

Which authentication method allows JWTs to authenticate?

- A. OpenId connect
- B. Client 'TLS' certificates
- C. OPA gatekeeper
- D. Anonymous

Answer: A

NEW QUESTION 17

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 20

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

CLOUD NATIVE COMPUTING FOUNDATION

CLOUD NATIVE TRAIL MAP

The Cloud Native Landscape (CNLF) 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.

www.cncf.io
v20200501

1. CONTAINERIZATION

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

2. CI/CD

- Setup Continuous Integration/Continuous Delivery (CI/CD) so that changes to your source code automatically result in a new container being built, tested, and deployed to staging and eventually, perhaps, to production
- Setup automated rollouts, roll backs and testing
- Argo is a set of Kubernetes-native tools for deploying and running jobs, applications, workflows, and events using GitOps paradigms such as continuous and progressive delivery and MLOps

3. ORCHESTRATION & APPLICATION DEFINITION

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

4. OBSERVABILITY & ANALYSIS

- Pick solutions for monitoring, logging and tracing
- Consider CNCF projects Prometheus for monitoring, Fluentd for logging and Jaeger for tracing
- For tracing, look for an OpenTracing-compatible implementation like Jaeger

5. SERVICE PROXY, DISCOVERY, & MESH

- CoreDNS is a fast and flexible tool that is useful for service discovery
- Envoy and Linkerd each enable service mesh architectures
- They offer health checking, routing, and load balancing

6. NETWORKING, POLICY, & SECURITY

To enable more flexible networking, use a CNF-compliant network project like Calico, Flannel, or Weave Net. Open Policy Agent (OPA) is a general purpose policy engine with uses ranging from authorization and admission control to data filtering. Falco is an anomaly detection engine for cloud native.

7. DISTRIBUTED DATABASE & STORAGE

When you need more resiliency and scalability than you can get from a single database, Vitess is a good option for running MySQL, at scale through sharding. Rook is a storage orchestrator that integrates a diverse set of storage solutions into Kubernetes. Serving as the "brain" of Kubernetes, etcd provides a reliable way to store data across a cluster of machines. TiKV is a high performance distributed transactional key-value store written in Rust.

8. STREAMING & MESSAGING

When you need higher performance than JSON-Rest, consider using gRPC or NATS. gRPC is a universal RPC framework. NATS is a multi-modal messaging system that includes request/reply, pub/sub and load balanced queues. CloudEvents is a specification for describing event data in common ways.

9. CONTAINER REGISTRY & RUNTIME

Harbor is a registry that stores, signs, and scans content. You can use alternative container runtimes. The most common, both of which are OCI-compliant, are containerd and CRI-O.

10. SOFTWARE DISTRIBUTION

If you need to do secure software distribution, evaluate Notary, an implementation of The Update Framework.

NEW QUESTION 22

What are the two major components of service mesh?

- A. Control plane and Data plane
- B. Master plane and Data plane
- C. None of the options
- D. Controller plane and User plane
- E. Master plane and User plane

Answer: A

Explanation:

<https://istio.io/latest/about/service-mesh/>

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

How it Works

Istio has two components: the data plane and the control plane.

The data plane is the communication between services. Without a service mesh, the network doesn't understand the traffic being sent over, and can't make any decisions based on what type of traffic it is, or who it is from or to.

NEW QUESTION 24

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 26

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 31

What command can you use to get documentation about a resource type from the command line?

- A. kubectl api-resources
- B. kubectl explain
- C. kubectl get
- D. kubeadm get-resource

Answer: B

Explanation:

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

explain

List the fields for supported resources.

This command describes the fields associated with each supported API resource. Fields are identified via a simple JSONPath identifier:

<type>.<fieldName>[.<fieldName>]

Add the --recursive flag to display all of the fields at once without descriptions. Information about each field is retrieved from the server in OpenAPI format.

Use "kubectl api-resources" for a complete list of supported resources.

Usage

\$ kubectl explain RESOURCE

Get the documentation of the resource and its fields

`kubectl explain pods`

Get the documentation of a specific field of a resource

`kubectl explain pods.spec.containers`

NEW QUESTION 36

Which of the following is not the Kubernetes AutoScaling Strategy?

- A. Horizontal Pod Autoscaler
- B. Cluster Autoscaler
- C. Vertical Pod Autoscaler
- D. Load Balancing AutoScaler

Answer: D

Explanation:

<https://learnk8s.io/kubernetes-autoscaling-strategies>

Graphical user interface, text Description automatically generated with medium confidence

In Kubernetes, several things are referred to as "autoscaling", including:

- [Horizontal Pod Autoscaler.](#)
- [Vertical Pod Autoscaler.](#)
- [Cluster Autoscaler.](#)

NEW QUESTION 37

What framework allows developers to write code without worrying about the servers and operating systems they will run on?

- A. Virtualization
- B. Docker
- C. Serverless
- D. Kubernetes

Answer: C

NEW QUESTION 42

How would you return all the pod data in the json format using kubectl command?

- A. kubectl get pods -o json
- B. kubectl get pods --all-namespaces
- C. kubectl get pods -o wide
- D. kubectl get pods -o jsonpath

Answer: A

Explanation:

<https://kubernetes.io/docs/reference/generated/kubectl/kubectl-commands#get>

NEW QUESTION 46

Flux is built using which toolkit?

- A. CI/CD
- B. DevSecOps
- C. GitOps
- D. DevOps

Answer: C

Explanation:

<https://fluxcd.io/>

Graphical user interface, text, application Description automatically generated

**Flux provides
GitOps for both
apps and
infrastructure**

Flux and Flagger deploy apps with canaries, feature flags, and A/B rollouts. Flux can also manage any Kubernetes resource. Infrastructure and workload dependency management is built in.

**Just push to Git
and Flux does the
rest**

Flux enables application deployment (CD) and (with the help of Flagger) progressive delivery (PD) through automatic reconciliation. Flux can even push back to Git for you with automated container image updates to Git (image scanning and patching).

NEW QUESTION 47

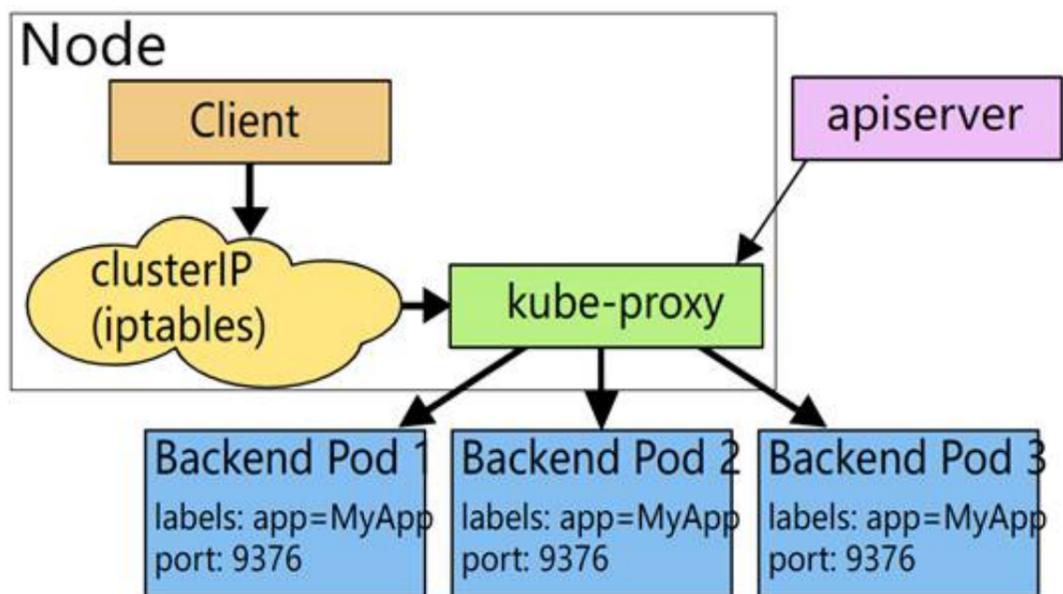
How does service logical group set of pods?

- 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 49

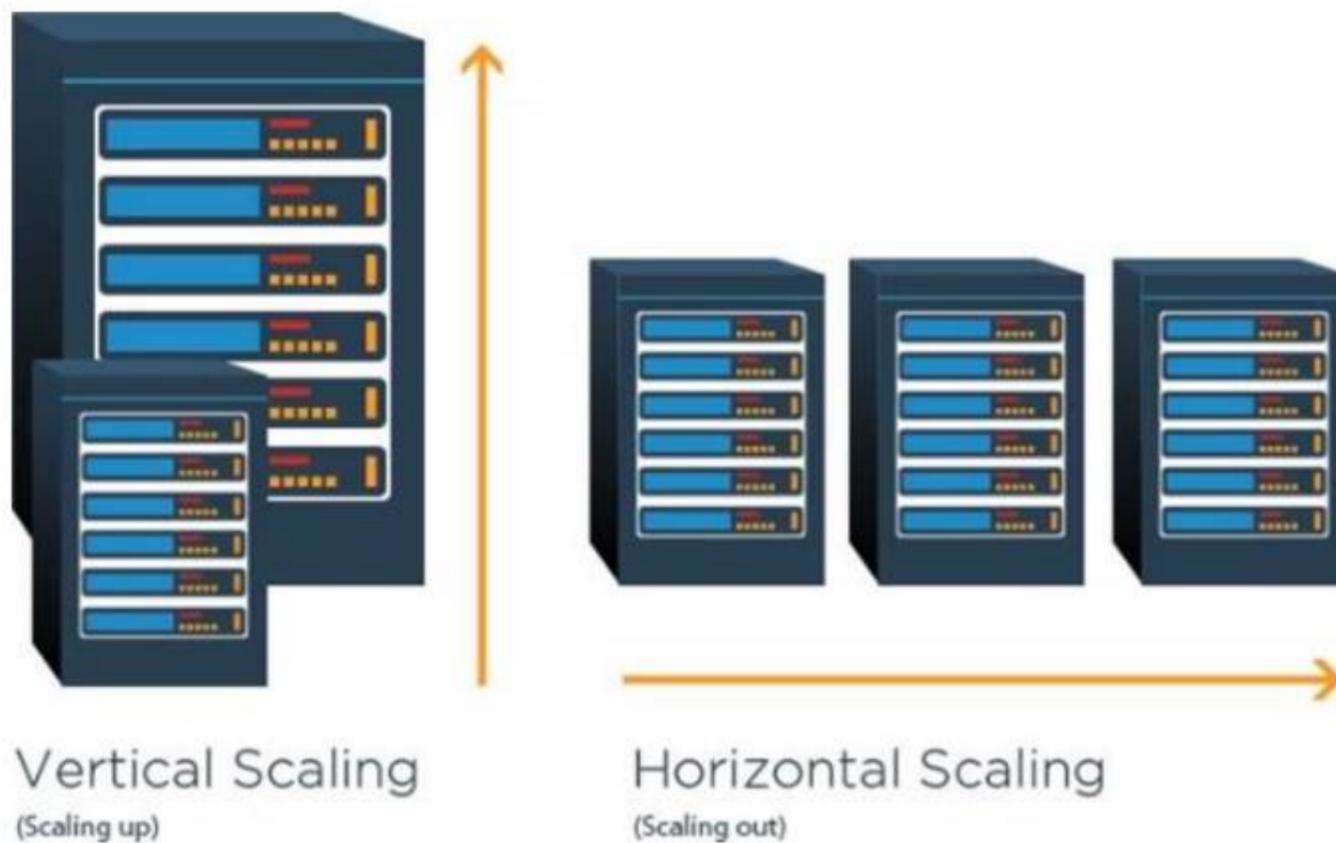
An application that is nearing its usage limit. To increase the amount of users it can handle, you allo-cate additional memory resources to each instance of the application. What type of scaling is this?

- A. Horizontal Scaling
- B. Cluster Autoscaling
- C. Recursive Scaling
- D. Vertical Scaling

Answer: D

Explanation:

Graphical user interface, diagram Description automatically generated



NEW QUESTION 52

What are cluster-wide objects

- A. Service and Pods
- B. Volumes and Nodes
- C. ConfigMaps and Secrets

Answer: B

Explanation:

https://kubernetes.io/docs/concepts/overview/working-with-objects/_print/ Text, letter Description automatically generated

4 - Namespaces

In Kubernetes, *namespaces* provides a mechanism for isolating groups of resources within a single cluster. Names of resources need to be unique within a namespace, but not across namespaces. Namespace-based scoping is applicable only for namespaced objects (e.g. *Deployments*, *Services*, etc) and not for cluster-wide objects (e.g. *StorageClass*, *Nodes*, *PersistentVolumes*, etc).

NEW QUESTION 56

What feature is used for selecting the container runtime configuration?

- A. RuntimeClass
- B. RuntimeContainer
- C. Runtime
- D. RuntimeConfig

Answer: A

Explanation:

<https://kubernetes.io/docs/concepts/containers/runtime-class/>

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

Runtime Class

FEATURE STATE: Kubernetes v1.20 [stable]

This page describes the RuntimeClass resource and runtime selection mechanism.

RuntimeClass is a feature for selecting the container runtime configuration. The container runtime configuration is used to run a Pod's containers.

Motivation

You can set a different RuntimeClass between different Pods to provide a balance of performance versus security. For example, if part of your workload deserves a high level of information security assurance, you might choose to schedule those Pods so that they run in a container runtime that uses hardware virtualization. You'd then benefit from the extra isolation of the alternative runtime, at the expense of some additional overhead.

You can also use RuntimeClass to run different Pods with the same container runtime but with different settings.

NEW QUESTION 59

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 60

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 62

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 63

Which of the following is an advantage a cloud-native microservices application has over monolithic applications?

- A. Cloud-native microservices applications tend to be faster and more responsive than monolithic applications.
- B. Cloud-native microservice applications tend to be easier to troubleshoot.
- C. Cloud-native microservice applications tend to be easier to scale and perform updates on.

Answer: C

Explanation:

Cloud-native applications tend to be microservice base, they have individual services that can be independently scaled, updated and rolled back. This makes scaling and update operations simpler and less risky.

NEW QUESTION 68

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 72

What is the default service type in Kubernetes?

- A. ClusterIP
- 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 77

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 79

Have a pod 'hello' and a container in that pod 'green'. Which of the following commands would get the logs for that container?

- A. `alias k='kubectl'`
`k logs -p hello -c green`

- B. alias k='kubect'l'k logs hello -c green
- C. alias k='kubect'l'k get logs -p hello -c green
- D. alias k='kubect'l'k logs -p hello green

Answer: B

Explanation:

<https://kubernetes.io/docs/reference/generated/kubect'l/kubect'l-commands#logs> Graphical user interface, text, application, email Description automatically generated

logs

Print the logs for a container in a pod or specified resource. If the pod has only one container, the container name is optional.

Usage

```
$ kubect'l logs [-f] [-p] (POD | TYPE/NAME) [-c CONTAINER]
```

Flags

Name	Shorthand	Default	Usage
all-containers		false	Get all containers' logs in the pod(s).
container	c		Print the logs of this container
follow	f	false	Specify if the logs should be streamed.

Return snapshot logs from pod nginx with only one container

```
kubect'l logs nginx
```

Return snapshot logs from pod nginx with multi containers

```
kubect'l logs nginx --all-containers=true
```

Return snapshot logs from all containers in pods defined by label app=nginx

```
kubect'l logs -l app=nginx --all-containers=true
```

Return snapshot of previous terminated ruby container logs from pod web-1

```
kubect'l logs -p -c ruby web-1
```

Begin streaming the logs of the ruby container in pod web-1

```
kubect'l logs -f -c ruby web-1
```

NEW QUESTION 80

How can you achieve cost optimization in the cloud environment?

- A. Use On Demand instances
- B. Use Spot Instances
- C. Use Reserved Instances
- D. Use Bare Metal

Answer: C

NEW QUESTION 85

What is not semantic versioning?

- A. 1.0.0
- B. 2022-05-04
- C. 1.0.0-alpha
- D. 1.0.0-beta.2

Answer: B

Explanation:

<https://semver.org/>
RegEx SemVer at <https://regex101.com/r/vkijKf/1/>

NEW QUESTION 86

What command to view the kube config?

- A. kubect'l view config
- B. kubect'l config view
- C. kubect'l get kubeconfig

Answer: B

Explanation:

<https://kubernetes.io/docs/reference/generated/kubect'l/kubect'l-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 87

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

- A. Containers
- B. REST
- C. YAML
- D. ETCD

Answer: B

Explanation:

Kubernetes objects are RESTful objects.

NEW QUESTION 89

Continuous delivery is .

- A. Manually deploying the code
- B. Coding, Building and Testing the code
- C. Automatically deploying code to [container or server] environment

Answer: C

NEW QUESTION 93

What is the name for a service that has no clusterIp address?

- A. Headless
- B. NodePort
- C. ClusterIP
- D. LoadBalancer

Answer: A

Explanation:

<https://kubernetes.io/docs/concepts/services-networking/service/#headless-services>
Text, letter Description automatically generated

Headless Services

Sometimes you don't need load-balancing and a single Service IP. In this case, you can create what are termed "headless" Services, by explicitly specifying "None" for the cluster IP (`.spec.clusterIP`).

You can use a headless Service to interface with other service discovery mechanisms, without being tied to Kubernetes' implementation.

For headless Services , a cluster IP is not allocated, kube-proxy does not handle these Services, and there is no load balancing or proxying done by the platform for them. How DNS is automatically configured depends on whether the Service has selectors defined:

NEW QUESTION 95

Which of the following provides cloud-native storage orchestration?

- A. Cloud Provider Specific storage (EBS, EFS, Cloud Storage)
- B. Cloud Storage
- C. Storage IO

Answer: A

Explanation:

<https://kubernetes.io/docs/concepts/storage/persistent-volumes/#types-of-persistent-volumes> Table Description automatically generated with medium confidence

Types of Persistent Volumes

PersistentVolume types are implemented as plugins. Kubernetes currently supports the following plugins:

- `awsElasticBlockStore` - AWS Elastic Block Store (EBS)
- `azureDisk` - Azure Disk
- `azureFile` - Azure File
- `cephfs` - CephFS volume
- `csi` - Container Storage Interface (CSI)
- `fc` - Fibre Channel (FC) storage
- `gcePersistentDisk` - GCE Persistent Disk
- `glusterfs` - Glusterfs volume
- `hostPath` - HostPath volume (for single node testing only; WILL NOT WORK in a multi-node cluster; consider using `local` volume instead)
- `iscsi` - iSCSI (SCSI over IP) storage
- `local` - local storage devices mounted on nodes.
- `nfs` - Network File System (NFS) storage
- `portworxVolume` - Portworx volume
- `rbd` - Rados Block Device (RBD) volume
- `vsphereVolume` - vSphere VMDK volume

The following types of PersistentVolume are deprecated. This means that support is still available but will be removed in a future Kubernetes release.

- `cinder` - Cinder (OpenStack block storage) (**deprecated** in v1.18)

NEW QUESTION 99

Which is NOT a use case for the Kubernetes dashboard?

- A. Troubleshooting any issues with applications
- B. Managing running applications
- C. Installing new Kubernetes cluster
- D. Managing the entire Kubernetes cluster

Answer: C

NEW QUESTION 100

What cloud-native construct does a kubernetes pod wrap?

- A. Container
- B. Virtual Machine (VM)
- C. side car process
- D. Docker image

Answer: A

Explanation:

Kubernetes is an orchestrator of containerized apps. However, containers must be wrapped in pods before they can be deployed on kubernetes.

NEW QUESTION 101

Which control plane component is responsible for scheduling pods?

- A. kube-proxy
- B. kube scheduler
- C. kubelet
- D. kube api-server

Answer: B

Explanation:

<https://kubernetes.io/docs/concepts/overview/components/>
Graphical user interface, text, application Description automatically generated

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.

NEW QUESTION 102

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 COMPUTING FOUNDATION

CLOUD NATIVE TRAIL MAP

The Cloud Native Landscape (CNLF) 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/kspp

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.

cncf.io
v20200501

- 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
- 2. CI/CD**
 - Setup Continuous Integration/Continuous Delivery (CI/CD) so that changes to your source code automatically result in a new container being built, tested, and deployed to staging and eventually, perhaps, to production
 - Setup automated rollouts, roll backs and testing
 - Argo is a set of Kubernetes-native tools for deploying and running jobs, applications, workflows, and events using GitOps paradigms such as continuous and progressive delivery and MLops
- 3. ORCHESTRATION & APPLICATION DEFINITION**
 - Kubernetes is the market leading orchestration solution
 - You should select a Certified Kubernetes Distribution, Hosted Platform, or Installer: cncf.io/ckd
 - Helm Charts help you define, install, and upgrade even the most complex Kubernetes application
- 4. OBSERVABILITY & ANALYSIS**
 - Pick solutions for monitoring, logging and tracing
 - Consider CNCF projects Prometheus for monitoring, Fluentd for logging and Jaeger for Tracing
 - For tracing, look for an OpenTracing-compatible implementation like Jaeger
- 5. SERVICE PROXY, DISCOVERY, & MESH**
 - Coredns is a fast and flexible tool that is useful for service discovery
 - Envoy and Linkerd each enable service mesh architectures
 - They offer health checking, routing, and load balancing
- 6. NETWORKING, POLICY, & SECURITY**
 - To enable more flexible networking, use a CNF-compliant network project like Calico, Flannel, or Weave Net. Open Policy Agent (OPA) is a general purpose policy engine with users ranging from authorization and admission control to data filtering. Falco is an anomaly detection engine for cloud native.
- 7. DISTRIBUTED DATABASE & STORAGE**
 - When you need more resiliency and scalability than you can get from a single database, Vitess is a good option for running MySQL at scale through sharding. Rook is a storage orchestrator that integrates a diverse set of storage solutions into Kubernetes. Serving as the "brain" of Kubernetes, etcd provides a reliable way to store data across a cluster of machines. TiKV is a high performance distributed transactional key-value store written in Rust.
- 8. STREAMING & MESSAGING**
 - When you need higher performance than JSON-RPC, consider using gRPC or NATS. gRPC is a universal RPC framework. NATS is a multi-modal messaging system that includes request/reply, pub/sub and load balanced queues. CloudEvents is a specification for describing event data in common ways.
- 9. CONTAINER REGISTRY & RUNTIME**
 - Harbor is a registry that stores, signs, and scans content. You can use alternative container runtimes. The most common, both of which are OCI-compliant, are containerd and cri-o.
- 10. SOFTWARE DISTRIBUTION**
 - If you need to do secure software distribution, evaluate Notary, an implementation of The Update Framework.

NEW QUESTION 107

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 109

What is a commonly used package manager for kubernetes applications?

- A. npm
- B. apt
- C. helm
- D. kubernetes manifest

Answer: C

Explanation:

<https://helm.sh/>

NEW QUESTION 114

The three typical opentelemetry data is?

- A. Metrics
- B. Traces
- C. Logs
- D. All of the options

Answer: D

Explanation:

<https://opentelemetry.io/docs/concepts/data-sources/> Text Description automatically generated

What is OpenTelemetry?

OpenTelemetry is a set of APIs, SDKs, tooling and integrations that are designed for the creation and management of *telemetry data* such as traces, metrics, and logs. The project provides a vendor-agnostic implementation that can be configured to send telemetry data to the backend(s) of your choice. It supports a variety of popular open-source projects including Jaeger and Prometheus.

NEW QUESTION 116

In distributed system tracing, is the term used to refer to a request as it passes through a single component of the distributed system?

- A. Log
- B. Span
- C. Trace
- D. Bucket

Answer: B

Explanation:

https://www.splunk.com/en_us/data-insider/what-is-distributed-tracing.html Text, letter Description automatically generated

How does distributed tracing work?

To quickly grasp how distributed tracing works, it's best to look at how it handles a single request. Tracing starts the moment an end user interacts with an application. When the user sends an initial request — an HTTP request, to use a common example — it is assigned a unique trace ID. As the request moves through the host system, every operation performed on it (called a “span” or a “child span”) is tagged with that first request's trace ID, as well as its own unique ID, plus the ID of the operation that originally generated the current request (called the “parent span”).

Each span is a single step on the request's journey and is encoded with important data relating to the microservice process that is performing that operation. These include:

- The service name and address of the process handling the request.
- Logs and events that provide context about the process's activity.
- Tags to query and filter requests by session ID, database host, HTTP method, and other identifiers.
- Detailed stack traces and error messages in the event of a failure.

A distributed tracing tool like Zipkin or Jaeger (both of which we will explore in more detail in a bit) can correlate the data from all the spans and format them into visualizations that are available on request through a web interface.

Now think of a popular online video game with millions of users, the epitome of a modern microservices-driven app. It must track each end user's location, each interaction with other players and the environment, every item the player acquires, end time, and a host of other in-game data. Keeping the game running smoothly would be unthinkable with traditional tracing methods. But distributed request tracing makes it possible.

NEW QUESTION 119

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 120

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

- 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/>

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

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
```

NEW QUESTION 124

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

- A. TRUE
- B. FALSE

Answer: A

Explanation:

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



CLOUD NATIVE TRAIL MAP

The Cloud Native Landscape (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

B. Consulting Help

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

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.

cncf.io

v20200501



NEW QUESTION 126

What is the functionality of the daemon set?

- A. To run a copy of the pod in all the nodes of the cluster
- B. To initialize the pod before starting the main pod
- C. To run a copy of the pod in a single node of the cluster

Answer: A

Explanation:

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

Graphical user interface, text, application Description automatically generated with medium confidence

DaemonSet

A DaemonSet ensures that all (or some) Nodes run a copy of a Pod. As nodes are added to the cluster, Pods are added to them. As nodes are removed from the cluster, those Pods are garbage collected. Deleting a DaemonSet will clean up the Pods it created.

Some typical uses of a DaemonSet are:

- running a cluster storage daemon on every node
- running a logs collection daemon on every node
- running a node monitoring daemon on every node

NEW QUESTION 127

What is scheduling in Kubernetes

- A. Determining when to execute a cron-job
- B. Assigning pods to nodes
- C. Joining a new nodes to the clusters
- D. Setting a time for automated tasks

Answer: B

Explanation:

<https://kubernetes.io/docs/concepts/scheduling-eviction/>

Graphical user interface, application Description automatically generated

Scheduling

- [Kubernetes Scheduler](#)
- [Assigning Pods to Nodes](#)
- [Pod Overhead](#)
- [Taints and Tolerations](#)
- [Scheduling Framework](#)
- [Scheduler Performance Tuning](#)
- [Resource Bin Packing for Extended Resources](#)

NEW QUESTION 130

.....

Thank You for Trying Our Product

* 100% Pass or Money Back

All our products come with a 90-day Money Back Guarantee.

* One year free update

You can enjoy free update one year. 24x7 online support.

* Trusted by Millions

We currently serve more than 30,000,000 customers.

* Shop Securely

All transactions are protected by VeriSign!

100% Pass Your KCNA Exam with Our Prep Materials Via below:

<https://www.certleader.com/KCNA-dumps.html>