

## KCNA Dumps

### Kubernetes and Cloud Native Associate (KCNA)

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



**NEW QUESTION 1**

Which CNCF project is the dominant project with respect to container registries

- A. Envoy
- B. Harbor
- C. Kubernetes
- D. Rook

**Answer:** B

**Explanation:**

<https://goharbor.io/>

**NEW QUESTION 2**

What is container runtime?

- A. The amount of time it takes a container to execute
- B. A container image format
- C. Another term of kubelet or kubectl
- D. Software that runs containers

**Answer:** D

**Explanation:**

<https://www.aquasec.com/cloud-native-academy/container-security/container-runtime/> Text Description automatically generated

## What Is a Container Runtime?

A container runtime, also known as container engine, is a software component that can run containers on a host operating system. In a [containerized architecture](#), container runtimes are responsible for loading container images from a repository, monitoring local system resources, isolating system resources for use of a container, and managing container lifecycle.

Common container runtimes commonly work together with container orchestrators. The orchestrator is responsible for managing clusters of containers, taking care of concerns like container scalability, networking, and security. The container engine takes responsibility for managing the individual containers running on every compute node in the cluster.

Common examples of container runtimes are runC, containerd, Docker, and Windows Containers. There are three main types of container runtimes—low-level runtimes, high-level runtimes, and sandboxed or virtualized runtimes.

**NEW QUESTION 3**

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 4

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

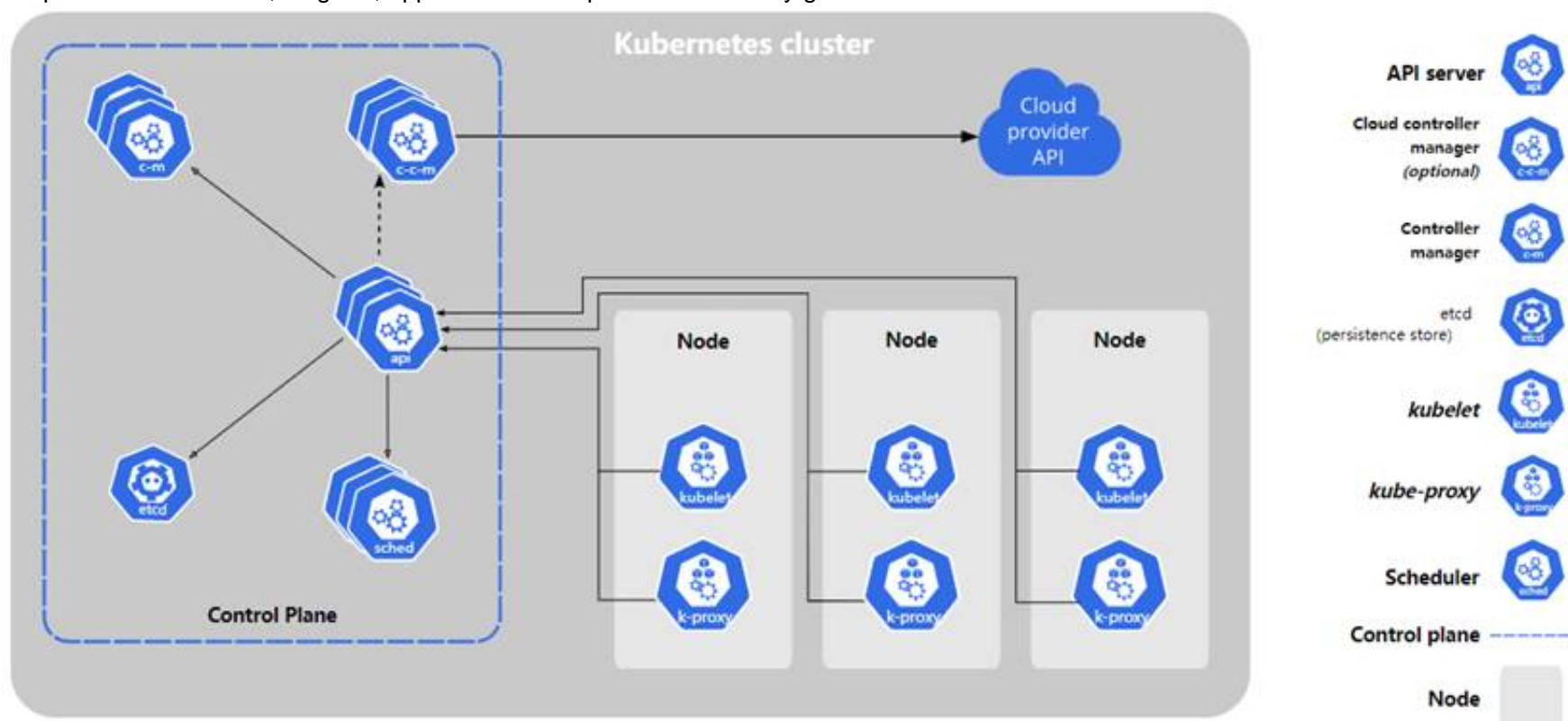
- A. kubelet
- B. systemd
- C. kube-proxy
- D. pod

**Answer:** A

## Explanation:

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

Graphical user interface, diagram, application Description automatically generated



## NEW QUESTION 5

What Kubernetes resource would allow you to run one Pod on some of your Nodes?

- A. DaemonSet
- B. ClusterSet
- C. Deployment
- D. ReplicaSet

**Answer:** A

## Explanation:

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



# 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

In a simple case, one DaemonSet, covering all nodes, would be used for each type of daemon. A more complex setup might use multiple DaemonSets for a single type of daemon, but with different flags and/or different memory and cpu requests for different hardware types.

## NEW QUESTION 6

Which of the following are not the metrics for Site Reliability Engineering?

- A. Service Level Objectives 'SLO'
- B. Service Level Agreements 'SLA'
- C. Service Level Indicators 'SLI'
- D. Service Level Definition 'SLD'

**Answer:** D

### Explanation:

SLI defined quantitative measure of some aspect of the level of service that is provided.

SLOs are key to making data-driven decisions about reliability, they're at the core of SRE practices.

SLAs an explicit or implicit contract with your users that includes consequences of meeting (or missing) the SLOs they contain.

## NEW QUESTION 7

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 8

What kind of limitation cgroups allows?

- A. Prioritization

- B. Resource limiting
- C. Accounting
- D. None of the options
- E. Control
- F. Server cpu and memory

**Answer:** ABCE

#### NEW QUESTION 9

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 10

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 10

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 13

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 ([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 Service Provider. [cncf.io/csp](https://cncf.io/csp)

#### C. Join CNCF's End User Community

For companies that don't offer cloud native services externally [cncf.io/enduser](https://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](https://cncf.io)

v20200501



### NEW QUESTION 14

What is container orchestration?

- A. Packaging code and all of its dependencies into a single executable
- B. Adding code to a container image so it can run as a container
- C. Using automation to manage containers
- D. Spinning a new containers to replace old ones

**Answer: C**

#### Explanation:

<https://www.redhat.com/en/topics/containers/what-is-container-orchestration> Text Description automatically generated

Container orchestration automates the deployment, management, scaling, and networking of containers. Enterprises that need to deploy and manage hundreds or thousands of Linux® containers and hosts can benefit from container orchestration.

Container orchestration can be used in any environment where you use containers. It can help you to deploy the same application across different environments without needing to redesign it. And microservices in containers make it easier to orchestrate services, including storage, networking, and security.



## NEW QUESTION 19

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

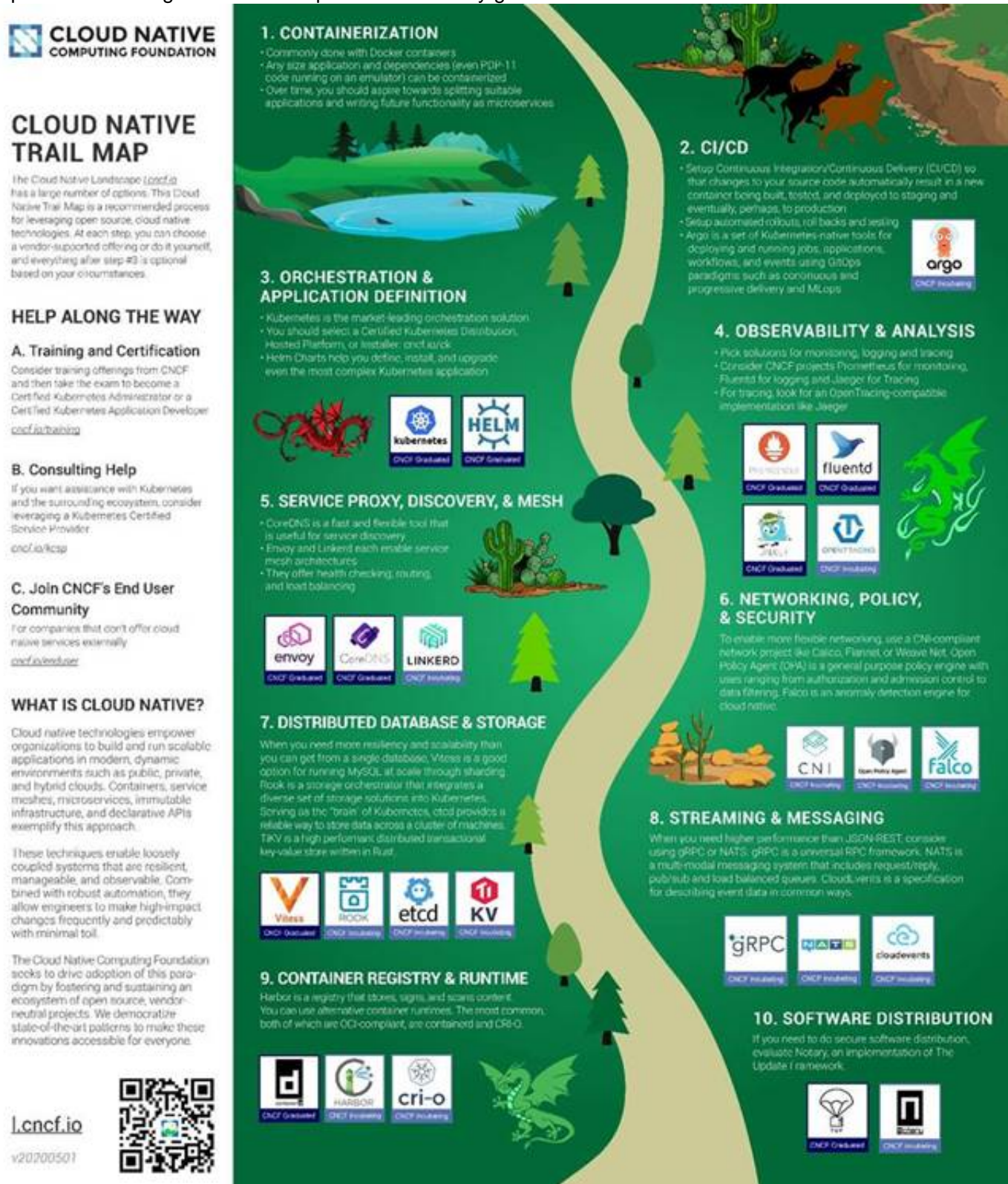
- A. Envoy
- B. Vitess
- C. Rook
- D. TiKV

**Answer: A**

### Explanation:

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

A picture containing timeline Description automatically generated



## NEW QUESTION 23

What tool allows you to create self-managing, self-scaling, self-healing storage?

- A. Persistent Volume
- B. Persistent Volume Claim
- C. Storage Class
- D. Rook
- E. Volume

**Answer: D**

### Explanation:

<https://rook.io/>

Text Description automatically generated



# Storage Operators for Kubernetes

Rook turns distributed storage systems into self-managing, self-scaling, self-healing storage services. It automates the tasks of a storage administrator: deployment, bootstrapping, configuration, provisioning, scaling, upgrading, migration, disaster recovery, monitoring, and resource management.

Rook uses the power of the Kubernetes platform to deliver its services via a Kubernetes Operator for each storage provider.

## NEW QUESTION 28

Which authentication method allows JWTs to authenticate?

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

**Answer:** A

## NEW QUESTION 29

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

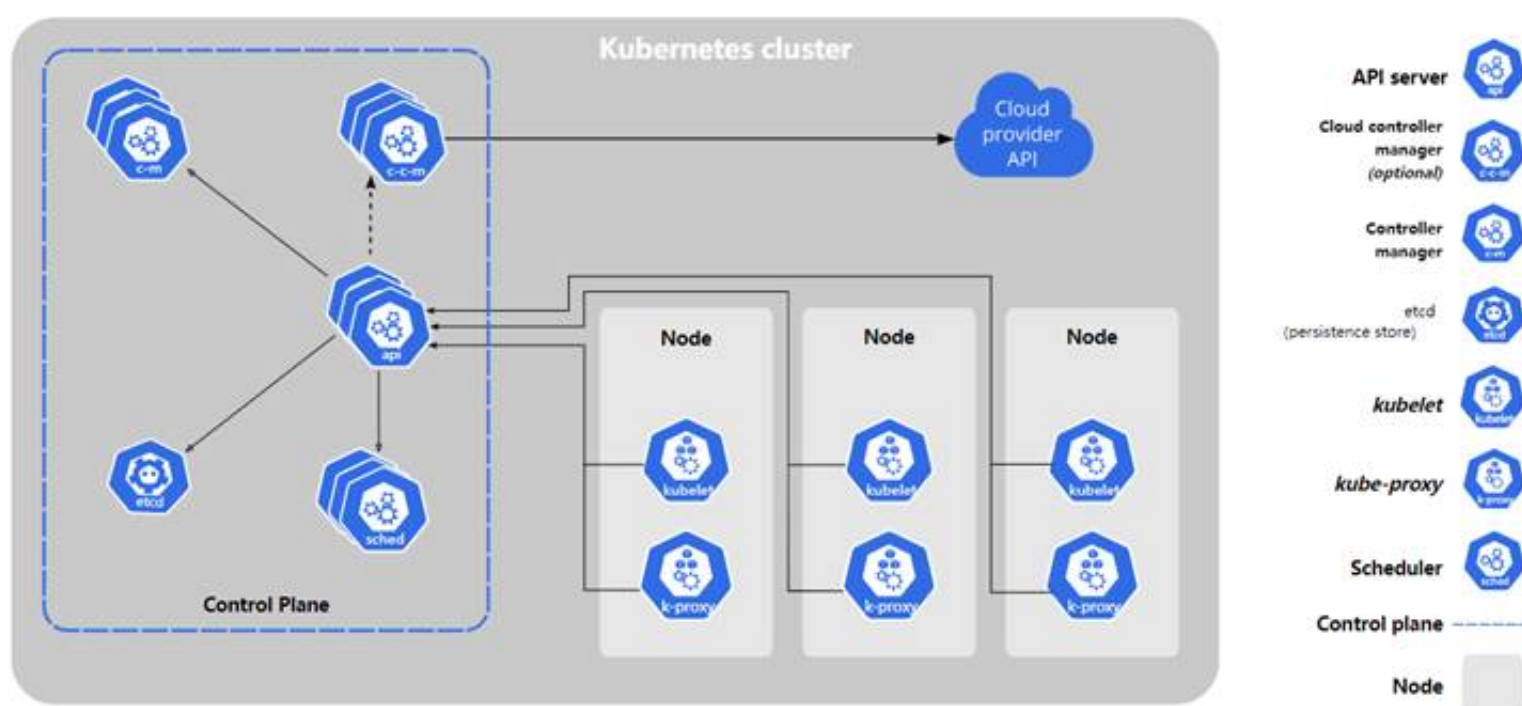
- A. kube scheduler
- B. etcd (pronounce: esty-d)
- C. kube api-server
- D. kube-proxy

**Answer:** D

### Explanation:

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

Graphical user interface, diagram, application Description automatically generated



## NEW QUESTION 30

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 31

What is OPA?

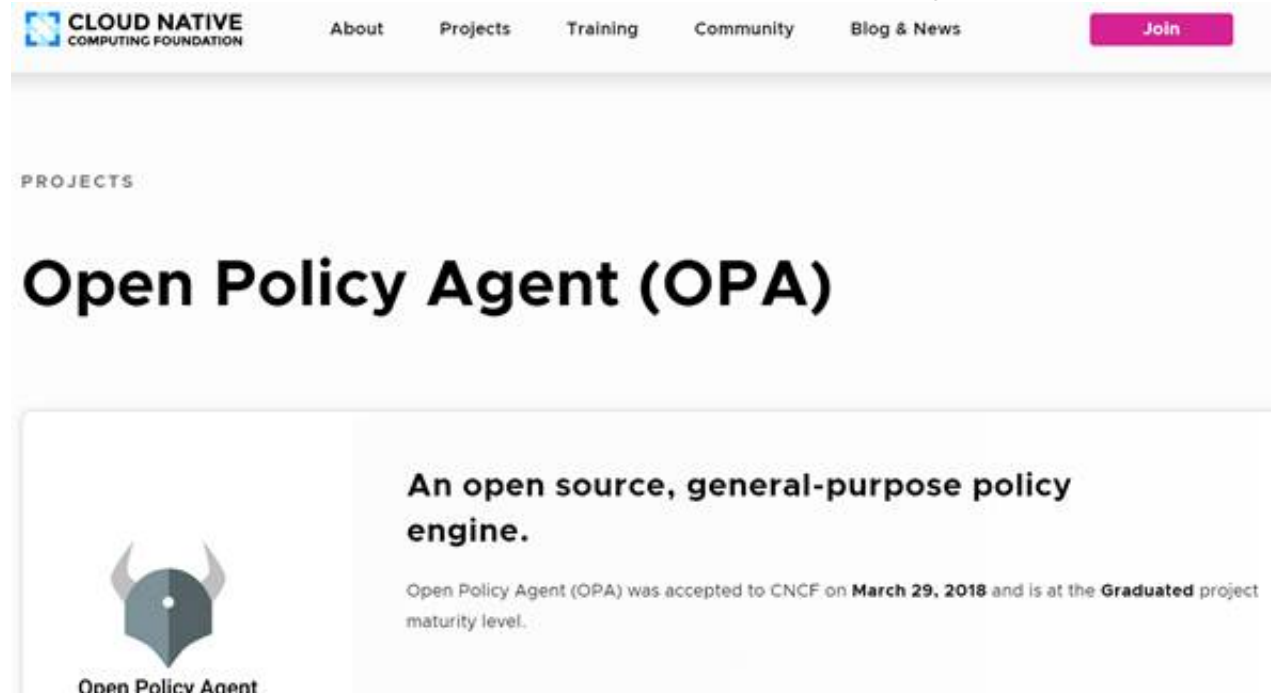
- A. Open Permission Agent
- B. Online Policy Audit
- C. Open Policy Agent
- D. Offline Policy Accessor

**Answer:** C

**Explanation:**

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

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



### NEW QUESTION 34

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 35

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



**NEW QUESTION 39**

What are the two goals of Cloud-Native?

- A. Rapid innovation and automation
- B. Slow innovation and stable applications
- C. Frequent deployments and well-defined organizational silos
- D. Rapid innovation and reliability

**Answer:** D

**Explanation:**

<https://www.redhat.com/en/topics/cloud-native-apps>

**NEW QUESTION 44**

What is the use of labels in Kubernetes?

- A. All of the options
- B. It is used to assign annotation to an object
- C. It is used to assign key-value pair to an object
- D. It is used to assign a name to an object.

**Answer:** C

**Explanation:**

<https://kubernetes.io/docs/concepts/overview/working-with-objects/labels/> Text Description automatically generated

## Labels and Selectors

Labels are key/value pairs that are attached to objects, such as pods.

Labels are intended to be used to specify identifying attributes of objects that are meaningful and relevant to users, but do not directly imply semantics to the core system. Labels can be used to organize and to select subsets of objects. Labels can be attached to objects at creation time and subsequently added and modified at any time. Each object can have a set of key/value labels defined. Each Key must be unique for a given object.

**NEW QUESTION 49**

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 51

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 55

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 58

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 62



Open Container Initiative set container standards for

- A. Code, Build, Distribute, Deploy containers
- B. Run, build, and image
- C. Code, Build, Distribute containers
- D. Run, Build, Distribute containers

**Answer:** D

**NEW QUESTION 66**

Observability and monitoring are not the same?

- A. True
- B. False

**Answer:** A

**NEW QUESTION 67**

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 71**

What is autoscaling?

- A. Automatically measuring resource usage
- B. Automatically assigning workloads to nodes in a cluster
- C. Automatically repairing broken application instances
- D. Automatically adding or removing compute resources as needed

**Answer:** D

**Explanation:**

<https://kubernetes.io/blog/2016/07/autoscaling-in-kubernetes/>

Autoscaling means automatically scaling up or down in response to real-time usage data.

**NEW QUESTION 72**

What kubectl command is used to edit a resource on the server?

- A. kubectl resource modify
- B. kubectl update resource
- C. kubectl edit
- D. kubectl resource edit

**Answer:** C

**Explanation:**

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

## edit

Edit a resource from the default editor.

The edit command allows you to directly edit any API resource you can retrieve via the command-line tools. It will open the editor defined by your KUBE\_EDITOR, or EDITOR environment variables, or fall back to 'vi' for Linux or 'notepad' for Windows. You can edit multiple objects, although changes are applied one at a time. The command accepts file names as well as command-line arguments, although the files you point to must be previously saved versions of resources.

Editing is done with the API version used to fetch the resource. To edit using a specific API version, fully-qualify the resource, version, and group.

The default format is YAML. To edit in JSON, specify "-o json".

The flag --windows-line-endings can be used to force Windows line endings, otherwise the default for your operating system will be used.

In the event an error occurs while updating, a temporary file will be created on disk that contains your unapplied changes. The most common error when updating a resource is another editor changing the resource on the server. When this occurs, you will have to apply your changes to the newer version of the resource, or update your temporary saved copy to include the latest resource version.

Edit the service named 'registry'

```
kubectl edit svc/registry
```

Use an alternative editor

```
KUBE_EDITOR="nano" kubectl edit svc/registry
```

Edit the job 'myjob' in JSON using the v1 API format

```
kubectl edit job.v1.batch/myjob -o json
```

Edit the deployment 'mydeployment' in YAML and save the modified config in its annotation

```
kubectl edit deployment/mydeployment -o yaml --save-config
```

Edit the deployment/mydeployment's status subresource

```
kubectl edit deployment mydeployment --subresource='status'
```

### NEW QUESTION 75

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 79

What CNCF project is the leading DNS project in the CNCF landscape?

- A. Kubernetes
- B. gRPC
- C. KubeDNS
- D. CoreDNS

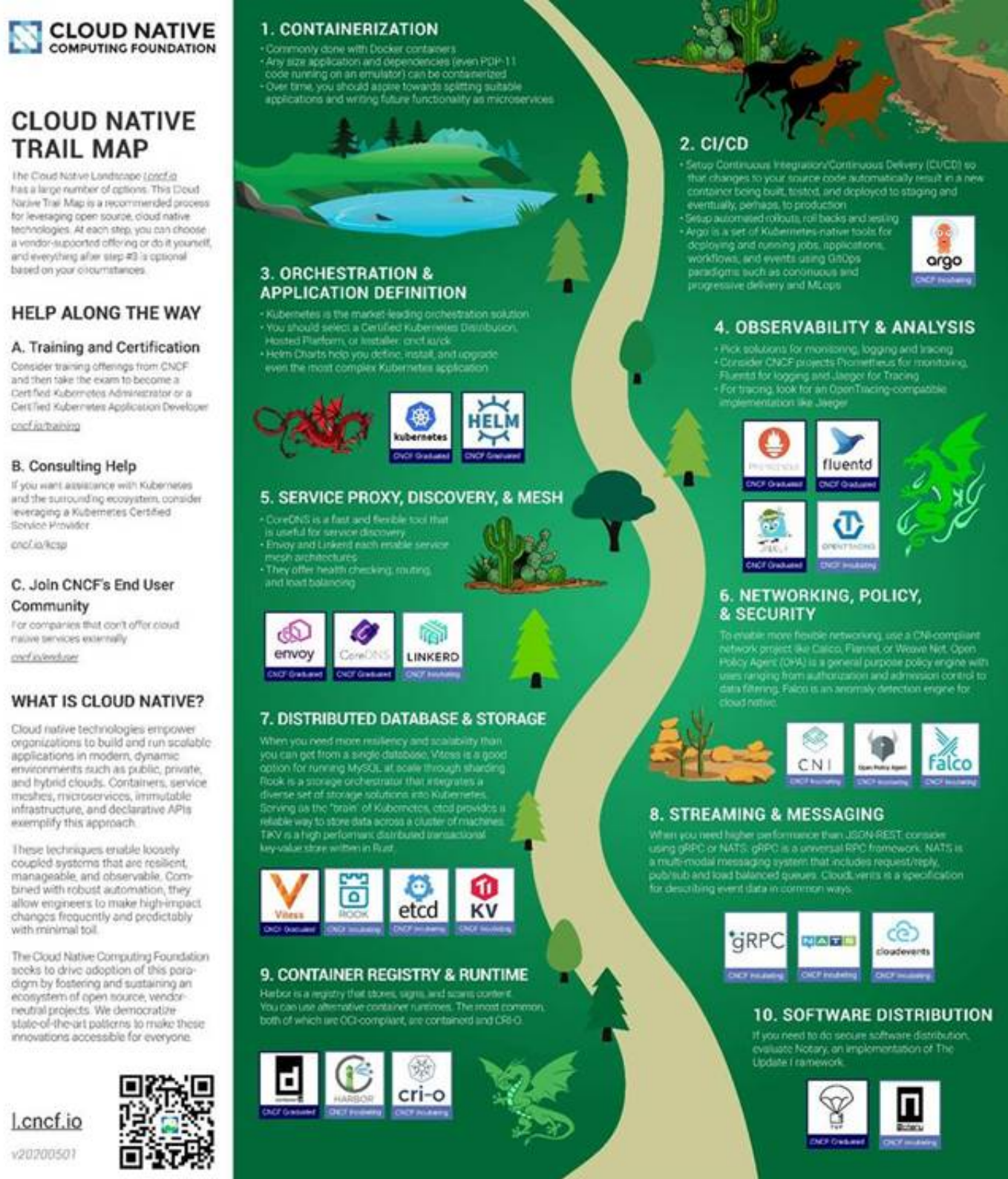


Answer: D

**Explanation:**

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

A picture containing timeline Description automatically generated



**NEW QUESTION 84**

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 87

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 91

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 96

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](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 101

What are container runtimes with Kubernetes?

- A. CRI-O
- B. Ixd

- C. containerd
- D. Dockershim

**Answer:** AC

**Explanation:**

<https://kubernetes.io/docs/setup/production-environment/container-runtimes/> Graphical user interface, text, application, email Description automatically generated

## Container Runtimes

**Note:** Dockershim has been removed from the Kubernetes project as of release 1.24. Read the [Dockershim Removal FAQ](#) for further details.

You need to install a container runtime into each node in the cluster so that Pods can run there. This page outlines what is involved and describes related tasks for setting up nodes.

Kubernetes 1.25 requires that you use a runtime that conforms with the Container Runtime Interface (CRI).

See [CRI version support](#) for more information.

This page provides an outline of how to use several common container runtimes with Kubernetes.

- [containerd](#)
- [CRI-O](#)
- [Docker Engine](#)
- [Mirantis Container Runtime](#)

**Note:**

Kubernetes releases before v1.24 included a direct integration with Docker Engine, using a component named *dockershim*. That special direct integration is no longer part of Kubernetes (this removal was [announced](#) as part of the v1.20 release). You can read [Check whether Dockershim removal affects you](#) to understand how this removal might affect you. To learn about migrating from using dockershim, see [Migrating from dockershim](#).

If you are running a version of Kubernetes other than v1.25, check the documentation for that version.

### NEW QUESTION 102

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](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 Service Provider: [cncf.io/csp](https://cncf.io/csp)

#### C. Join CNCF's End User Community

For companies that don't offer cloud native services externally [cncf.io/enduser](https://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](https://cncf.io)

v20200501



### NEW QUESTION 104

Various Container Orchestrator Systems (COS)?

- A. Apache Mesos
- B. None of the options
- C. Docker Swarm
- D. Kubernetes

**Answer:** ACD

### NEW QUESTION 106

Which of the following is NOT a Kubernetes component?

- A. Scheduler
- B. Docker
- C. Cloud Controller manager
- D. Kube-proxy

**Answer:** B

### Explanation:

Docker is not a Kubernetes component.

### NEW QUESTION 107

What is the command used to login to the pod?

- A. kubectl login
- B. kubectl list
- C. kubectl exec
- D. kubectl get

**Answer:** C

**Explanation:**

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

List contents of /usr from the first container of pod mypod and sort by modification time # If the command you want to execute in the pod has any flags in common (e.g. -i), # you must use two dashes (--) to separate your command's flags/arguments # Also note, do not surround your command and its flags/arguments with quotes # unless that is how you would execute it normally (i.e., do ls -t /usr, not "ls -t /usr")

```
kubectl exec mypod -i -t -- ls -t /usr
```

Text Description automatically generated

**NEW QUESTION 112**

How to get the logs of the previously terminated nginx container from the web pod?

- A. kubectl logs -p -c nginx web
- B. kubectl logs nginx
- C. kubectl logs -p -c web nginx
- D. kubectl logs -f -c nginx web

**Answer:** A

**Explanation:**

<https://kubernetes.io/docs/reference/generated/kubect/kubectl-commands#logs> Text Description automatically generated with medium confidence

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

```
kubectl logs -p -c ruby web-1
```

**NEW QUESTION 114**

Which kubernetes object do deployments use behind the scenes when they need to scale pods?

- A. POD
- B. Deployment
- C. Horizontal pod autoscaler
- D. Api Scheduler
- E. Replicasets

**Answer:** E

**Explanation:**

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

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

## ReplicaSet

A ReplicaSet's purpose is to maintain a stable set of replica Pods running at any given time. As such, it is often used to guarantee the availability of a specified number of identical Pods.

**NEW QUESTION 119**

Which of the following computing model doesn't require you to provision infrastructure?

- A. None of the above
- B. Bare Metal
- C. Compute Engine
- D. Virtual Machines
- E. Serverless

**Answer:** E

**NEW QUESTION 123**

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

- A. etcd (pronounce: esty-d)



- B. api server
- C. kubelet
- D. scheduler

**Answer:** A

**Explanation:**

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

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

## 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](#).

### NEW QUESTION 125

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 129

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 132

.....

## 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>