Linux Foundation KCNA Practice Test Software Gives an Exact Impression of the Real Exam

DOWNLOAD the newest PassReview KCNA PDF dumps from Cloud Storage for free: https://drive.google.com/open?id=1D_vG17A3OswoWPM3ifx0ybhQXi6mf0O2

You won't need anything else if you prepare for the exam with our Linux Foundation KCNA Exam Questions. Our experts have prepared Linux Foundation KCNA dumps questions that will eliminate your chances of failing the exam. We are conscious of the fact that most of the candidates have a tight schedule which makes it tough to prepare for the Linux Foundation KCNA Exam Preparation.

Linux Foundation KCNA (Kubernetes and Cloud Native Associate) Exam is a certification program developed by the Linux Foundation that is designed to test an individual's knowledge and skills in managing and deploying cloud-native applications and infrastructure using Kubernetes. Kubernetes and Cloud Native Associate certification is an entry-level program that is intended for IT professionals who are new to Kubernetes and cloud-native technologies.

>> KCNA Latest Dumps Files <<

Valid KCNA Test Book - KCNA Reliable Test Test

We will continue to pursue our passion for better performance and human-centric technology of latest KCNA quiz prep. And we guarantee you to pass the KCNA exam for we have confidence to make it with our technological strength. A good deal of researches has been made to figure out how to help different kinds of candidates to get the KCNA Certification. We have made classification to those faced with various difficulties, aiming at which we adopt corresponding methods. According to the statistics shown in the feedback chart, the general pass rate for latest KCNA test prep is 98%.

Linux Foundation KCNA Certification Exam is a valuable credential for IT professionals who want to advance their careers in the cloud-native industry. Kubernetes and Cloud Native Associate certification validates the candidate's understanding of Kubernetes and other cloud-native technologies, and it helps them stay up-to-date with the latest trends and best practices in the industry. Kubernetes and Cloud Native Associate certification is also beneficial for organizations that want to ensure that their IT teams have the necessary skills and knowledge to develop and deploy cloud-native applications.

Linux Foundation Kubernetes and Cloud Native Associate Sample Questions (Q159-Q164):

NEW QUESTION # 159
Which component in Kubernetes is responsible to watch newly created Pods with no assigned node, and selects a node for them to run on?

- A. kube-proxy
- B. kube-scheduler
- C. kube-controller-manager
- D. etcd

Answer: B

Explanation:
The correct answer is D: kube-scheduler. The kube-scheduler is the control plane component responsible for assigning Pods to nodes. It watches for newly created Pods that do not have a spec.nodeName set (i.e., unscheduled Pods). For each such Pod, it evaluates the available nodes against scheduling constraints and chooses the best node, then performs a "bind" operation by setting the Pod's spec.nodeName.
Scheduling decisions consider many factors: resource requests vs node allocatable capacity, taints/tolerations, node selectors and affinity/anti-affinity, topology spread constraints, and other policy inputs. The scheduler typically runs a two-phase process: filtering (find feasible nodes) and scoring (rank feasible nodes) before selecting one.
Option A (etcd) is the datastore that persists cluster state; it does not make scheduling decisions. Option B (kube-controller-manager) runs controllers (Deployment, Node, Job controllers, etc.) but not scheduling. Option C (kube-proxy) is a node component for Service networking; it doesn't place Pods.
Understanding this separation is key for troubleshooting. If Pods are stuck Pending with "no nodes available," the scheduler's feasibility checks are failing (insufficient CPU/memory, taints not tolerated, affinity mismatch). If Pods schedule but land unexpectedly, it's often due to scoring preferences or missing constraints. In all cases, the component that performs the node selection is the kube-scheduler.
Therefore, the verified correct answer is D.

NEW QUESTION # 160
Which statement about Secrets is correct?

- A. A Secret is part of a Pod specification.
- B. Secret data is base64 encoded and stored unencrypted by default.
- C. A Secret can only be used for confidential data.
- D. Secret data is encrypted with the cluster private key by default.

Answer: B

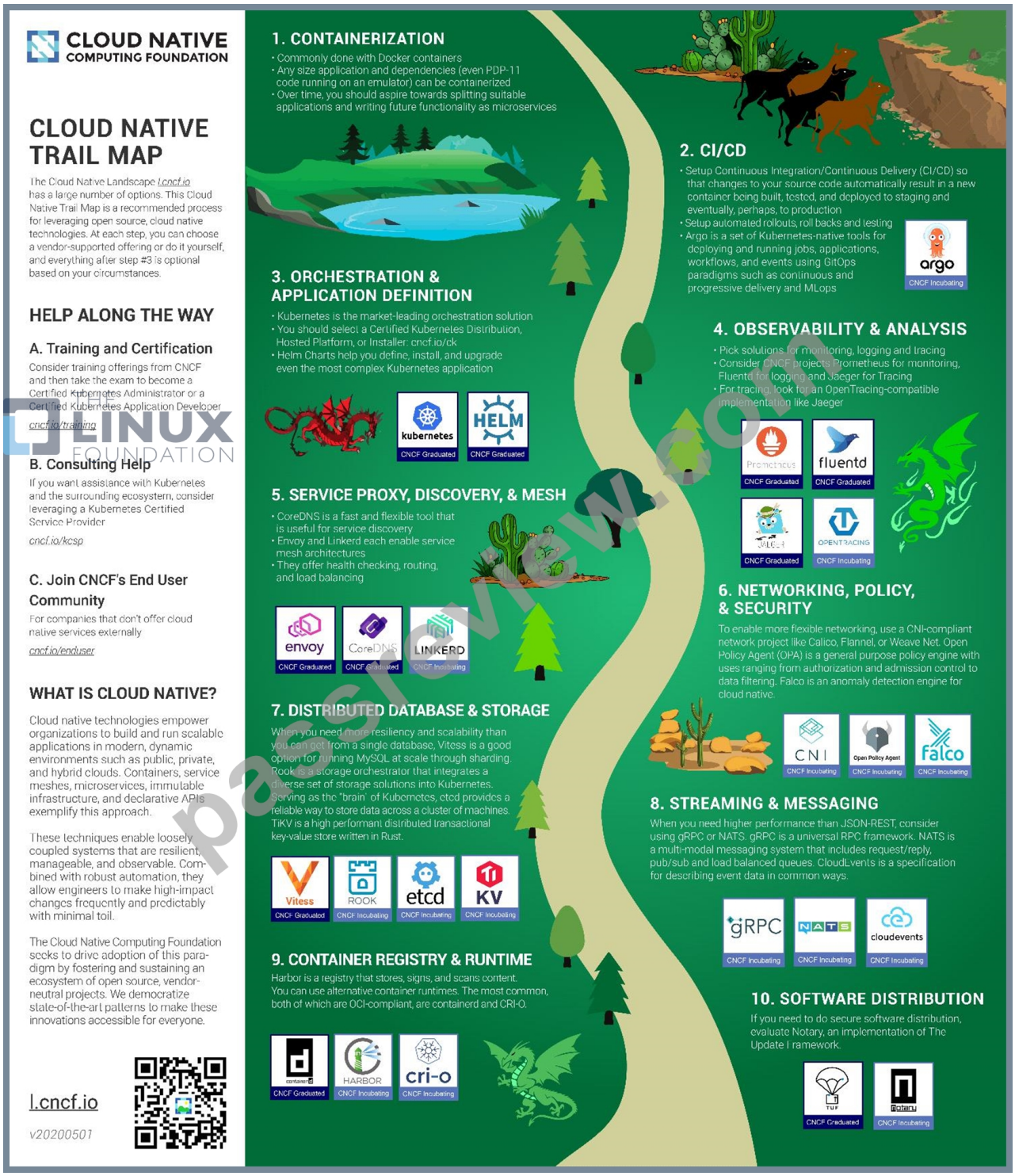
Explanation:
The correct answer is C. By default, Kubernetes Secrets store their data as base64-encoded values in the API (backed by etcd). Base64 is an encoding mechanism, not encryption, so this does not provide confidentiality. Unless you explicitly configure encryption at rest for etcd (via the API server encryption provider configuration) and secure access controls, Secret contents should be treated as potentially readable by anyone with sufficient API access or access to etcd backups.
Option A is misleading: a Secret is its own Kubernetes resource (kind: Secret). While Pods can reference Secrets (as environment variables or mounted volumes), the Secret itself is not "part of the Pod spec" as an embedded object. Option B is incorrect because Kubernetes does not automatically encrypt Secret data with a cluster private key by default; encryption at rest is optional and must be enabled. Option D is incorrect because Secrets can store a range of sensitive or semi-sensitive data (tokens, certs, passwords), but Kubernetes does not enforce "only confidential data" semantics; it's a storage mechanism with size and format constraints.
Operationally, best practices include: enabling encryption at rest, limiting access via RBAC, avoiding broad "list/get secrets" permissions, using dedicated service accounts, auditing access, and considering external secrets managers (Vault, cloud KMS-backed solutions) for higher assurance. Also, don't confuse "Secret" with "secure by default." The default protection is mainly about avoiding accidental plaintext exposure in manifests, not about cryptographic security.
So the only correct statement in the options is C.

NEW QUESTION # 161
Notary and the update framework leading security projects in CNCF

- A. FALSE
- B. TRUE

Answer: B

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



NEW QUESTION # 162
Kubernetes ___ allows you to automatically manage the number of nodes in your cluster to meet demand.

- A. Node Autoscaler
- B. Horizontal Pod Autoscaler
- C. Vertical Pod Autoscaler
- D. Cluster Autoscaler

Answer: D

Explanation:
Kubernetes supports multiple autoscaling mechanisms, but they operate at different layers. The question asks specifically about automatically managing the number of nodes in the cluster, which is the role of the Cluster Autoscaler-therefore B is correct.

Cluster Autoscaler monitors the scheduling state of the cluster. When Pods are pending because there are not enough resources (CPU/memory) available on existing nodes-meaning the scheduler cannot place them- Cluster Autoscaler can request that the underlying infrastructure (typically a cloud provider node group / autoscaling group) add nodes. Conversely, when nodes are underutilized and Pods can be rescheduled elsewhere, Cluster Autoscaler can drain those nodes (respecting disruption constraints like PodDisruptionBudgets) and then remove them to reduce cost. This aligns with cloud-native elasticity: scale infrastructure up and down automatically based on workload needs.
The other options are different: Horizontal Pod Autoscaler (HPA) changes the number of Pod replicas for a workload (like a Deployment) based on metrics (CPU utilization, memory, or custom metrics). It scales the application layer, not the node layer. Vertical Pod Autoscaler (VPA) changes resource requests/limits (CPU
/memory) for Pods, effectively "scaling up/down" the size of individual Pods. It also does not directly change node count, though its adjustments can influence scheduling pressure. "Node Autoscaler" is not the canonical Kubernetes component name used in standard terminology; the widely referenced upstream component for node count is Cluster Autoscaler.
In real systems, these autoscalers often work together: HPA increases replicas when traffic rises; that may cause Pods to go Pending if nodes are full; Cluster Autoscaler then adds nodes; scheduling proceeds; later, traffic drops, HPA reduces replicas and Cluster Autoscaler removes nodes. This layered approach provides both performance and cost efficiency.
==========

NEW QUESTION # 163
Which Kubernetes-native deployment strategy supports zero-downtime updates of a workload?

- A. RollingUpdate
- B. Canary
- C. BlueGreen
- D. Recreate

Answer: A

Explanation:
D (RollingUpdate) is correct. In Kubernetes, the Deployment resource's default update strategy is RollingUpdate, which replaces Pods gradually rather than all at once. This supports zero-downtime updates when the workload is properly configured (sufficient replicas, correct readiness probes, and appropriate maxUnavailable / maxSurge settings). As new Pods come up and become Ready, old Pods are terminated in a controlled way, keeping the service available throughout the rollout.
RollingUpdate's "zero downtime" is achieved by maintaining capacity while transitioning between versions. For example, with multiple replicas, Kubernetes can create new Pods, wait for readiness, then scale down old Pods, ensuring traffic continues to flow to healthy instances. Readiness probes are critical: they prevent traffic from being routed to a Pod until it's actually ready to serve.
Why other options are not the Kubernetes-native "strategy" answer here:
Recreate (B) explicitly stops old Pods before starting new ones, causing downtime for most services.
Canary (A) and BlueGreen (C) are real deployment patterns, but in "Kubernetes-native deployment strategy" terms, the built-in Deployment strategies are RollingUpdate and Recreate. Canary/BlueGreen typically require additional tooling/controllers (service mesh, ingress controller features, or progressive delivery operators) to manage traffic shifting between versions.
So, for a Kubernetes-native strategy that supports zero-downtime updates, the correct and verified choice is RollingUpdate (D).

NEW QUESTION # 164
.....

- Free PDF Quiz Professional Linux Foundation - KCNA Latest Dumps Files □ Search for □ KCNA □ and download it for free immediately on □ www.practicevce.com □ □KCNA Pass Guide
- Get The UP-To-Date Linux Foundation KCNA Exam Questions □ Enter ✔ www.pdfvce.com □✔□ and search for □ KCNA □ to download for free ❣ Real KCNA Exams
- Pass KCNA Exam with 100% Pass Rate KCNA Latest Dumps Files by www.prep4sures.top ～ Download ☀ KCNA □☀□ for free by simply searching on ➤ www.prep4sures.top □ □Online KCNA Version
- Get Ready for KCNA with Linux Foundation's Realistic Exam Questions and Accurate Answers □ Open ✔ www.pdfvce.com □✔□ and search for ➽ KCNA □ to download exam materials for free □KCNA Examcollection
- Get Ready for KCNA with Linux Foundation's Realistic Exam Questions and Accurate Answers □ Easily obtain （ KCNA ） for free download through ➥ www.examcollectionpass.com □ □Reliable KCNA Exam Test
- hamzashkt840427.activoblog.com, roxannwfdv117043.wikinstructions.com, nicoleaecs645808.blog2freedom.com, zaynabbccu907802.vigilwiki.com, pr1bookmarks.com, siobhanczep539160.blogtov.com, dawudqrst010749.blog2news.com, bouchesocial.com, poppieibrl429916.buscawiki.com, kiarafdmi045467.blog-mall.com, Disposable vapes

P.S. Free 2026 Linux Foundation KCNA dumps are available on Google Drive shared by PassReview: https://drive.google.com/open?id=1D_vG17A3OswoWPM3ifx0ybhQXi6mf0O2