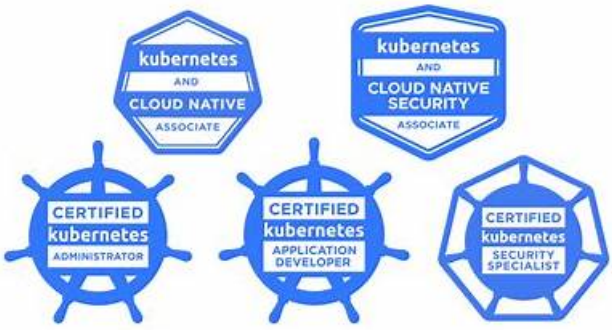
# 100% Pass Quiz Linux Foundation - High Pass-Rate Updated KCSA Test Cram




Dumpleader was established in 2008, now we are the leading position in this field as we have good reputation of high-pass-rate KCSA guide torrent materials. Our KCSA exam questions are followed by many peers many years but never surpassed. We build a mature and complete KCSA learning guide R&D system, customers' information safety system & customer service system since past 10 years. Every candidate who purchases our valid KCSA Preparation materials will enjoy our high-quality guide torrent, information safety and golden customer service.

## Linux Foundation KCSA Exam Syllabus Topics:

| Topic | Details |
|---|---|
| Topic 1 | • Compliance and Security Frameworks: This section of the exam measures the skills of a Compliance Officer and focuses on applying formal structures to ensure security and meet regulatory demands. It covers working with industry-standard compliance and threat modeling frameworks, understanding supply chain security requirements, and utilizing automation tools to maintain and prove an organization's security posture. |
| Topic 2 | • Overview of Cloud Native Security: This section of the exam measures the skills of a Cloud Security Architect and covers the foundational security principles of cloud-native environments. It includes an understanding of the 4Cs security model, the shared responsibility model for cloud infrastructure, common security controls and compliance frameworks, and techniques for isolating resources and securing artifacts like container images and application code. |
| Topic 3 | • Platform Security: This section of the exam measures the skills of a Cloud Security Architect and encompasses broader platform-wide security concerns. This includes securing the software supply chain from image development to deployment, implementing observability and service meshes, managing Public Key Infrastructure (PKI), controlling network connectivity, and using admission controllers to enforce security policies. |
| Topic 4 | • Kubernetes Threat Model: This section of the exam measures the skills of a Cloud Security Architect and involves identifying and mitigating potential threats to a Kubernetes cluster. It requires understanding common attack vectors like privilege escalation, denial of service, malicious code execution, and network-based attacks, as well as strategies to protect sensitive data and prevent an attacker from gaining persistence within the environment. |

**>> Updated KCSA Test Cram <<**

## KCSA Books PDF & KCSA Top Exam Dumps

With the development of society and the perfection of relative laws and regulations, the KCSA certificate in our career field

becomes a necessity for our countryPassing the KCSA and obtaining the certificate may be the fastest and most direct way to change your position and achieve your goal. And we are just right here to give you help. Being considered the most authentic brand in this career, our professional experts are making unremitting efforts to provide our customers the latest and valid KCSA Exam simulation.

## Linux Foundation Kubernetes and Cloud Native Security Associate Sample Questions (Q15-Q20):

**NEW QUESTION # 15**
A container running in a Kubernetes cluster has permission to modify host processes on the underlying node.
What combination of privileges and capabilities is most likely to have led to this privilege escalation?

- A. hostPath and AUDIT_WRITE
- B. hostPID and SYS_PTRACE
- C. hostNetwork and NET_RAW
- D. There is no combination of privileges and capabilities that permits this.

**Answer: B**

Explanation:
* hostPID:When enabled, the container shares the host's process namespace # container can see and potentially interact with host processes.
* SYS_PTRACE capability:Grants the container the ability to trace, inspect, and modify other processes (e.g., via ptrace).
* Combination of hostPID + SYS_PTRACE allows a container toattach to and modify host processes, which is a direct privilege escalation.
* Other options explained:
* hostPath + AUDIT_WRITE:hostPath exposes filesystem paths but does not inherently allow process modification.
* hostNetwork + NET_RAW:grants raw socket access but only for networking, not host process modification.
* A:Incorrect - such combinationsdo exist(like B).
References:
Kubernetes Docs - Configure a Pod to use hostPID: https://kubernetes.io/docs/tasks/configure-pod-container /share-process-namespace/
Linux Capabilities man page: https://man7.org/linux/man-pages/man7/capabilities.7.html

**NEW QUESTION # 16**
How do Kubernetes namespaces impact the application of policies when using Pod Security Admission?

- A. Each namespace can have only one active policy.
- B. Different policies can be applied to specific namespaces.
- C. The default namespace enforces the strictest security policies by default.
- D. Namespaces are ignored; Pod Security Admission policies apply cluster-wide only.

**Answer: B**

Explanation:
* Pod Security Admission (PSA)enforces policies by applyinglabels on namespaces, not globally across the cluster.
* Exact extract (Kubernetes Docs - Pod Security Admission):
* "You can apply Pod Security Standards to namespaces by adding labels such as pod- security.kubernetes.io/enforce. Different namespaces can enforce different policies."
* Clarifications:
* A: Incorrect, namespaces are the unit of enforcement.
* C: Misleading - a namespace can have multiple enforcement modes (enforce, audit, warn).
* D: Default namespace doesnotenforce strict policies unless labeled.
References:
Kubernetes Docs - Pod Security Admission: https://kubernetes.io/docs/concepts/security/pod-security- admission/

**NEW QUESTION # 17**
Which step would give an attacker a foothold in a cluster butno long-term persistence?

- A. Modify file on host filesystem.
- B. Starting a process in a running container.
- C. Modify Kubernetes objects stored within etcd.
- D. Create restarting container on host using Docker.

**Answer: B**

Explanation:
* Starting a process in a running containerprovides an attacker withtemporary execution (foothold) inside the cluster, but once the container is stopped or restarted, that malicious process is lost. This means the attacker has nolong-term persistence.
* Incorrect options:
* (A) Modifying objects inetcdgrants persistent access since cluster state is stored in etcd.
* (B) Modifying files on thehost filesystemcan create persistence across reboots or container restarts.
* (D) Creating a restarting container directly on the host via Docker bypasses Kubernetes but persists across pod restarts if Docker restarts it.
References:
CNCF Security Whitepaper - Threat Modeling section: Describes howephemeral processes inside containersprovide attackers short-term control but not durable persistence.
Kubernetes Documentation - Cluster Threat Model emphasizes ephemeral vs. persistent attacker footholds.

**NEW QUESTION # 18**
A container image istrojanizedby an attacker by compromising the build server. Based on the STRIDE threat modeling framework, which threat category best defines this threat?

- A. Denial of Service
- B. Repudiation
- C. Spoofing
- D. Tampering

**Answer: D**

Explanation:
* In STRIDE,Tamperingis the threat category forunauthorized modification of data or code/artifacts. A trojanized container image is, by definition, an attacker'smodificationof the build output (the image) after compromising the CI/build system-i.e., tampering with the artifact in the software supply chain.
* Why not the others?
* Spoofingis about identity/authentication (e.g., pretending to be someone/something).
* Repudiationis about denying having performed an action without sufficient audit evidence.
* Denial of Servicetargets availability (exhausting resources or making a service unavailable).The scenario explicitly focuses on analtered imageresulting from a compromised build server-this squarely maps toTampering.
Authoritative references (for verification and deeper reading):
* Kubernetes (official docs)- Supply Chain Security (discusses risks such as compromised CI/CD pipelines leading to modified/poisoned images and emphasizes verifying image integrity/signatures).
* Kubernetes Docs#Security#Supply chain securityandSecuring a cluster(sections on image provenance, signing, and verifying artifacts).
* CNCF TAG Security - Cloud Native Security Whitepaper (v2)- Threat modeling in cloud-native and software supply chain risks; describes attackers modifying build outputs (images/artifacts) via CI
/CD compromise as a form oftamperingand prescribes controls (signing, provenance, policy).
* CNCF TAG Security - Software Supply Chain Security Best Practices- Explicitly covers CI/CD compromise leading tomaliciously modified imagesand recommends SLSA, provenance attestation, and signature verification (policy enforcement via admission controls).
* Microsoft STRIDE (canonical reference)- DefinesTamperingasmodifying data or code, which directly fits a trojanized image produced by a compromised build system.

**NEW QUESTION # 19**
An attacker compromises a Pod and attempts to use its service account token to escalate privileges within the cluster. Which Kubernetes security feature is designed tolimit what this service account can do?

- A. NetworkPolicy

- B. RuntimeClass
- C. Role-Based Access Control (RBAC)
- D. PodSecurity admission

**Answer: C**

Explanation:
* When a Pod is created, Kubernetes automatically mounts aservice account tokenthat can authenticate to the API server.
* TheRole-Based Access Control (RBAC)system defines what actions a service account can perform.
* By carefully restricting Roles and RoleBindings, administrators limit the blast radius of a compromised Pod.
* Incorrect options:
* (A)PodSecurity admissionenforces workload-level security settings but does not control API access.
* (B)NetworkPolicycontrols network communication, not API privileges.
* (D)RuntimeClassselects container runtimes, unrelated to privilege escalation through API tokens.
References:
Kubernetes Documentation - Using RBAC Authorization
CNCF Security Whitepaper - Identity & Access Management: limiting lateral movement by constraining service account permissions.

**NEW QUESTION # 20**
.....