Linux-Foundation

Exam Questions KCSA

Kubernetes and Cloud Native Security Associate (KCSA)

NEW QUESTION 1
Which of the following statements best describes the role of the Scheduler in Kubernetes?

A. The Scheduler is responsible for monitoring and managing the health of the Kubernetes cluster.
B. The Scheduler is responsible for ensuring the security of the Kubernetes cluster and its components.
C. The Scheduler is responsible for managing the deployment and scaling of applications in the Kubernetes cluster.
D. The Scheduler is responsible for assigning Pods to nodes based on resource availability and other constraints.

Answer: D

Explanation:

- TheKubernetes Schedulerassigns Pods to nodes based on:
- Resource requests & availability (CPU, memory, GPU, etc.)
- Constraints (affinity, taints, tolerations, topology, policies)
- Exact extract (Kubernetes Docs – Scheduler):
- ??The scheduler is a control plane process that assigns Pods to Nodes. Scheduling decisions take into account resource requirements, affinity/anti-affinity, constraints, and policies.??
- Other options clarified:
- A: Monitoring cluster health is theController Manager??s/kubelet??s job.
- B: Security is enforced throughRBAC, admission controllers, PSP/PSA, not the scheduler.
- C: Deployment scaling is handled by theController Manager(Deployment/ReplicaSet controller).

References:
Kubernetes Docs — Scheduler: https://kubernetes.io/docs/concepts/scheduling-eviction/kube-scheduler/

NEW QUESTION 2
In Kubernetes, what isPublic Key Infrastructure (PKI)used for?

A. To manage certificates and ensure secure communication in a Kubernetes cluster.
B. To automate the scaling of containers in a Kubernetes cluster.
C. To manage networking in a Kubernetes cluster.
D. To monitor and analyze performance metrics of a Kubernetes cluster.

Answer: A

Explanation:
Kubernetes usesPKI certificatesextensively to secure communication between control plane components (API server, etcd, kube-scheduler, kube-controller-manager) and with kubelets.
Certificates enablemutual TLS authentication and encryptionacross components.
PKI does not handle scaling, networking, or monitoring.
References:
Kubernetes Documentation – Certificates
CNCF Security Whitepaper – Cluster communication security and the role of PKI.

NEW QUESTION 3
In a Kubernetes cluster, what are the security risks associated with using ConfigMaps for storing secrets?

A. Storing secrets in ConfigMaps does not allow for fine-grained access control via RBAC.
B. Storing secrets in ConfigMaps can expose sensitive information as they are stored in plaintext and can be accessed by unauthorized users.
C. Using ConfigMaps for storing secrets might make applications incompatible with the Kubernetes cluster.
D. ConfigMaps store sensitive information in etcd encoded in base64 format automatically, which does not ensure confidentiality of data.

Answer: B

Explanation:

- ConfigMaps are explicitly not for confidential data.
- Exact extract (ConfigMap concept):"A ConfigMap is an API object used to store non-confidential data in key-value pairs."
- Exact extract (ConfigMap concept):"ConfigMaps are not intended to hold confidential data. Use a Secret for confidential data."
- Why this is risky:data placed into a ConfigMap is stored as regular (plaintext) string values in the API and etcd (unless you deliberately use binaryData for base64 content you supply). That means if someone has read access to the namespace or to etcd/APIServer storage, they can view the values.
- Secrets vs ConfigMaps (to clarify distractor D):
- Exact extract (Secret concept):"By default, secret data is stored as unencrypted base64-encoded strings.You canenable encryption at restto protect Secrets stored in etcd."
- This base64 behavior applies toSecrets, not to ConfigMap data. Thus optionDis incorrect for ConfigMaps.

About RBAC (to clarify distractor A):Kubernetesdoessupport fine-grained RBAC forbothConfigMaps and Secrets; the issue isn't lack of RBAC but that ConfigMaps arenotdesigned for confidential material.

About compatibility (to clarify distractor C):Using ConfigMaps for secrets doesn't make apps "incompatible"; it's simplyinsecureand against guidance.
[References:, Kubernetes Docs —ConfigMaps: https://kubernetes.io/docs/concepts/configuration/configmap/, Kubernetes Docs —Secrets: https://kubernetes.io/docs/concepts/configuration/secret/, Kubernetes Docs —Encrypting Secret Data at Rest: https://kubernetes.io/docs/tasks/administer-cluster/encrypt-data/, Note: The citations above are from the official Kubernetes documentation and reflect the stated guidance that ConfigMaps are fornon-confidentialdata, while Secrets (with encryption at rest enabled) are forconfidentialdata, and that the 4C's map todefense in depth.,]

NEW QUESTION 4
What is the difference between gVisor and Firecracker?

A. gVisor is a user-space kernel that provides isolation and security for container
B. At the same time, Firecracker is a lightweight virtualization technology for creating and managing secure, multi-tenant container and function-as-a-service (FaaS) workloads.
C. gVisor is a lightweight virtualization technology for creating and managing secure, multi-tenant container and function-as-a-service (FaaS) workload
D. At the same time, Firecracker is a user-space kernel that provides isolation and security for containers.
E. gVisor and Firecracker are both container runtimes that can be used interchangeably.
F. gVisor and Firecracker are two names for the same technology, which provides isolation and security for containers.

Answer: A

Explanation:
gVisor:
Google-developed, implemented as auser-space kernelthat intercepts and emulates syscalls made by containers.
Providesstrong isolationwithout requiring a full VM.
Official docs: "gVisor is a user-space kernel, written in Go, that implements a substantial portion of the Linux system call interface."
Source: https://gvisor.dev/docs/
Firecracker:
AWS-developed,lightweight virtualization technologybuilt on KVM, used in AWS Lambda and Fargate.
Optimized for running secure, multi-tenant microVMs (MicroVMs) for containers and FaaS.
Official docs: "Firecracker is an open-source virtualization technology that is purpose-built for creating and managing secure, multi-tenant container and function-based services."
Source: https://firecracker-microvm.github.io/
Key difference:gVisor ?? syscall interception in userspace kernel (container isolation). Firecracker ?? lightweight virtualization with microVMs (multi-tenant security).
Therefore, optionAis correct.
[References:, gVisor Docs: https://gvisor.dev/docs/, Firecracker Docs: https://firecracker-microvm.github.io/,]

NEW QUESTION 5
Is it possible to restrict permissions so that a controller can only change the image of a deployment (without changing anything else about it, e.g., environment variables, commands, replicas, secrets)?

A. Yes, by granting permission to the /image subresource.
B. Not with RBAC, but it is possible with an admission webhook.
C. No, because granting access to the spec.containers.image field always grants access to the rest of the spec object.
D. Yes, with a 'managed fields' annotation.

Answer: B

Explanation:
RBAC in Kubernetesis coarse-grained: it controlsverbs(get, update, patch, delete) onresources(e.g., deployments), butnot individual fieldswithin a resource.
There isno /image subresource for deployments(there is one for pods but only for ephemeral containers).
Therefore,RBAC cannot restrict changes only to the image field.
Admission Webhooks(mutating/validating)canenforce fine-grained policies (e.g., deny updates that change anything other than spec.containers[*].image).
Exact extract (Kubernetes Docs – Admission Webhooks):
"Admission webhooks can be used to enforce custom policies on objects being admitted."
[References:, Kubernetes Docs — RBAC: https://kubernetes.io/docs/reference/access-authn-authz/rbac/, Kubernetes Docs — Admission Webhooks: https://kubernetes.io/docs/reference/access-authn-authz/extensible-admission-controllers/,]

NEW QUESTION 6
You want to minimize security issues in running Kubernetes Pods. Which of the following actions can help achieve this goal?

A. Sharing sensitive data among Pods in the same cluster to improve collaboration.
B. Running Pods with elevated privileges to maximize their capabilities.
C. Implement Pod Security standards in the Pod's YAML configuration.
D. Deploying Pods with randomly generated names to obfuscate their identities.

Answer: C

Explanation:
Pod Security Standards (PSS):
Kubernetes providesPod Security Admission (PSA)to enforce security controls based on policies.
Official extract: ??Pod Security Standards define different isolation levels for Pods. The standards focus on restricting what Pods can do and what they can access.??

The three standard profiles are:
Privileged: unrestricted (not recommended).
Baseline: minimal restrictions.
Restricted: highly restricted, enforcing least privilege.

Why option C is correct:
Applying Pod Security Standards in YAML ensures Pods adhere tobest practiceslike:
No root user.
Restricted host access.
No privilege escalation.
Seccomp/AppArmor profiles.
This directly minimizes security risks.

Why others are wrong:
A:Sharing sensitive data increases risk of exposure.
B:Running with elevated privileges contradicts least privilege principle.
D:Random Pod names donotcontribute to security.
[References:, Kubernetes Docs — Pod Security Standards: https://kubernetes.io/docs/concepts/security/pod-security-standards/, Kubernetes Docs — Pod Security Admission: https://kubernetes.io/docs/concepts/security/pod-security-admission/,]

NEW QUESTION 7
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. There is no combination of privileges and capabilities that permits this.
B. hostPID and SYS_PTRACE
C. hostPath and AUDIT_WRITE
D. hostNetwork and NET_RAW

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 8
When using a cloud provider's managed Kubernetes service, who is responsible for maintaining the etcd cluster?

A. Kubernetes administrator
B. Namespace administrator
C. Cloud provider
D. Application developer

Answer: C

Explanation:
Inmanaged Kubernetes services(EKS, GKE, AKS), the control plane is operated by thecloud provider.
This includesetcd, API server, controller manager, scheduler.
Users manageworker nodes(in some models) and workloads, but not the control plane.
Exact extract (GKE Docs):
"The control plane, including the API server and etcd database, is managed and maintained by Google."
Similarly forEKSandAKS, etcd is fully managed by the provider.
[References:, GKE Architecture: https://cloud.google.com/kubernetes-engine/docs/concepts/cluster-architecture, EKS Architecture: https://docs.aws.amazon.com/eks/latest/userguide/eks-architecture.html, AKS Docs: https://learn.microsoft.com/en-us/azure/aks/concepts-clusters-workloads,]

NEW QUESTION 9
An attacker has access to the network segment that the cluster is on.
What happens when a compromised Pod attempts to connect to the API server?

A. The compromised Pod is automatically isolated from the network to prevent any connections to the API server.
B. The compromised Pod is allowed to connect to the API server without any restrictions.
C. The compromised Pod attempts to connect to the API server, but its requests may be blocked due to network policies.
D. The compromised Pod connects to the API server and is granted elevated privileges by default.

Answer: C

Explanation:
By default,Pods can connect to the API server(since ServiceAccount tokens are mounted).
However, whether they succeed in acting depends on:
Network Policies(may block egress).

RBAC(controls permissions).
Exact extract (Kubernetes Docs – API Access):
??Pods authenticate to the API server using the service account token mounted into the Pod.
Authorization is then enforced by RBAC. NetworkPolicies may further restrict access.??
Clarifications:
A: No default automatic isolation.
B: Not always unrestricted; policies may apply.
D: Pods get minimal default privileges, not automatic elevation.
References:
Kubernetes Docs — API Access to Pods: https://kubernetes.io/docs/concepts/security/service-accounts/ Kubernetes Docs — Network Policies: https://kubernetes.io/docs/concepts/services-networking/network-policies/

NEW QUESTION 10
Which label should be added to the Namespace to block any privileged Pods from being created in that Namespace?

A. privileged: false
B. privileged: true
C. pod-security.kubernetes.io/enforce: baseline
D. pod.security.kubernetes.io/privileged: false

Answer: C

Explanation:
KubernetesPod Security Admission (PSA)enforcesPod Security Standardsby applying labels on Namespaces.
Exact extract (Kubernetes Docs – Pod Security Admission):
??You can label a namespace with pod-security.kubernetes.io/enforce: baseline to enforce the Baseline policy.??
Thebaselineprofile explicitly disallowsprivileged podsand other unsafe features.
Why others are wrong:
A & D: These labels do not exist in Kubernetes.
B: Setting privileged: true would allow privileged pods, not block them.
References:
Kubernetes Docs — Pod Security Admission: https://kubernetes.io/docs/concepts/security/pod-security- admission/
Kubernetes Docs — Pod Security Standards: https://kubernetes.io/docs/concepts/security/pod-security- standards/

NEW QUESTION 10
What is the reasoning behind considering the Cloud as the trusted computing base of a Kubernetes cluster?

A. The Cloud enforces security controls at the Kubernetes cluster level, so application developers can focus on applications only.
B. A Kubernetes cluster can only be trusted if the underlying Cloud provider is certified against international standards.
C. A vulnerability in the Cloud layer has a negligible impact on containers due to Linux isolation mechanisms.
D. A Kubernetes cluster can only be as secure as the security posture of its Cloud hosting.

Answer: D

Explanation:
The4C??s of Cloud Native Security(Cloud, Cluster, Container, Code) model starts withCloudas the base layer.
If the Cloud (infrastructure layer) is compromised, every higher layer (Cluster, Container, Code) inherits that compromise.
Exact extract (Kubernetes Security Overview):
??The 4C??s of Cloud Native security are Cloud, Clusters, Containers, and Code. You can think of the 4C??s as a layered approach. A Kubernetes cluster can only be as secure as the cloud infrastructure it is deployed on.??
This means the cloud is part of thetrusted computing baseof a Kubernetes cluster.
References:
Kubernetes Docs — Security Overview (4C??s): https://kubernetes.io/docs/concepts/security/overview/#the- 4cs-of-cloud-native-security

NEW QUESTION 13
Why mightNetworkPolicyresources have no effect in a Kubernetes cluster?

A. NetworkPolicy resources are only enforced if the Kubernetes scheduler supports them.
B. NetworkPolicy resources are only enforced if the networking plugin supports them.
C. NetworkPolicy resources are only enforced for unprivileged Pods.
D. NetworkPolicy resources are only enforced if the user has the right RBAC permissions.

Answer: B

Explanation:
NetworkPolicies define how Pods can communicate with each other and external endpoints.
However, Kubernetes itselfdoes not enforce NetworkPolicy. Enforcement depends on theCNI plugin used (e.g., Calico, Cilium, Kube-Router, Weave Net).
If a cluster is using a network plugin that does not support NetworkPolicies, then creating NetworkPolicy objects hasno effect.
References:
Kubernetes Documentation – Network Policies
CNCF Security Whitepaper – Platform security section: notes that security enforcement relies on CNI capabilities.

NEW QUESTION 15
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. PodSecurity admission
B. NetworkPolicy
C. Role-Based Access Control (RBAC)
D. RuntimeClass

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
What kind of organization would need to be compliant with PCI DSS?

A. Retail stores that only accept cash payments.
B. Government agencies that collect personally identifiable information.
C. Non-profit organizations that handle sensitive customer data.
D. Merchants that process credit card payments.

Answer: D

Explanation:
PCI DSS (Payment Card Industry Data Security Standard):applies to any entity thatstores, processes, or transmits cardholder data.
Exact extract (PCI DSS official summary):
"PCI DSS applies to all entities that store, process or transmit cardholder data (CHD) and /or sensitive authentication data (SAD)."
Therefore,merchants who process credit card paymentsmust comply.
Why others are wrong:
A: No card payments, so no PCI scope.
B: This falls underFISMA / NIST 800-53, not PCI DSS.
C: Non-profits may handle sensitive data, but PCI only applies if they processcredit cards.
References:
PCI Security Standards Council — PCI DSS Summary: https://www.pcisecuritystandards.org/pci_security/

NEW QUESTION 22
Which of the following is a valid security risk caused by having no egress controls in a Kubernetes cluster?

A. Denial of Service
B. Data exfiltration
C. Increased attack surface
D. Unauthorized access to external resources

Answer: B

Explanation:
Egress NetworkPoliciesrestrict outbound traffic from Pods.
Without egress restrictions, a compromised Pod could exfiltrate sensitive data (secrets, logs, customer data) to an attacker-controlled server.
Exact extract (Kubernetes Docs – Network Policies):
"Egress rules control outbound connections from Pods. Without such restrictions, compromised workloads can connect freely to external endpoints."
Other options clarified:
A: DoS is more about flooding, not egress absence.
C: ??Increased attack surface?? is vague but not the main risk.
D: True in a sense, but the precise and most common risk isdata exfiltration.
[References:, Kubernetes Docs — Network Policies: https://kubernetes.io/docs/concepts/services-networking/network-policies/,]

NEW QUESTION 24
What is the purpose of an egress NetworkPolicy?

A. To control the incoming network traffic to a Kubernetes cluster.
B. To control the outbound network traffic from a Kubernetes cluster.
C. To secure the Kubernetes cluster against unauthorized access.
D. To control the outgoing network traffic from one or more Kubernetes Pods.

Answer: D

Explanation:
NetworkPolicycontrols network trafficat the Pod level.
Ingress rules:controlincomingconnections to Pods.
Egress rules:controloutgoingconnectionsfrom Pods.
Exact extract (Kubernetes Docs – Network Policies):
"An egress rule controls outgoing connections from Pods that match the policy."
Clarifying wrong answers:

A/B: Too broad (cluster-level); policies apply per Pod/Namespace.
C: Security against unauthorized access is broader than egress policies.
[References:, Kubernetes Docs — Network Policies: https://kubernetes.io/docs/concepts/services-networking/network-policies/,]

NEW QUESTION 27
You are responsible for securing thekubeletcomponent in a Kubernetes cluster.
Which of the following statements about kubelet security is correct?

A. Kubelet runs as a privileged container by default.
B. Kubelet does not have any built-in security features.
C. Kubelet supports TLS authentication and encryption for secure communication with the API server.
D. Kubelet requires root access to interact with the host system.

Answer: C

Explanation:
Thekubeletis the primary agent that runs on each node in a Kubernetes cluster and communicates with the control plane.
Kubeletsupports TLS (Transport Layer Security)for both authentication and encryption when interacting with the API server. This is a core security feature that ensures secure node-to-control-plane communication.
Incorrect options:
(A) Kubelet does not run as a privileged container by default; it runs as a system process (typically systemd-managed) on the host.
(B) Kubelet does include built-in security features such asTLS authentication, authorization modes, and read-only vs secured ports.
(D) While kubelet interacts with the host system (e.g., cgroups, container runtimes), it does not inherently require root access for communication security; RBAC and TLS handle authentication.
[References:, Kubernetes Documentation – Kubelet authentication/authorization, CNCF Security Whitepaper – Cluster Component Security (discusses TLS and mutual authentication between kubelet and API server).,]

NEW QUESTION 29
Which of the following snippets from a RoleBinding correctly associates user bob with Role pod-reader ?

A. subjects:- kind: Username: pod-readerapiGroup: rbac.authorization.k8s.ioroleRef:kind: Rolename: bobapiGroup: rbac.authorization.k8s.io
B. subjects:- kind: Username: bobapiGroup: rbac.authorization.k8s.ioroleRef:kind: Rolename: pod-readerapiGroup: rbac.authorization.k8s.io
C. subjects:- kind: Username: bobapiGroup: rbac.authorization.k8s.ioroleRef:kind: ClusterRolename: pod-readerapiGroup: rbac.authorization.k8s.io
D. subjects:- kind: Groupname: bobapiGroup: rbac.authorization.k8s.ioroleRef:kind: Rolename: pod-readerapiGroup: rbac.authorization.k8s.io

Answer: B

Explanation:
Kubernetes RBAC usesRoleBindingto grant permissions defined in aRoleto asubject(user, group, or service account) within a namespace. The official example shows binding user jane to Role pod-reader:
"A RoleBinding grants the permissions defined in a Role to a user or set of users??."
Example:
subjects:
- kind: User
name: jane
apiGroup: rbac.authorization.k8s.io
roleRef:
kind: Role
name: pod-reader
apiGroup: rbac.authorization.k8s.io
— Kubernetes docs, RBAC: RoleBinding and ClusterRoleBinding
OptionBmatches this pattern exactly, with name: bob as theUsersubject and roleRef pointing to theRolenamed pod-reader.
Aswaps the names (subject is pod-reader, role is bob) ?? incorrect.
Creferences aClusterRole, not aRole(the question asks for Role).
Duses kind: Group even though we need theUserbob.
[References:, Kubernetes Docs — Using RBAC Authorization ??RoleBinding and ClusterRoleBinding: https://kubernetes.io/docs/reference/access-authn-authz/rbac/#rolebinding-and-clusterrolebinding,]

NEW QUESTION 33
In order to reduce the attack surface of the Scheduler, which default parameter should be set to false?

A. --scheduler-name
B. --profiling
C. --secure-kubeconfig
D. --bind-address

Answer: B

Explanation:
Thekube-schedulerexposes aprofiling/debugging endpointwhen --profiling=true (default).
This can unnecessarily increase the attack surface.
Best practice: set --profiling=false in production.
Exact extract (Kubernetes Docs – kube-scheduler flags):
"--profiling (default true): Enable profiling via web interface host:port/debug/pprof/."
Why others are wrong:
--scheduler-name: just identifies the scheduler, not a security risk.
--secure-kubeconfig: not a valid flag.
--bind-address: changing it limits exposure but is not the default risk parameter for profiling.
[References:, Kubernetes Docs — kube-scheduler options: https://kubernetes.io/docs/reference/command-line-tools-reference/kube-scheduler/, ,]

NEW QUESTION 36
Which step would give an attacker a foothold in a cluster butno long-term persistence?

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

Answer: C

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 41
.....

Thank You for Trying Our Product

We offer two products:

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

2nd - Questons and Answers in PDF Format

KCSA Practice Exam Features:

* KCSA Questions and Answers Updated Frequently

* KCSA Practice Questions Verified by Expert Senior Certified Staff

* KCSA Most Realistic Questions that Guarantee you a Pass on Your FirstTry

* KCSA Practice Test Questions in Multiple Choice Formats and Updatesfor 1 Year