# Exam Questions KCSA

Kubernetes and Cloud Native Security Associate (KCSA)

**https://www.2passeasy.com/dumps/KCSA/**

**NEW QUESTION 1**
Which standard approach to security is augmented by the 4C??s of Cloud Native security?

A. Zero Trust
B. Least Privilege
C. Defense-in-Depth
D. Secure-by-Design

**Answer:** C

**Explanation:**

The 4C??s model (Cloud, Cluster, Container, Code) is presented in the official Kubernetes documentation as alayeredmodel that explicitly maps todefense-in-depth.

Exact extracts from Kubernetes docs(security overview):

??The 4C??s of Cloud Native Security are Cloud, Clusters, Containers, and Code.??

??You can think of the 4C??s asa layered approach to security; applying security measures at each layer reduces risk.??

??This layered approach is commonly known asdefense in depth.??

References:
Kubernetes Docs — Security overview #The 4C??s of Cloud Native Security: https://kubernetes.io/docs/concepts/security/overview/#the-4cs-of-cloud-native-security

**NEW QUESTION 2**
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 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**
A cluster administrator wants to enforce the use of a different container runtime depending on the application a workload belongs to.

A. By manually modifying the container runtime for each workload after it has been created.
B. By modifying the kube-apiserver configuration file to specify the desired container runtime for each application.
C. By configuring avalidating admission controllerwebhook that verifies the container runtime based on the application label and rejects requests that do not comply.
D. By configuring amutating admission controllerwebhook that intercepts new workload creation requests and modifies the container runtime based on the application label.

**Answer:** D

**Explanation:**

Kubernetes supports workload-specific runtimes viaRuntimeClass.

Amutating admission controllercan enforce this automatically by:

Intercepting workload creation requests.

Modifying the Pod spec to set runtimeClassName based on labels or policies.

Incorrect options:
(A) Manual modification is not scalable or secure.
(B) kube-apiserver cannot enforce per-application runtime policies.
(C) A validating webhook can onlyreject, not modify, the runtime.
[References:, Kubernetes Documentation – RuntimeClass, CNCF Security Whitepaper – Admission controllers for enforcing runtime policies., ]

**NEW QUESTION 5**
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 6**
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 7**
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 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**
Which of the following statements correctly describes a container breakout?

A. A container breakout is the process of escaping the container and gaining access to the Pod's network traffic.
B. A container breakout is the process of escaping a container when it reaches its resource limits.

C. A container breakout is the process of escaping the container and gaining access to the cloud provider's infrastructure.
D. A container breakout is the process of escaping the container and gaining access to the host operating system.

**Answer:** D

**Explanation:**
Container breakoutrefers to an attacker escaping container isolation and reaching thehost OS.
Once the host is compromised, the attacker can accessother containers, Kubernetes nodes, or escalate further.
Exact extract (Kubernetes Security Docs):
??If an attacker gains access to a container, they may attempt a container breakout to gain access to the host system.??
Other options clarified:
A: Network access inside a Pod # breakout.
B: Resource exhaustion is aDoS, not a breakout.
C: Cloud infrastructure compromise is possibleafterhost compromise, but not the definition of breakout.
References:
Kubernetes Security Concepts: https://kubernetes.io/docs/concepts/security/
CNCF Security Whitepaper (Threats section):https://github.com/cncf/tag-security

**NEW QUESTION 10**
A Kubernetes cluster tenant can launch privileged Pods in contravention of therestricted Pod Security Standardmandated for cluster tenants and enforced by the built-inPodSecurity admission controller.
The tenant has full CRUD permissions on the namespace object and the namespaced resources. How did the tenant achieve this?

A. The scope of the tenant role means privilege escalation is impossible.
B. By tampering with the namespace labels.
C. By deleting the PodSecurity admission controller deployment running in their namespace.
D. By using higher-level access credentials obtained reading secrets from another namespace.

**Answer:** B

**Explanation:**
ThePodSecurity admission controllerenforces Pod Security Standards (Baseline, Restricted, Privileged)based on namespace labels.
If a tenant has full CRUD on the namespace object, they canmodify the namespace labelsto remove or weaken the restriction (e.g., setting pod-security.kubernetes.io/enforce=privileged).
This allows privileged Pods to be admitted despite the security policy.
Incorrect options:
(A) is false — namespace-level access allows tampering.
(C) is invalid — PodSecurity admission is not namespace-deployed, it??s a cluster-wide admission controller.
(D) is unrelated — Secrets from other namespaces wouldn??t directly bypass PodSecurity enforcement.
References:
Kubernetes Documentation – Pod Security Admission
CNCF Security Whitepaper – Admission control and namespace-level policy enforcement weaknesses.

**NEW QUESTION 10**
What mechanism can I use to block unsigned images from running in my cluster?

A. Enabling Admission Controllers to validate image signatures.
B. Using PodSecurityPolicy (PSP) to enforce image signing and validation.
C. Using Pod Security Standards (PSS) to enforce validation of signatures.
D. Configuring Container Runtime Interface (CRI) to enforce image signing and validation.

**Answer:** A

**Explanation:**
KubernetesAdmission Controllers(particularlyValidatingAdmissionWebhooks) can be used to enforce policies that validate image signatures.
This is commonly implemented withtools like Sigstore/cosign, Kyverno, or OPA Gatekeeper.
PodSecurityPolicy (PSP):deprecated and never supported image signature validation.
Pod Security Standards (PSS):only apply to pod security fields (privilege, users, host access), not image signatures.
CRI:while runtimes (containerd, CRI-O) may integrate with signature verification tools, enforcement in Kubernetes is generally done viaAdmission Controllersat the API layer.
Exact extract (Admission Controllers docs):
??Admission webhooks can be used to enforce custom policies on the objects being admitted.?? (e.g., validating signatures).
References:
Kubernetes Docs — Admission Controllers: https://kubernetes.io/docs/reference/access-authn-authz
/admission-controllers/
Sigstore Project (cosign): https://sigstore.dev/
Kyverno ImageVerify Policy: https://kyverno.io/policies/pod-security/require-image-verification/

**NEW QUESTION 11**
When should soft multitenancy be used over hard multitenancy?

A. When the priority is enabling resource sharing and efficiency between tenants.
B. When the priority is enabling complete isolation between tenants.
C. When the priority is enabling fine-grained control over tenant resources.
D. When the priority is enabling strict security boundaries between tenants.

**Answer:** A

**Explanation:**
Soft multitenancy(Namespaces, RBAC, Network Policies) # assumes some level of trust between tenants, focuses onresource sharing and efficiency.
Hard multitenancy(separate clusters or strong virtualization) # strict isolation, used when tenants are untrusted.
Exact extract (CNCF TAG Security Multi-Tenancy Whitepaper):
??Soft multi-tenancy refers to multiple workloads running in the same cluster with some trust assumptions. It provides resource sharing and operational efficiency.
Hard multi- tenancy requires stronger isolation guarantees, typically separate clusters.??
References:
CNCF Security TAG — Multi-Tenancy Whitepaper:https://github.com/cncf/tag-security/tree/main/multi- tenancy

**NEW QUESTION 12**
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 17**
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 22**
Which security knowledge-base focuses specifically onoffensive tools, techniques, and procedures?

A. MITRE ATT&CK
B. OWASP Top 10
C. CIS Controls
D. NIST Cybersecurity Framework

**Answer:** A

**Explanation:**
MITRE ATT&CKis a globally recognizedknowledge base of adversary tactics, techniques, and procedures (TTPs). It is focused on describingoffensive behaviorsattackers use.
Incorrect options:
(B)OWASP Top 10highlights common application vulnerabilities, not attacker techniques.
(C)CIS Controlsare defensive best practices, not offensive tools.
(D)NIST Cybersecurity Frameworkprovides a risk-based defensive framework, not adversary TTPs.
References:
MITRE ATT&CK Framework
CNCF Security Whitepaper – Threat intelligence section: references MITRE ATT&CK for describing attacker behavior.

**NEW QUESTION 27**
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 28**
To restrict the kubelet's rights to the Kubernetes API, whatauthorization modeshould be set on the Kubernetes API server?

A. Node
B. AlwaysAllow
C. kubelet
D. Webhook

**Answer:** A

**Explanation:**
TheNode authorization modeis designed to specifically limit what kubelets can do when they connect to the Kubernetes API server.
It authorizes requests from kubelets based on the Pods scheduled to run on their nodes, ensuring kubelets cannot interact with resources beyond their scope.
Incorrect options:
(B)AlwaysAllowallows unrestricted access (insecure).
(C) No kubelet authorization mode exists.
(D)Webhookmode delegates authorization decisions to an external service, not specifically for kubelets.
[References:, Kubernetes Documentation – Node Authorization, CNCF Security Whitepaper – Access control: kubelet authorization and Node authorizer., ]

**NEW QUESTION 31**
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 34**
What is the purpose of the Supplier Assessments and Reviews control in the NIST 800-53 Rev. 5 set of
controls for Supply Chain Risk Management?

A. To evaluate and monitor existing suppliers for adherence to security requirements.
B. To conduct regular audits of suppliers' financial performance.
C. To establish contractual agreements with suppliers.
D. To identify potential suppliers for the organization.

**Answer:** A

**Explanation:**

In NIST SP 800-53 Rev. 5,SR-6: Supplier Assessments and Reviewsrequires evaluating and monitoring suppliers' security and risk practices.
Exact extract (NIST SP 800-53 Rev. 5, SR-6):
"The organization assesses and monitors suppliers to ensure they are meeting the security requirements specified in contracts and agreements."
This is aboutongoing monitoringof supplier adherence, not financial audits, not contract creation, and
not supplier discovery.
References:
NIST SP 800-53 Rev. 5, Control SR-6 (Supplier Assessments and Reviews): https://csrc.nist.gov/publications/detail/sp/800-53/rev-5/final

**NEW QUESTION 36**
What is the main reason an organization would use a Cloud Workload Protection Platform (CWPP) solution?

A. To protect containerized workloads from known vulnerabilities and malware threats.
B. To automate the deployment and management of containerized workloads.
C. To manage networking between containerized workloads in the Kubernetes cluster.
D. To optimize resource utilization and scalability of containerized workloads.

**Answer:** A

**Explanation:**
CWPP (Cloud Workload Protection Platform):As defined by Gartner and adopted across cloud
security practices, CWPPs are designed tosecure workloads(VMs, containers, serverless functions) in
hybrid and cloud environments.
They providevulnerability scanning, runtime protection, compliance checks, and malware detection.
Exact extract (Gartner CWPP definition):??Cloud workload protection platforms protect workloads regardless of location, including physical machines, VMs, containers, and serverless workloads. They provide vulnerability management, system integrity protection, intrusion detection and prevention, and malware protection.??
References:
Gartner: Cloud Workload Protection Platforms Market Guide (summary): https://www.gartner.com/reviews
/market/cloud-workload-protection-platforms
CNCF Security Whitepaper:https://github.com/cncf/tag-security

**NEW QUESTION 41**
.....

# THANKS FOR TRYING THE DEMO OF OUR PRODUCT

Visit Our Site to Purchase the Full Set of Actual KCSA Exam Questions With Answers.

We Also Provide Practice Exam Software That Simulates Real Exam Environment And Has Many Self-Assessment Features. Order the KCSA Product From:

**https://www.2passeasy.com/dumps/KCSA/**

## Money Back Guarantee

## 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