email : techcenter@trainocate.com.my
contact : +603-2281-1111

CKS: Certified Kubernetes Security Specialist

Course Code: CKS
Duration: 4 days
Instructor-led Training (ILT) | Virtual Instructor-led Training (VILT)

OVERVIEW

The CKS was created by the Linux Foundation and the Cloud Native Computing Foundation (CNCF) as a part of their ongoing effort to help develop the Kubernetes ecosystem. The exam is online, proctored, performance-based test that requires solving multiple tasks from a command line running Kubernetes.

Once enrolled you will receive access to an exam simulator, provided by **Killer.sh**, allowing you to experience the exam environment. You will have two simulation attempts (36 hours of access for each attempt from the start of activation). The simulation includes 20-25 questions that are exactly the same for every attempt and every user, unlike the actual exam. The simulation will provide graded results.

SKILLS COVERED

- Proves High-Demand Security Skills
- Career Advancement
- Industry-wide Credential Recognition
- Networking Opportunities

WHO SHOULD ATTEND?

- DevOps engineers
- Kubernetes administrators
- Cloud security engineers
- IT professionals

PREREQUISITES

- Certified Kubernetes Security Specialist (CKS) candidates must have taken and passed the Certified Kubernetes Administrator (CKA) exam prior to attempting the CKS exam.

MODULES

Module 1: Cluster Setup

- Use Network security policies to restrict cluster level access
- Use CIS benchmark to review the security configuration of Kubernetes components (etcd, kubelet, kubedns, kubeapi)
- Properly set up Ingress with TLS
- Protect node metadata and endpoints
- Verify platform binaries before deploying

Module 2: Cluster Hardening

- Use Role Based Access Controls to minimize exposure
- Exercise caution in using service accounts e.g. disable defaults, minimize permissions on newly created ones
- Restrict access to Kubernetes API
- Upgrade Kubernetes to avoid vulnerabilities

Module 3: System Hardening

- Minimize host OS footprint (reduce attack surface)
- Using least-privilege identity and access management
- Minimize external access to the network
- Appropriately use kernel hardening tools such as AppArmor, seccomp

Module 4: Minimize Microservice Vulnerabilities

- Use appropriate pod security standards
- Manage Kubernetes secrets
- Understand and implement isolation techniques (multi-tenancy, sandboxed containers, etc.)
- Implement Pod-to-Pod encryption using Cilium

Module 5: Supply Chain Security

- Minimize base image footprint
- Understand your supply chain (e.g. SBOM, CI/CD, artifact repositories)
- Secure your supply chain (permitted registries, sign and validate artifacts, etc.)
- Perform static analysis of user workloads and container images (e.g. Kubesec, KubeLinter)

Module 6: Monitoring, Logging and Runtime Security

- Perform behavioral analytics to detect malicious activities
- Detect threats within physical infrastructure, apps, networks, data, users and workloads
- Investigate and identify phases of attack and bad actors within the environment
- Ensure immutability of containers at runtime
- Use Kubernetes audit logs to monitor access

END OF PAGE