Kubernetes Security Best Practices:
Safeguarding Cloud-native Containerized Applications

Course Outline

Overview

The course is a comprehensive program designed to provide participants with the necessary knowledge and practical skills to secure Kubernetes environments effectively. The course covers a range of essential topics, including tracking critical vulnerabilities and staying up to date with Common Vulnerabilities and Exposures (CVE) updates. Participants will learn how to implement security measures following the widely accepted Center for Internet Security (CIS) Benchmarks for Kubernetes, cloud services, and containers.

Through hands-on exercises, participants will gain expertise in conducting security configuration assessments, identifying misconfigurations, and generating comprehensive reports for improved risk management. They will also explore the deployment of Open Policy Agent (OPA) to enforce security policies and govern access control within Kubernetes clusters.

The course also covers configuring Ingress Controllers with Transport Layer Security (TLS) certificates, image analysis for security vulnerabilities, container security scanning, monitoring cluster activities, network interfaces, and implementing host-based intrusion detection.

By the end of the training, participants will have a solid foundation in Kubernetes security best practices and be equipped to implement critical security measures to protect Kubernetes environments, cloud services, and containers in today's increasingly challenging threat landscape.

Prerequisites

Participants should have a solid understanding of Kubernetes fundamentals and experience working with containerization technologies. Familiarity with concepts such as pods, deployments, services, and namespaces is essential. Prior experience in managing and deploying applications on Kubernetes clusters is highly recommended.

Duration

3 days

Course Outline

1. Tracking Vulnerabilities and CVE Updates
   - Stay updated on critical vulnerabilities and Common Vulnerabilities and Exposures (CVE) updates.
   - Understand the impact and mitigation strategies for these vulnerabilities.

2. Strengthening Security Defenses for Kubernetes, Cloud Services, and Containers
   - Implement security measures based on the Center for Internet Security (CIS) Benchmarks to enhance the security posture.
   - Identify and address potential weaknesses within the Kubernetes environment.

Kubernetes Security Best Practices:
Safeguarding Cloud-native Containerized Applications

Course Outline

3. Security Configuration Assessment and Reporting
   - Perform security configuration assessment to identify any misconfigurations or vulnerabilities.
   - Generate comprehensive reports to provide visibility into the security status of the environment.

4. Deployment of Open Policy Agent (OPA)
   - Deploy Open Policy Agent to enforce security policies and govern access control within the Kubernetes cluster.

5. Enhancing Application Security through Lightweight Virtualization Techniques
   - Explore techniques like gVisor and Kata Containers to improve the security and efficiency of containerized applications.

6. Securing the kube-apiserver
   - Implement Role-Based Access Control (RBAC) mechanisms to control access to the kube-apiserver.
   - Enable API server auditing to monitor and track activities for better visibility.

7. Encrypting Secrets
   - Configure the API server to encrypt secrets, ensuring sensitive information is protected.

8. Implementing Network Policies
   - Secure cluster communication by defining and enforcing network policies.

9. Configuring Ingress Controller with TLS Certificates
   - Set up an Ingress Controller that utilizes Transport Layer Security (TLS) certificates to secure incoming traffic.

10. Image Analysis for Security:
    - Perform security checks on container images to identify and mitigate potential security vulnerabilities.

11. Container Security Scanning:
    - Conduct regular security scans on running containers to detect and address any security issues.

12. Tracking and Monitoring Cluster Activity:
    - Use monitoring tools to track and monitor the activity within the Kubernetes cluster.
    - Monitor network interfaces and control host-based intrusion detection.

Kubernetes Security Best Practices:
Safeguarding Cloud-native Containerized Applications

Course Outline

Exam Details

Certified Kubernetes Security Specialist (CKS)

The Certified Kubernetes Security Specialist (CKS) program provides assurance that a CKS has the skills, knowledge, and competence on a broad range of best practices for securing container-based applications and Kubernetes platforms during build, deployment and runtime. CKA certification is required to sit for this exam.

EXAM SIMULATOR: Learners will now have access to an exam simulator, provided by Killer.sh, to experience the exam environment. You will have two exam simulation attempts (36 hours of access for each attempt from the start of activation). Simulation includes 20-25 questions (which are exactly the same for every attempt and every user (unlike those found on the actual exams) and graded simulation results.

This exam is an online, proctored, performance-based test that requires solving multiple tasks from a command line running Kubernetes. Candidates have 2 hours to complete the tasks.

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

The CKS exam environment will be aligned with the most recent K8s minor version within approximately 4 to 8 weeks of the K8s release date.

The cost is USD395 which includes:

- Online Exam Delivery
- Duration of Exam 2 Hours
- Certification Valid 2 Years
- 12 Month Exam Eligibility
- One Retake
- PDF Certificate and Digital Badge
- Performance-Based Exam
- Exam Simulator

Prerequisites

Active (non-expired) CKA certification is a prerequisite for this exam.

Exam Domains & Competencies

Domain	***Weight***
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)*	10%

Kubernetes Security Best Practices:
Safeguarding Cloud-native Containerized Applications

Course Outline

• *Properly set up Ingress objects with security control* • *Protect node metadata and endpoints* • *Minimize use of, and access to, GUI elements* • *Verify platform binaries before deploying*	
Cluster Hardening • *Restrict access to Kubernetes API* • *Use Role Based Access Controls to minimize exposure* • *Exercise caution in using service accounts e.g. disable defaults, minimize permissions on newly created ones* • *Update Kubernetes frequently*	15%
System Hardening • *Minimize host OS footprint (reduce attack surface)* • *Minimize IAM roles* • *Minimize external access to the network* • *Appropriately use kernel hardening tools such as AppArmor, seccomp*	15%
Minimize Microservice Vulnerabilities • *Setup appropriate OS level security domains* • *Manage Kubernetes secrets* • *Use container runtime sandboxes in multi-tenant environments (e.g. gvisor, kata containers)* • *Implement pod to pod encryption by use of mTLS*	20%
Supply Chain Security • *Minimize base image footprint* • *Secure your supply chain: whitelist allowed registries, sign and validate images* • *Use static analysis of user workloads (e.g.Kubernetes resources, Docker files)* • *Scan images for known vulnerabilities*	20%
Monitoring, Logging and Runtime Security • *Perform behavioral analytics of syscall process and file activities at the host and container level to detect malicious activities* • *Detect threats within physical infrastructure, apps, networks, data, users and workloads* • *Detect all phases of attack regardless where it occurs and how it spreads* • *Perform deep analytical investigation and identification of bad actors within environment* • *Ensure immutability of containers at runtime* • *Use Audit Logs to monitor access*	20%