Certified Kubernetes Security Specialist

Duration: 4days (8hrs/day)

Prerequisites: Candidates must have taken and passed the Certified Kubernetes Administrator (CKA) exam prior to attempting the CKS exam.

Course Objective: These objectives focus on fortifying Kubernetes clusters by emphasizing security measures, from initial setup and hardening to supply chain security, monitoring, and threat detection.

Kubernetes Version: 1.22.x

Lab Requirement: Koenig DC (Ubuntu)

Module 1 - Cluster Setup and Hardening

Kubernetes Architecture

Lab: Creating a cluster with kubeadm

Use CIS Benchmark to Review the Security Configuration of Kubernetes Components

Lab: Perform Security Benchmark checks using CIS-CAT Lite and Kube-Bench Tool

Pod to Pod Communication

Public Key Infrastructure (PKI) – Certificate Authority (CA)

Lab: Find Certificates

Lab: Implementing Network Policies on Pods

Minimize Use of, and Access to, GUI Elements

Lab: Install Kubernetes Dashboard

Lab: Verify Platform Binaries - Theory and Hashes

Exercise Caution in Using Service Accounts e.g., Disable Defaults, Minimize Permissions on Newly Created Ones

Lab: Create User and assign RBAC (Role Based Access Control)

Lab: Disable Automount Service Account Token and Anonymous Access

Lab: Node Restriction Admission Controller

Lab: Update Kubernetes Frequently

Module 2 - Minimize Microservice Vulnerabilities

Lab: Managing Secrets

Lab: Encrypt Secrets in ETCD

Setup Appropriate OS Level Security Domains e.g. Using PSP, OPA, Security Contexts

Lab: Implementing Security Context in Pods and Containers

Lab: Creating privileged containers using security context

Lab: Disable Privilege Escalation

Pod Security Policy

Container Runtime Sandboxes

Open Container Initiative

Kata Containers - Sandbox

Lab: Contact the Linux Kernel of worker node From Inside a Container

Lab: Implementing Gvisor on pods

Lab: Custom Security Policies using OPA Gatekeeper

Module 3 - Supply Chain Security

Minimize Base Image Footprint Use Static Analysis of User Workloads (e.g. Kubernetes Resources, Docker Files)

Lab: Static Analysis with Kubesec

Lab: Static Analysis with OPA Conftest

Lab: Checking Image Vulnerabilities with Trivy

Secure Supply Chain

Lab: Whitelist Some Registering Using OPA

ImagePolicyWebhook

Module 4 - Monitoring, Logging and Runtime Security

Perform Behavioral Analytics of Syscall Process and File Activities at the Host and Container Level to Detect Malicious Activities

Kernel vs User Space

Lab: Using Strace command to trace Syscall

Falco

Immutability of Containers at Runtime

Lab: Implementing Immutability on Containers

Lab: Enforce Read-Only Root File system

Use Audit Logs to Monitor Access

Lab: Configure API Server To Store Audit Logs

Lab: Restrict Amount of Audit Logs to Collect

Module 5 - System Hardening

Kernel Hardening Tools

Linux Kernel Isolation

Lab: AppArmor

Lab: Kubernetes with AppArmor

Lab: Seccomp with Kubernetes

Minimize OS Footprint - Reduce Attack Surface

Lab: Reduce Attack Surface