VI215 Container 4 - Kubernetes Security

Kurzbeschreibung:

Dieser Kurs **VI215 Container 4 - Kubernetes Security** vermittelt Kenntnisse und Fähigkeiten, die für die Aufrechterhaltung der Sicherheit in einer hochkomplexen und dynamischen Kubernetes-Umgebung erforderlich sind. Dieser Kurs befasst sich mit Sicherheitsbelangen für **Cloud-Produktionsumgebungen** und deckt Themen im Zusammenhang mit der Sicherheits-Container-Lieferkette ab.
Es werden Themen behandelt, die vor der Konfiguration eines Clusters, während der Bereitstellung und im laufenden Betrieb sowie bei der agilen Nutzung auftreten, einschließlich der Frage, wo Sie aktuelle Informationen zu Sicherheit und Schwachstellen finden.
Der Kurs umfasst praktische Übungen zum Aufbau und zur **Sicherung eines Kubernetes-Clusters** sowie zur Überwachung und Protokollierung von Sicherheitsereignissen.

Zielgruppe:

Das Training **Kubernetes Security - Einbruch schwer gemacht** ist ideal geeignet für:

- DevOps und DevSecOps
- Linux Administratoren

Voraussetzungen:

Um Kursinhalten und Lerntempo des Workshops **VI215 Container 4 - Kubernetes Security** gut folgen zu können, sind gute Linux-Kenntnisse nötig.
Alternativ empfehlen wir Ihnen vorab folgendes Training:

- VI213 Container 2 - Kubernetes Basics

Sonstiges:

Dauer: 3 Tage

Preis: 2250 Euro plus Mwst.

Ziele:

Dieser Kurs vermittelt Kenntnisse und Fähigkeiten, die für die Aufrechterhaltung der Sicherheit in einer hochkomplexen und dynamischen Kubernetes-Umgebung erforderlich sind.

Inhalte/Agenda:

- **Kubernetes Architektur**
  - ◊ Komponenten
  - ◊ Angriffsvektoren
  - ◊
- **Kubernetes Cluster Installation**
  - ◊ Zeig mir deine Zertifikate
  - ◊ Static Pods
  - ◊
- **Control Plane-Security**
  - ◊ Ports und Firewalling
  - ◊ kubelet Absicherung
    - ◊ · TLS
      · RBAC
      · ServiceAccounts
      · Key-Rotation
  - ◊ etcd Absicherung
    - ◊ · Separierung
      · Redundanz/Clustering
      ·
- **Node-Security** ·
  - ◊ Absicherung Betriebssystem (z.B. AppArmor)
  - ◊ Auswahl der richtigen Distribution (minimal Host-OS)
  - ◊ Patching
  - ◊
- **Cluster-Security**
  - ◊ Networking (CNI)
  - ◊ Network Policies
  - ◊ Secret Handling
  - ◊ Projekte und Namespaces
  - ◊ RBAC
  - ◊ Rollout eines neuen Releases
  - ◊
- **Container-Security**
  - ◊ Shift-left Security
  - ◊ CI/CD
  - ◊ Auswahl Base Image (z.B. Distroless)
  - ◊ Bauen eines minimalen Images (Multi-Staging)
  - ◊ Image-Scanning (z.B. aquascan trivy)
  - ◊
- **Workload-Security**
  - ◊ Pod Security Admission
  - ◊ Policy Management (z.B. Kyverno)
  - ◊
- **Audit, Monitoring und Observability**
  - ◊ Audit Policy Logs
  - ◊ Sicherheits Logging
  - ◊ Cilium - Hubble
  - ◊ Thread Detection (z.B. Falco)
  - ◊ Compliance (z.B. kube-bench)
  - ◊
- **Backup und Restore**
  - ◊ etcd
  - ◊ Cluster-Backup (z.B. Kasten)
  - ◊