

## ***LI560 Kubernetes Security***

### **Kurzbeschreibung:**

Der Kurs **LI560 Kubernetes Security** vermittelt Kenntnisse und Fähigkeiten, die für die Aufrechterhaltung der Sicherheit in einer hochkomplexen und dynamischen Kubernetes-Umgebung erforderlich sind. Dieses Kubernetes Security Training 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. Diese Kubernetes Schulung umfasst praktische Übungen zum Aufbau und zur Sicherung eines Kubernetes-Clusters sowie zur Überwachung und Protokollierung von Sicherheitsereignissen.

### **Zielgruppe:**

Das Training **LI560 Kubernetes Security** ist ideal geeignet für:

- DevOps und DevSecOps
- Linux Administratoren

### **Voraussetzungen:**

Um Kursinhalten und Lerntempo des Workshops **LI560 Kubernetes Security** gut folgen zu können, sind zwingend gute Linux- und Kubernetes-Kenntnisse nötig.

Wir empfehlen Ihnen den vorherigen Besuch der folgenden Kurse:

- [LI500 Container Grundlagen](#)
- [LI510 Kubernetes Basics](#)
- [LI530 Kubernetes Advanced](#)

### **Sonstiges:**

**Dauer:** 4 Tage

**Preis:** 2950 Euro plus Mwst.

### **Ziele:**

Der Kurs **LI560 Kubernetes Security** 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
- **◆ 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
  - ◆ ◇ Service Mesh
  - ◆ ◇ Secret Handling (Externes Secret Management z.B. Hashicorp Vault)
  - ◆ ◇ Role Based Access (RBAC) Least Privileged
- **◆ 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
  - ◆ ◇ Konfiguration und Aufbau von Admission Controllern (z.B. Kyverno)
- **◆ Audit, Monitoring und Observability**
  - ◆ ◇ Audit Policy Logs
  - ◆ ◇ EBPF Tooling zum Loggen (z.B. Cilium Hubble)
  - ◆ ◇ Thread Detection (z.B. Falco)
  - ◆ ◇ Compliance (z.B. kube-bench)
- **◆ Backup und Restore**
  - ◆ ◇ etcd
  - ◆ ◇ Cluster-Backup (z.B. Kasten)
- **◆ Verschlüsselung**
  - ◆ ◇ Datenbankverschlüsselung (ETCD Verschlüsselung)
  - ◆ ◇ mTLS mit Service Mesh
- **◆ Erarbeitung einer möglichen Security Konfiguration eines Kubernetes Clusters**