

LI530 Kubernetes Advanced

Kurzbeschreibung:

Erfahrene Linux/Unix-Administratoren mit soliden Kubernetes-Kenntnissen vertiefen ihr Wissen in fortgeschrittenen Konzepten. Vermittelt werden Container-Deployments, Helm-Templates und der Aufbau eigener Charts. Behandelt werden Zugriffskontrollen, Authentifizierung, PodSecurityPolicy, NetworkPolicies sowie Image-Sicherheit, CVE-Scanning und Container-Signierung. Ergänzt wird das Training durch ServiceMesh mit Istio und Kubernetes Operatoren.

Zielgruppe:

Das Seminar **LI530 Kubernetes Advanced** richtet sich in erster Linie an Linux/Unix Administratoren.

Voraussetzungen:

Um dem Kursinhalten und dem Lerntempo des Workshops **LI530 Kubernetes Advanced** gut folgen zu können, sind stabile Vorkenntnisse in den Bereichen Linux und Kubernetes nötig.

Wir empfehlen Ihnen den vorherigen Besuch der folgenden Kurse:

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

Sonstiges:

Dauer: 5 Tage

Preis: 2990 Euro plus Mwst.

Ziele:

Nach Abschluss des Kurses **LI530 Kubernetes Advanced** sind die Teilnehmer in der Lage, die vorgestellten Konzepte im Besonderen das Sicherheitskonzept und darauf aufbauend die ServiceMesh Technologie in eigenen Projekten sinnvoll einzusetzen.

Inhalte/Agenda:

- - ◆ **Bereitstellung von Applikationen im Container**
 - ◆ ◊ Übersicht der Deployment und Template Engines
 - ◊ Helm, Charts und Repo
 - ◊ Aufbau von Helm Charts
 - ◆ **Zugriffsberechtigungen auf die Kubernetes API**
 - ◆ ◊ Übersicht:
 - ◊ · Service Account Tokens (JWT)
 - OpenID Connect Tokens (JWT)
 - Webhook Token Authentication
 - ◊ Beispiele und Übungen
 - ◆ **Sicherheitseinstellungen für den Kubernetes-Cluster**
 - ◆ ◊ Absicherung der Kubernetes-Serverkomponenten
 - ◆ ◊ Policies im Kubernetes-Cluster
 - ◊ · PodSecurePolicy
 - NetworkPolicies
 - Webhook Admission Controller
 - Beispiele und Übungen
 - ◊ Image & Container Security
 - ◊ · Container Image CVE Scanning
 - Container Image signieren
 - ◆ **ServiceMesh**
 - ◆ ◊ Was ist ein ServiceMesh?
 - ◊ Übersicht
 - ◊ ServiceMesh am Beispiel Istio
 - ◆ **Kubernetes Operatoren**
 - ◆ ◊ Funktionsweise von Operatoren in Kubernetes
 - ◊ Erstellen eines Operators