

# Application Programmable Interfaces (API)

## Programmierschnittstellen zu Netzwerken

Bei der Steuerung und Kontrolle von Netzwerken wird zunehmend auf Automatisierung und Programmierung gesetzt. Daraus resultiert der Wunsch nach hochgradig performanten Schnittstellen zu Routern und Switches. Dem haben die meisten namhaften Hersteller Rechnung getragen, indem sie den Zugang zu den Betriebssystemen ihrer Geräte über ein Application Programmable Interface (API) gestatten. Diese Schnittstellen haben sich in der Software-Entwicklung bewährt und finden nun auch in Netzwerken Anwendung. Gegenüber den klassischen Zugriffsmethoden über das CLI oder SNMP bieten moderne APIs einen für andere Applikationen optimierten Zugriff auf die volle Funktionalität. Eine API eignet sich deshalb auch hervorragend für den Einsatz einer zentralen Kontrolle bei SDN. Diese Schulung soll ein grundlegendes Verständnis für die Funktionsweise und den Zugriff auf eine API erzeugen, was durch praktische Übungen und Demonstrationen verdeutlicht und vertieft wird. Des Weiteren werden sinnvolle Anwendungen von APIs zur Netzwerk-Steuerung diskutiert und anhand von Praxis-Beispielen illustriert.

### Kursinhalt

- Datenstrukturen und ihre Repräsentation
- Grundlagen zu APIs
- API-Typen
- Programmierung von APIs
- Zugriff auf APIs
- REST API
- NETCONF und RESTCONF
- Use Cases für APIs
- Einsatz von APIs bei SDN
- Automatisierung und Orchestrierung
- APIs bei Cisco und Juniper
- Praktische Demonstrationen und Übungen

**E-Book** Das ausführliche deutschsprachige digitale Unterlagenpaket, bestehend aus PDF und E-Book, ist im Kurspreis enthalten.

### Zielgruppe

Die Schulung richtet sich an Planer und Administratoren, die sich über die Möglichkeiten der neuartigen APIs im Netzwerk kundig machen sowie die Struktur einer API und die Zugriffsmöglichkeiten auf eine API erlernen möchten. Wer in der Lage sein muss, das Potenzial einer API gegenüber klassischen Methoden abzugrenzen, und gängige Tools zu nutzen, um ohne Programmierkenntnisse auf eine API zuzugreifen, wird in diesem Kurs fündig.

### Voraussetzungen

Vorkenntnisse in Konfiguration und Operating von Netzwerken sind für eine erfolgreiche Teilnahme wünschenswert. Programmierkenntnisse sind nicht unbedingt erforderlich.

### Dieser Kurs im Web



Alle tagesaktuellen Informationen und Möglichkeiten zur Bestellung finden Sie unter dem folgenden Link: [www.experteach.at/go/API1](http://www.experteach.at/go/API1)

### Vormerkung

Sie können auf unserer Website einen Platz kostenlos und unverbindlich für 7 Tage reservieren. Dies geht auch telefonisch unter 06074 4868-0.

### Garantierte Kurstermine

Für Ihre Planungssicherheit bieten wir stets eine große Auswahl garantierter Kurstermine an.

### Ihr Kurs maßgeschneidert

Diesen Kurs können wir für Ihr Projekt exakt an Ihre Anforderungen anpassen.

Training		Preise zzgl. MwSt.	
<b>Termine in Deutschland</b>	<b>3 Tage</b>	<b>€ 2.195,-</b>	
<b>Online Training</b>	<b>3 Tage</b>	<b>€ 2.195,-</b>	
<b>Termin/Kursort</b>	Kursprache Deutsch		
04.08.-06.08.25	Frankfurt	10.11.-12.11.25	Frankfurt
04.08.-06.08.25	Online	10.11.-12.11.25	Online

Stand 07.05.2025



EXPERTeach



# Inhaltsverzeichnis

## Application Programmable Interfaces (API) – Programmierschnittstellen zu Netzwerken

<b>1 Einführung in die Thematik</b>	<b>3.1.4</b> RPC und RPC Reply	<b>4.6.2</b> Swagger UI des Servers
<b>1.1</b> Definition von API	<b>3.1.5</b> NETCONF-Operationen	<b>5 Google Remote Procedure Call (gRPC)</b>
<b>1.1.1</b> API - Lokal oder Remote	<b>3.1.6</b> Capability Advertisement	<b>5.1</b> gRPC - Funktion und Features
<b>1.1.2</b> Software Libraries	<b>3.1.7</b> NETCONF Notifications	<b>5.1.1</b> Funktionsprinzip
<b>1.1.3</b> Beispiel mit Python	<b>3.2</b> RESTCONF	<b>5.1.2</b> Protocol Buffers: Formatierung und Encoding
<b>1.2</b> APIs in Netzwerken	<b>3.2.1</b> CRUD Operationen	<b>5.1.3</b> Verpackung mit HTTP/2
<b>1.2.1</b> Command Line Interface	<b>3.2.2</b> Identifikation von Listen-Instanzen	<b>5.1.4</b> Betriebsmodi
<b>1.2.2</b> Datenmodelle für das Netzwerk	<b>3.2.3</b> Query Parameter bei lesendem Zugriff	<b>5.2</b> Übersicht über Anwendungen von gRPC
<b>1.2.3</b> Messaging	<b>3.2.4</b> Beispiel für GET	<b>5.3</b> Abgrenzung zu REST API
<b>1.2.4</b> SNMP: Ausgedient oder Totgesagte leben länger?	<b>3.2.5</b> Beispiel für POST	<b>6 API Tools und Use Cases</b>
<b>1.2.5</b> Software Defined Networking und API	<b>4 REST API</b>	<b>6.1</b> API Tools und Software
<b>1.3</b> API-Klassifizierung	<b>4.1</b> Definition von REST API	<b>6.1.1</b> cURL
<b>1.4</b> HTTP	<b>4.1.1</b> Generische Definition	<b>6.1.2</b> pyang
<b>1.4.1</b> HTTP Methoden	<b>4.1.2</b> REST als Web Architecture	<b>6.1.3</b> ncclient
<b>1.4.2</b> Caching	<b>4.1.3</b> Umsetzung mit HTTP(s)	<b>6.1.4</b> requests
<b>1.4.3</b> HTTP Header	<b>4.1.4</b> Software Architektur von REST APIs	<b>6.1.5</b> Postman
<b>1.4.4</b> MIME Types	<b>4.1.5</b> Software Development Kits	<b>6.1.6</b> Yangesuite
<b>1.4.5</b> Query-Parameter	<b>4.2</b> Details der Realisierung	<b>6.2</b> APIs von Netzwerk-Komponenten
<b>1.4.6</b> Status Codes	<b>4.2.1</b> Authentisierung	<b>6.2.1</b> API Infrastruktur des NX-OS
<b>1.4.7</b> HTTPs	<b>4.2.2</b> Autorisierung	<b>6.2.2</b> JUNOS
<b>2 Datenstrukturen und Datenmodelle</b>	<b>4.2.3</b> Content Negotiation	<b>6.2.3</b> Check Point
<b>2.1</b> Strukturierte Daten	<b>4.2.4</b> Paging	<b>6.3</b> SDN
<b>2.1.1</b> CRUD	<b>4.2.5</b> Asynchronous Operations	<b>6.3.1</b> Architektur des Controllers
<b>2.1.2</b> Hierarchische Daten	<b>4.2.6</b> API Versioning	<b>6.3.2</b> OpenDaylight
<b>2.1.3</b> Data Model und Data Modelling Language	<b>4.2.7</b> Angriffsvektoren und Absicherung	<b>6.3.3</b> ONOS Controller
<b>2.2</b> Serialization Languages	<b>4.2.8</b> Webhooks	<b>6.3.4</b> Cisco DNA Center
<b>2.2.1</b> XML	<b>4.3</b> REST API Dokumentation	<b>6.3.5</b> Meraki APIs
<b>2.2.2</b> JSON	<b>4.3.1</b> Grundstruktur	<b>6.3.6</b> Cisco SD-WAN
<b>2.2.3</b> YAML	<b>4.3.2</b> Methode POST	<b>6.4</b> Weitere Anwendungen von APIs
<b>2.2.4</b> Character Set und Encoding Scheme	<b>4.3.3</b> Methode GET mit Query-Parametern	<b>6.4.1</b> Network Services Orchestrator
<b>2.3</b> YANG	<b>4.3.4</b> Schema-Definitionen	<b>6.4.2</b> APIs im ETSI-Modell für NFV
<b>2.3.1</b> YANG Statements	<b>4.3.5</b> Swagger Editor	<b>6.4.3</b> Model Driven Telemetry
<b>2.3.2</b> Obligatorische und informelle Angaben	<b>4.3.6</b> Swagger User Interface	<b>6.4.4</b> Webex
<b>2.3.3</b> Objekt-Deklarationen	<b>4.4</b> API Design Richtlinien	
<b>2.3.4</b> JSON Encoding	<b>4.4.1</b> HTTP Methoden	
<b>2.3.5</b> Hilfsfunktionen	<b>4.4.2</b> Refinement	
<b>2.3.6</b> Modul und Submodul	<b>4.4.3</b> Consistency	
<b>2.3.7</b> Quellen für YANG-Modelle	<b>4.4.4</b> Errors	
<b>3 YANG-basierende APIs</b>	<b>4.4.5</b> Rate Limit	
<b>3.1</b> NETCONF	<b>4.4.6</b> Nested Ressource Expansion	
<b>3.1.1</b> NETCONF Datastores	<b>4.5</b> Message Queueing	
<b>3.1.2</b> Protokoll-Schichten	<b>4.5.1</b> MQTT	
<b>3.1.3</b> Subtree Filter	<b>4.5.2</b> REST API und / oder Message Queueing	
	<b>4.6</b> Beispiel mit Python und fastapi	
	<b>4.6.1</b> Code für den REST API Server	

