

# Puppet

## Einstieg in die Automatisierung und Orchestrierung

Der Wunsch nach stärkerer Automatisierung ist in vielen Unternehmen nicht mehr wegzudenken. Mit Puppet kann die Installation, Konfiguration und Wartung der Systeme automatisiert und vereinfacht werden. Neben der Möglichkeit administrativen Aufgaben zu beschleunigen, können auch Aufgaben wiederverwendbar implementiert werden. Nicht nur für Administratoren schafft Puppet neue Möglichkeiten, sondern auch Anwendungsentwickler profitieren maßgeblich von einer solchen Plattform. In diesem Workshop erlernen Sie die Grundlagen für den Umgang mit Puppet. Neben der Vermittlung theoretischen Wissens werden alle Ideen auch praxisnah umgesetzt.

### Kursinhalt

- Architektur von Puppet
- Installation und Setup von Puppet
- Ad-hoc-Tasks mit Puppet Bolt
- Erstellen von Manifest-Dateien in der Puppet Language
- Arbeiten mit Klassen, Modulen, Rollen und Profilen
- Einführung in das Datenmanagement via Hiera
- Einblick in Templates via Embedded Puppet
- Verwaltung von Linux- und Windows-Systemen
- Einführung in die Verwaltung von Netzwerk-Devices mit Hilfe von Puppet-Proxys

**E-Book** Sie erhalten das ausführliche deutschsprachige Unterlagenpaket aus der Reihe ExperTeach Networking – Print, E-Book und personalisiertes PDF! Bei Online-Teilnahme erhalten Sie das E-Book sowie das personalisierte PDF.

### Zielgruppe

Der Kurs richtet sich an alle, die einen Einstieg in das Thema Automatisierung mit Puppet suchen, ein gewisses technisches Verständnis mitbringen und sich nicht vor Neuerungen scheuen. Server-Administratoren, Plattform-Architekten, die im Automatisierungs-Umfeld arbeiten, liefert der Kurs ein fundiertes Know-How-Fundament und einen Einblick in die Möglichkeiten von Puppet.

### Voraussetzungen

Die Bereitschaft, sich mit den Themen Automatisierung und Puppet technisch auseinanderzusetzen und ein Grundverständnis der Rechenzentrums-Bausteine Computer, Network und Storage sind Voraussetzung für eine erfolgreiche Kursteilnahme. Grundlegende Linux-Kenntnisse sind erforderlich.

### Dieser Kurs im Web



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

### 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.

Stand 27.02.2024

Training	Preise zzgl. MwSt.
<b>Termine in Deutschland</b>	<b>3 Tage CHF 2.195,-</b>
<b>Online Training</b>	<b>3 Tage CHF 2.195,-</b>
<b>Termin/Kursort</b>	Kurssprache Deutsch
30.09.-02.10.24  Düsseldorf	30.09.-02.10.24  Online



# Inhaltsverzeichnis

## Puppet – Einstieg in die Automatisierung und Orchestrierung

- 1 Orchestrierungstools im Überblick**
  - 1.1 Orchestrierung im Überblick
  - 1.2 Ansible
  - 1.3 Salt
  - 1.4 Puppet
- 2 Puppet – Varianten**
  - 2.1 Open Source vs. Closed Source
  - 2.2 Puppet Bolt
  - 2.3 Puppet – standalone
  - 2.4 Puppet – Client/Server
- 3 Installation und Konfiguration der Komponenten**
  - 3.1 Puppet Bolt
    - 3.1.1 Installation Puppet Bolt
    - 3.1.2 Benutzer anlegen
    - 3.1.3 Passwordless Login
    - 3.1.4 Keys kopieren
  - 3.2 Puppet Client/Server
    - 3.2.1 Installation Puppet Server
    - 3.2.2 Installation Puppet Agent
    - 3.2.3 Starten der Dienste
    - 3.2.4 Signieren der CSR
    - 3.2.5 Anzeigen auf dem Client
- 4 Puppet Bolt**
  - 4.1 Das Kommando bolt – Subkommandos
    - 4.1.1 Das Verzeichnis .puppetlabs
  - 4.2 Das Inventory – die Liste der Hosts
    - 4.2.1 Einzelne Hosts
    - 4.2.2 Verbindungsdetails
    - 4.2.3 Gruppen von Hosts
    - 4.2.4 Inventory-Variable
  - 4.3 Adhoc-Kommandos mit bolt command
  - 4.4 Upload und Download von Dateien
  - 4.5 Upload und Ausführen von Skripten
  - 4.6 Bolt Tasks
    - 4.6.1 Software-Verwaltung
    - 4.6.2 Service-Management
  - 4.7 Bolt Plans
  - 4.8 Bolt Projects
    - 4.8.1 Bolt Projects – Beispiel Plan
    - 4.8.2 Bolt Projects – Beispiel Task
- 5 Manifeste**
- 5.1 Aufbau**
- 5.2 Ressourcen**
  - 5.2.1 File
  - 5.2.2 Package
  - 5.2.3 Service
  - 5.2.4 User und Group
  - 5.2.5 Defined Resources
- 5.3 Klassen**
  - 5.3.1 Verzeichnisstruktur
  - 5.3.2 Beispiel nginx
- 5.4 Module**
  - 5.4.1 Verzeichnisstruktur
  - 5.4.2 Beispiel
  - 5.4.3 Klassenaufruf: contain vs. include
- 5.5 Rollen und Profile**
  - 5.5.1 Verzeichnisstruktur
  - 5.5.2 Manifeste mit Rollen und Profilen
- 6 Variablen, Facts und Expressions**
  - 6.1 Variablen
    - 6.1.1 Arrays
    - 6.1.2 Hashes
  - 6.2 Facts
  - 6.3 Bedingte Ausdrücke
    - 6.3.1 case – Statement
  - 6.4 Expressions
    - 6.4.1 Vergleichsoperatoren
    - 6.4.2 Reguläre Ausdrücke
  - 6.5 Schleifen
    - 6.5.1 Schleife über ein Hash
  - 6.6 Datenmanagement mit Hiera
    - 6.6.1 hiera.yaml
    - 6.6.2 Die Lookup-Funktion
    - 6.6.3 Automatisierter Klassenparameter-Lookup in Hiera
  - 6.7 Dateimanagement mit Templates
    - 6.7.1 Template Tags
- 7 Puppet Forge**
  - 7.1 Puppet Forge
    - 7.1.1 Modul: stdlib
  - 7.2 Manuelle Installation
  - 7.3 Puppetfile
- 8 Verwaltung von Windows-Systemen & Netzwerkdevices**
  - 8.1 Puppet für Windows
    - 8.1.1 Paketinstallation
    - 8.1.2 Module für Windows
  - 8.2 Netzwerk-Orchestrierung
    - 8.2.1 Der Puppet Proxy
    - 8.2.2 Direktkommunikation via Bolt
- A Abkürzungsverzeichnis**
- B Befehle**

