

Installation von Ansible auf Debian-basierten Systemen

Einführung: Der erste Schritt in die Welt der Automatisierung

Ansible ist eines der flexibelsten und leistungsstärksten Tools für die IT-Automatisierung. Bevor Sie jedoch mit der Automatisierung starten können, müssen Sie Ansible auf Ihrem System installieren und konfigurieren. In diesem Artikel führen wir Sie Schritt für Schritt durch die Installation von Ansible auf Debian-basierten Linux-Systemen, insbesondere Ubuntu, und geben praktische Tipps, wie Sie sicherstellen können, dass Ihr Setup fehlerfrei funktioniert.

1. Voraussetzungen für die Installation

Bevor Sie Ansible installieren, sollten Sie sicherstellen, dass Ihr System die grundlegenden Anforderungen erfüllt:

1.1 Unterstützte Betriebssysteme

- Debian 10 („Buster“) oder höher.
- Ubuntu 20.04 („Focal Fossa“) oder höher.

1.2 Systemanforderungen

- Ein Benutzerkonto mit Root-Rechten oder Zugriff auf sudo.
- Ein funktionierendes Netzwerk (SSH-Verbindungen zu Zielsystemen erfordern Netzwerkzugang).

1.3 Abhängigkeiten

Ansible benötigt:

- Python (mindestens Version 3.8 auf dem Control Node).
- SSH-Zugriff auf die Zielsysteme (kein spezieller Ansible-Agent erforderlich).

2. Installation von Ansible: Schritt-für-Schritt-Anleitung

2.1 System aktualisieren

Starten Sie mit einem Update Ihres Systems, um sicherzustellen, dass alle Abhängigkeiten auf dem neuesten Stand sind:

```
[bash]
```

```
sudo apt update && sudo apt upgrade -y
```

2.2 Python und notwendige Pakete installieren

Ansible benötigt Python als Laufzeitumgebung. Installieren Sie Python und die erforderlichen Pakete:

```
[bash]
```

```
sudo apt install python3 python3-pip -y
```

Prüfen Sie anschließend, ob Python korrekt installiert ist:

```
[bash]
```

```
python3 --version
```

2.3 Ansible aus den Standard-Repositories installieren

Ansible ist in den offiziellen Repositories vieler Linux-Distributionen verfügbar. Installieren Sie es mit apt:

```
[bash]
```

```
sudo apt install ansible -y
```

Nach der Installation können Sie die Version überprüfen, um sicherzustellen, dass alles funktioniert:

```
[bash]
```

```
ansible --version
```

Hinweis: Die Version in den Standard-Repositories ist oft nicht die neueste. Falls Sie die aktuellste Version benötigen, folgen Sie Abschnitt **2.4**.

2.4 Installation der neuesten Version aus dem offiziellen PPA

Um die aktuellste Version von Ansible zu erhalten, fügen Sie das offizielle Ansible-Repository hinzu:

```
[bash]
sudo apt update
sudo apt install software-properties-common -y
sudo add-apt-repository --yes --update ppa:ansible/ansible
sudo apt install ansible -y
```

Prüfen Sie anschließend die Version:

```
[bash]
ansible --version
```

2.5 Optional: Installation mit pip

Falls Sie Ansible unabhängig von den Paketmanagern Ihrer Distribution verwalten möchten, können Sie es mit Python's Paketmanager pip installieren:

```
[bash]
pip3 install ansible --user
```

Nach der Installation stellen Sie sicher, dass der Pfad zu den Benutzerpaketen (~/.local/bin) in Ihrer Umgebungsvariablen \$PATH enthalten ist:

```
[bash]
export PATH=$PATH:~/.local/bin
```

3. Erste Schritte nach der Installation

3.1 Überprüfen der Installation

Führen Sie einen Testbefehl aus, um sicherzustellen, dass Ansible betriebsbereit ist:

```
[bash]
```

```
ansible localhost -m ping
```

Wenn die Ausgabe wie folgt aussieht, funktioniert Ansible korrekt:

```
localhost | SUCCESS => {  
    "changed": false,  
    "ping": "pong"  
}
```

3.2 Konfiguration der Inventardatei

Die Inventardatei listet die Zielsysteme auf, die Ansible verwalten soll. Erstellen Sie eine Datei namens `inventory` und fügen Sie die IP-Adressen oder Hostnamen Ihrer Server hinzu:

```
[text]
```

```
[webserver]  
192.168.1.10  
192.168.1.11
```

```
[dbserver]  
192.168.1.20
```

Speichern Sie die Datei und testen Sie die Verbindung:

```
[bash]
```

```
ansible -i inventory all -m ping
```

3.3 Authentifizierung mit SSH

Damit Ansible ohne Passwortabfrage arbeiten kann, sollten Sie SSH-Schlüssel verwenden:

1. Generieren Sie einen SSH-Schlüssel:

```
[bash]
```

```
ssh-keygen
```

2. Kopieren Sie den Schlüssel auf die Zielsysteme:

```
[bash]
```

```
ssh-copy-id user@zielsystem
```

4. Häufige Probleme und Lösungen

4.1 Fehlende Abhängigkeiten

Wenn Ansible beim Starten Fehler wie „Python not found“ anzeigt, installieren Sie die fehlenden Pakete:

```
[bash]
```

```
sudo apt install python3
```

4.2 Verbindung schlägt fehl

- Stellen Sie sicher, dass SSH auf den Zielsystemen aktiviert ist.
- Prüfen Sie die Firewall-Regeln.

4.3 Modul „ping“ nicht gefunden

Stellen Sie sicher, dass Ansible korrekt installiert ist und die Umgebungsvariablen korrekt gesetzt sind.

5. Praktisches Beispiel: Erstellen eines einfachen Playbooks

Nachdem Ansible installiert ist, erstellen wir ein erstes Playbook, das Apache2 auf einem Server installiert:

Playbook-Datei (install_apache.yaml)

[yaml]

```
---
- name: Installiere Apache Webserver
  hosts: webserver
  become: yes
  tasks:
    - name: Update des Paketmanagers
      apt:
        update_cache: yes

    - name: Apache installieren
      apt:
        name: apache2
        state: present

    - name: Apache starten
      service:
        name: apache2
        state: started
        enabled: yes
```

Befehl zum Ausführen des Playbooks:

[bash]

```
ansible-playbook -i inventory install_apache.yaml
```

Fazit und Ausblick

Die Installation von Ansible auf Debian-basierten Systemen ist ein einfacher Prozess, der den Grundstein für leistungsstarke Automatisierungsprojekte legt. Mit wenigen Befehlen können Sie Ansible einrichten und Ihre IT-Infrastruktur effizienter gestalten.

Im nächsten Artikel zeigen wir, wie Sie Ansible konfigurieren und Ihre ersten Playbooks erstellen, um praktische Automatisierungsaufgaben zu lösen. Bleiben Sie dran und erfahren Sie, wie Sie Ansible optimal nutzen können!

Über Achim Schmidt

Achim Schmidt ist ein erfahrener IT-Spezialist mit über 30 Jahren Expertise in den Bereichen IT-Sicherheit, Netzwerkarchitektur und Infrastrukturmanagement. Seit den frühen 90er Jahren begleitete er den Aufbau des Internets in Deutschland, besonders in Bayern. Als technischer Leiter und später Produktmanager prägte er bedeutende Internetprojekte und Netzwerklösungen für namhafte Unternehmen. Neben seiner Tätigkeit als Autor zahlreicher Fachartikel und Bücher, u. a. zu Linux, IT-Sicherheit und Netzwerktechnologien, engagiert er sich als zertifizierter Datenmanager und Teilnehmer des BSI IT-Sicherheitskongresses 2024 für aktuelle Themen der Datensicherheit und KI.

Im Internet findet man weitere Informationen zu ihm unter <http://www.achim-schmidt.de/> und <http://www.it-phoenix.de/>