

AWX Installieren

Hier eine erprobte Schritt-für-Schritt-Anleitung zur Installation der Open-Source-Version von AWX (aktuell v17.1.0) mit Docker Compose – ganz ohne Kubernetes, offiziell zwar nur für Tests gedacht, in der Praxis aber voll funktionsfähig.

1. System-Abhängigkeiten installieren

Unter Ubuntu/Debian

```
sudo apt update
sudo apt install -y git gcc g++ python3-pip ansible curl
pip3 install docker docker-compose
```

(mpolinowski.github.io, docs.zpesystems.com)

Unter RHEL/CentOS 8

```
sudo dnf install -y git gcc gcc-c++ python3-pip ansible curl
sudo alternatives --set python /usr/bin/python3
pip3 install docker docker-compose
```

(mpolinowski.github.io, docs.zpesystems.com)

“ Damit sind Docker, Docker Compose, Ansible und die nötigen Build-Tools auf dem Host.

2. AWX-Quellcode klonen & zum Installer wechseln

```
git clone -b 17.1.0 https://github.com/ansible/awx.git
cd awx/installer
```

(docs.zpesystems.com, mpolinowski.github.io)

3. Secret-Key erzeugen

Der Installer benötigt einen 30-stelligen Schlüssel für die interne Verschlüsselung.
Wahlweise mit OpenSSL:

```
openssl rand -base64 30 > secret_key.txt
```

(mpolinowski.github.io)

Oder mit pwgen:

```
pwgen -N1 -s30 > secret_key.txt
```

(docs.zpesystems.com)

4. Inventardatei anpassen

Öffne `installer/inventory` in deinem Lieblings-Editor und setze mindestens diese Werte im Block `[all:vars]`:

```
# Web-Zugriff
host_port=80
host_port_ssl=443

# Admin-Account
```

```
admin_user=admin
admin_password=MeinSicheresPasswort

# Secret Key (aus secret_key.txt)
secret_key=<Inhalt von secret_key.txt>

# (Optional) Pfade für persistente Daten
postgres_data_dir="/opt/awx/pgdocker"
project_data_dir="/opt/awx/projects"
docker_compose_dir="/opt/awx/awxcompose"
```

(docs.zpesystems.com, mpolinowski.github.io)

“ Achte darauf, dass die Verzeichnisse existieren und Docker-Prozesse darauf schreiben dürfen (z.B. `sudo mkdir -p /opt/awx/{pgdocker,projects,awxcompose}`).

5. Installation per Ansible-Playbook starten

```
ansible-playbook -i inventory install.yml
```

(docs.zpesystems.com, mpolinowski.github.io)

Der Playbook-Lauf erzeugt via Docker Compose Container für PostgreSQL, Redis, das AWX-Web-UI und den Task-Worker.

6. Kontrolle & Erster Zugriff

1. Status prüfen

```
docker ps
```

2. Installations-Logs beobachten

```
docker logs -f awx_task
```

Warte, bis dort die Datenbank-Migrations-Tasks erfolgreich durchgelaufen sind.

3. **Web-UI öffnen**

Rufe im Browser auf:

```
http://<Deine-Server-IP>:80
```

Login mit `admin / MeinSicheresPasswort`.

(docs.zpesystems.com, mpolinowski.github.io)

Fertig! Du hast nun AWX 17.1.0 vollständig in Docker-Containern laufen, ganz ohne Kubernetes.

Revision #1

Created 18 June 2025 11:39:20 by Stefan Mechler

Updated 18 June 2025 11:39:30 by Stefan Mechler