

# Inventory Datei - Welche Hosts dürfen es sein?

## Einleitung

Die Inventory Datei beschreibt welche Hosts mit den playbooks bespielt werden sollen.

Es können variablen und Parameter zu den Hosts angegeben werden.

in der Regel ist eine Inventory-Datei eine INI Datei.

kann aber auch in yaml erstellt werden.

## Einzelhosts und Gruppen

Hier ein Beispiel einer Inventory datei mit zwei Gruppen und einem Host ohne gruppe.

Die Inventory Datei hat INI Datei Syntax.

Einzelne Hosts werden vom script immer aufgerufen/abgearbeitet.

Alles was global also ohne Gruppe ist muss immer an den anfang und wird ohne weiteres dazutun aufgerufen.

Wenn Hostnamen verwendet werden, sicherstellen, das diese auch aufgelöst werden können.

Die Gruppen werden in eckigen Klammern geschrieben. und darunter dann die jeweiligen hosts mit Hostname oder ip Adresse eingetragen.

```
203.0.113.100
```

```
[nginx]
```

```
203.0.113.0
```

```
[wordpress]
```

```
203.0.113.1
```

Nun haben wir zwei Gruppen nginx und wordpress

Wir können auch ranges definieren. So braucht nicht jeder Hostname eingefügt werden, beispiel beim IP range oder Hostname range

```
[webservers]
```

```
wwwserver[01:50].example.com
```

Erster Server wäre dann

```
wwwserver01.example.com
```

Letzter Server wäre dann

```
wwwserver50.example.com
```

```
[webservers]
```

```
203.0.113.[1:50]
```

Erster Server wäre dann

```
203.0.113.1
```

Letzter Server wäre dann

```
203.0.113.50
```

Verschachtelte Gruppen.

Es ist möglich Gruppen in weitere Gruppen aufzudröseln.

Hier haben wir die Hosts Gruppen:

Routers

switches

firewalls

Diese Gruppen hab wir als Kinder in folgende Gruppen gepackt

network

security

lan

Wie zu sehen sind einige Gruppenkinder auch Kinder anderer Hauptgruppen.

Also eine Gruppe kann auch mehreren Übergeordneten Gruppen zu gewiesen sein.

```
[routers]
10.10.10.1
10.10.10.2

[switches]
172.16.1.1
172.16.1.2

[firewalls]
192.168.1.1

[network:children]
routers
switches
firewalls

[security:children]
routers
firewalls

[lan:children]
routers
switches
```

## Variablen in der Inventory Datei

Es können auch Variablen deklariert werden, die dann im Playbook zu Verfügung stehen.

Bei einem einzelnen Host egal ob Global oder in der Gruppe wird die Variable oder auch mehrere einfach hinter dem Hostnamen deklariert. Es gibt auch schon vorgefertigte Variablen die Ansible dann benutzt wenn sie deklariert werden.

Das sind alle Variablen die mit `ansible_` anfangen. Sie dienen um z.B. einen bestimmten SSH Benutzer anzugeben oder einen anderen SSH Port zu nehmen.

Die Syntax

```
variablename=wert
```

Sollen mehrere Variablen definiert werden einfach mit Leerzeichen trennen.

Werte mit Leerzeichen sind nicht gültig. Bzw. müssen in Anführungszeichen gesetzt werden.

Beispiel:

```
192.168.0.1 ansible_user=root ansible_port=222
```

Anhand den obigen Beispiels nimmt Ansible zum verbinden zu dem Host den Benutzer root und den Port 222 an statt den standard Port 22.

Wir können auch variablen einer Gruppe hinzufügen. Wenn jeder host in der Gruppe diese Variablen haben soll.

Beispiel:

```
[routers]
10.10.10.1
10.10.10.2

[switches]
172.16.1.1
172.16.1.2

[routers:vars]
ansible_user=root
ansible_port=222

[switches:vars]
ansible_user=admin
ansible_port=22
```

Im oben genannten Beispiel wird in eckigen Klammern der Gruppenname gschrieben und dann mit Doppelpunkt direkt dahinter der Tag vars

Nun werden die Variablen (Eine Zeile für eine Variable) untereinadner eingetragen.

Nun hat die Gruppe routers als Benutzername root und den port 222 und die Gruppe switches den Benutzernamen admin und den Port 22.

Möchte man eine Variable defienieren die für alle Gruppen giltt wird als Gruppenname all und der Tag vars verwendet.

Somit giltt für alle Gruppen der Benutzer admin und der Port 22

```
[routers]
10.10.10.1
```

```
10.10.10.2
```

```
[switches]
```

```
172.16.1.1
```

```
172.16.1.2
```

```
[all:vars]
```

```
ansible_user=admin
```

```
ansible_port=22
```

## Vorfefinierte Variablen

Die Variablen die in der Inventory Datei verwendet werden können

```
ansible_user=admin [ ]#Benutzername für die ssh verbindung
```

```
ansible_port=22[ ]#Port für die SSH Verbindung
```

```
ansible_password=mypass[ ]#SSH Passwort wenn keine Schlüsseldatei verwendet wird. Passwörter niemals  
Plaintext speichern. Nehmt dazu das ansible vault (Dazu mehr im Kapitel Ansible Vault)
```

```
ansible_ssh_private_key_file=path_to/ssh_privkey [ ]#Der Private Schlüssel der verwendet werden soll zum  
Verbinden, wenn der Schlüssel dazu nicht im .ssh Verzeichnis liegt
```

```
ansible_connection=ssh[ ]#Die Verbindungsmethode wie ansible zum Host verbinden soll. Hier ssh
```

```
ansible_become=yes[ ]#Hat der ssh Benutzer keine root rechte soll sudo verwendet werden (yes,no)
```

```
ansible_become_pass=passwort123[ ]#Braucht der sudo Benutzer ein Kennwort, kann das Kennwort übergeben  
werden. Wird nur ausgewertet wenn ansible_become yes festgelegt ist
```

```
ansible_shell_type=sh[ ]#Wählt den shell Type aus. Entweder sh csh oder fish shell
```

---

Version #12

Erstellt: 14 August 2022 12:40:10 von Admin

Zuletzt aktualisiert: 4 Dezember 2022 07:51:49 von Admin