

Ansible_

Grundlagen

Konfigurationsmanagement

Konfiguration von Systemen - Infrastructure as Code

Verteilung und Orchestrierung von Software

Orchestrierung von Systemen

Zero-Downtime Updates

Ad-hoc Kommandos

Operation am offenen Herzen

Python-Tool

Keine Programmierkenntnisse notwendig

Skripte in YAML-Syntax - Playbooks genannt

Voraussetzungen

Ansible Host (Linux) mit Ansible Installation

Auf den Zielrechnern: Python 2.x && ssh Verbindung

Vorteile

Ansible: Vorteile

Skripte unter Versionskontrolle

Verhinderung von Konfigurationsunterschieden

Sicht auf Systeme - nicht nur auf Applikationen

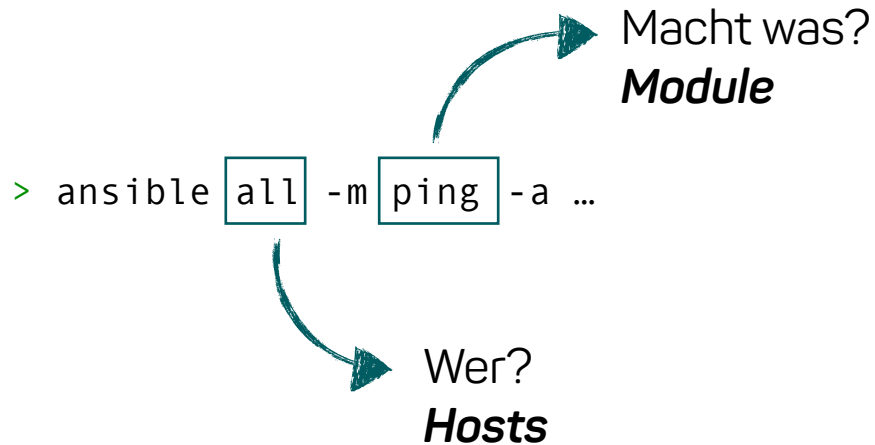
Applikationen auf identischen Systemen testen

Vereinfachung durch Abstraktion

How To?

- **\$ANSIBLE_CONFIG**
- **./ansible.cfg**
- **~/.ansible.cfg**
- **/etc/ansible/ansible.cfg**

Alive? Ad hoc



```
> ansible localhost -m ping
```

```
[WARNING]: Host file not found: /usr/local/etc/ansible/hosts
```

```
[WARNING]: provided hosts list is empty, only localhost is available
```

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


Inventory

```
> ansible all -m ping -i inv
```

```
> vi inv  
  
[all]  
localhost ansible_connection=local
```

```
> ansible all -m ping -i inv  
  
localhost | SUCCESS => {  
    "changed": false,  
    "ping": "pong"  
}
```

S

Inventory

Default Gruppen

iamalone
42.27.13.9



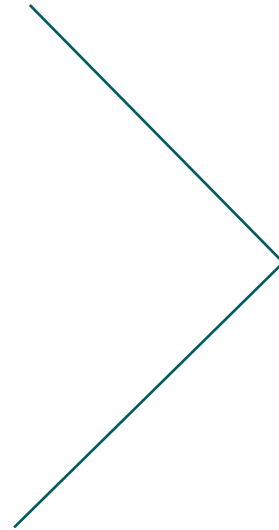
ungrouped

[muenster]
13.37.47.11
apache.mycompany.de

[koeln]
13.12.25.3

[proxyservers]
http[01:50].mycompany.de

[nrw:children]
koeln
muenster



all

> ansible proxyservers -m ping -i inv

Inventory

```
myserver ansible_host=12.15.32.9 ansible_port=5555 ansible_user=ec2user
```

Weitere Parameter

```
ansible_ssh_private_key_file
```

```
ansible_become
```

```
ansible_connection
```

```
ansible_ssh_common_args
```

http://docs.ansible.com/ansible/intro_inventory.html

Wir nutzen eine IAAS Plattform. Können Hosts dynamisch ermittelt werden?

```
> ansible all -m ping -i openstack.py
```

Viele weitere Plattformen

- Google Cloud Engine
- AWS
- Vagrant
- ...

Inventory Brainfood - Patterns

```
> ansible <#> -m <module>
# all                                # *
# web.mycompany.de
# 192.168.0.*                        # web*
# webserver:proxyserver
# webserver:!proxyserver
# webserver:&proxyserver
# webserver:proxyserver:&prod:!ffm
```

```
[webserver]
Keep
It
Simple

# webserver[0]    = Keep
# webserver[-1]   = Simple
# webserver[0:1]  = Keep, It
# webserver[1:]   = It, Simple
```

S

Ad-Hoc

Eigentliche Power liegt in Playbooks! Aber ...

- Schnelle und kleine Änderungen
- Tests
- Erste Schritte

... rechtfertigen Ad-Hoc Kommandos

Ansible Ad-Hoc

Remote Commands

- > `ansible all -a "sbin/reboot"`
- > `ansible all -m shell -a 'echo $var'`

Filetransfer

- > `ansible all -m copy -a "src=/tmp/foo dest=/srv/foo"`
- > `ansible all -m file -a "dest=/srv/foo/a.txt mode=600"`

Sind die wirklichen Arbeiter

Sind idempotent

Dokumentation

```
> ansible doc -l
```

```
> ansible doc yum
```

Hands-On: PONG?

Ziel: Erster Zugriff auf Remote Instanzen.

Was benötigen wir:

- ssh Zugang (Passwort/Key)
- Inventory-Datei

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