

Ansible

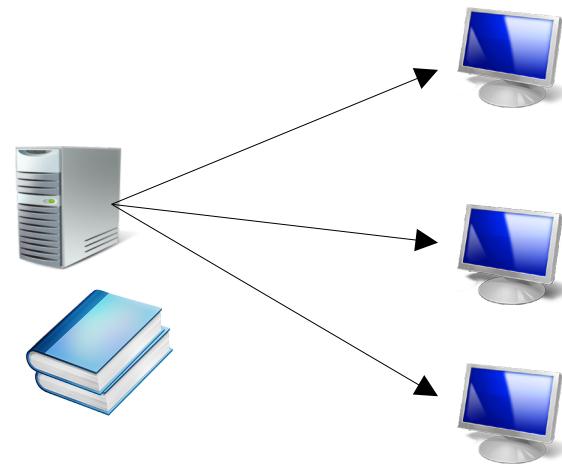


Wat is Ansible ?

■ Automation engine

- ◆ Automation cloud provisioning
- ◆ Configuration Management
- ◆ Application Deployment
- ◆ Intra-service orchestration

■ Gebruikt YAML playbooks



Hoe werkt ansible?

■ Eenvoudige taal (YAML)

- ◆ In die taal schrijf je een Playbook



■ Een Control (Management) Node beheert machines

- ◆ Unix based, MacOSX
- ◆ Windows Subsystem for Linux
- ◆ Windows Server



■ Geen databank, geen agents, geen daemons

- ◆ Enkel Python >=3.8
- ◆ Powershell

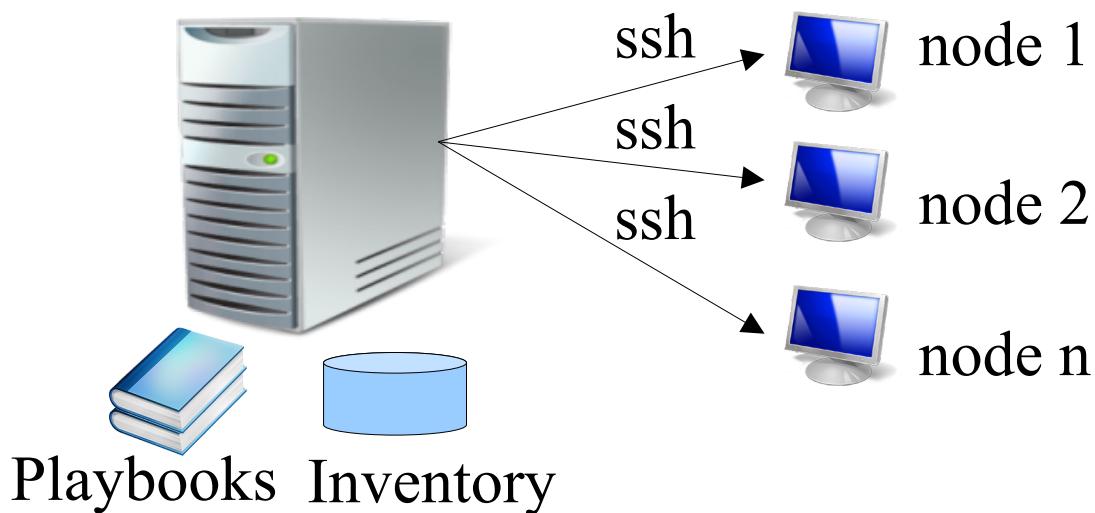


ANSIBLE

Hoe werkt ansible

■ Ansible Management Node voert een playbook uit

- Verbindt met nodes (die in inventory staan)
- Pusht Ansible modules naar nodes
- Start Ansible modules via SSH /WinRM
 - verwijderd modules wanneer ze uitgevoerd werden



Installatie Ubuntu

- **sudo apt update**
- **sudo apt install software-properties-common**
- **sudo add-apt-repository --yes --update ppa:ansible/ansible**
- **sudo apt install ansible**





Playbook voorbeeld

```
---
name: Stop firewall/Start IIS
hosts: windows-svr
become: yes
tasks:
  - name: Stop and disable firewall
    win_service
      name: MpsSvc
      state: stopped
      start_mode: disabled
  - name: Start IIS
    win_service
      name: W3Svc
      state: running
```





Play voorbeeld

- Specificatie voor aantal taken
- Zegt welke hosts welke taken moeten uitvoeren met welk privilege
- Meerdere plays vormen een playbook die draait op verschillende hosts

```
name: Stop firewall/Start IIS  
hosts: windows-svr  
become:yes
```



ANSIBLE



Playbook deploy nginx

- - name: Install nginx
 - hosts: webservers
 - tasks:
 - name: Ensure nginx is at the latest version
 - apt:
 - name:nginx
 - state:latest
 - name: Ensure nginx is running
 - service:
 - name: nginx
 - state: restarted



Playbook nakijken en uitvoeren

■ Ansible playbook lint

- ◆ sudo apt install ansible-lint
- ◆ ansible-lint playbook.yml

■ Uitvoeren van een playbook

- ◆ ansible-playbook playbook.yml



ANSIBLE

Ansible Inventories

- Lijst(en) met hosts
- Door deze lijst te filteren selecteer je de hosts waarmee je Ansible wilt gebruiken
- Default in bestand /etc/ansible/hosts (directory aanmaken)
 - ◆ Of eigen path met -i path
- Multiple inventory sources
 - ◆ Je kan meer dan 1 source opgeven en zo bijvoorbeeld een playbook toepassen op zowel de staging als productie omgeving
- `ansible-playbook get_logs.yml -i staging -i production`



ANSIBLE

Inventory in INI formaat

- **mail.example.com**
[webservers]
web1.example.com
web2.example.com
[dbservers]
db1.example.com
db2.example.com
db3.example.com



Inventory in YAML formaat

- **ungrouped:**
hosts:
 mail.example.com:
- **webservers:**
hosts:
 web1.example.com:
 web2.example.com:
- **dbservers:**
hosts:
 db1.example.com:
 db2.example.com:
 db3.example.com:



Inventory in YAML formaat met children

- **webservers:**

- hosts:**

- web1.example.com:**

- web2.example.com:**

- **dbservers:**

- hosts:**

- db1.example.com:**

- db2.example.com:**

- db3.example.com:**

- **servers:**

- children:**

- webservers:**

- dbservers:**



ANSIBLE

Inventory met ranges

- **webservers:**
hosts:
web[01:20].example.com:
- **dbservers:**
hosts:
db[a:f].example.com:

Modules

- **Modules voeren een bepaalde taak uit op de target machine:**

- copy kopieren/aanmaken bestanden
- apt installatie pakketten
- service opstarten/stoppem services
- shell commando of shellscript
- debug nakijken variabelen en werking



ANSIBLE

Handlers

- Tasks worden direct opgestart, een handler pas na een trigger (wanneer er iets is veranderd)
notify:
 - **Restart Apache Handler**

handlers:

- **name: Restart Apache Handler**

service:

name: apache2

state: restarted



ANSIBLE

Variabelen

- name: Variabele gebruiken

hosts: all

vars:

- bericht1: "hello"

- bericht2: "world"

tasks:

- name: Show var

debug: msg="Bericht {{ bericht1 }} {{ bericht2 }}"



ANSIBLE

Facts

- - name: Facts gebruiken
 - hosts: all
 - gather_facts: true
 - tasks:
 - name: Show a fact
 - debug: msg: "Date: {{ ansible_date_time.date }} "



Variabele met shell comando

- - name: Shell comando naar var
hosts: all
tasks:
 - name: Get timestamp from the system
shell: "date +%Y-%m-%d%H-%M-%S"
register: datetime
 - name: Set variables
set_fact:
cur_date: "{{ datetime.stdout[0:10] }}"
 - name: System timestamp
debug:msg: "Date: {{ cur_date }}"



Templates

Bestanden worden aangemaakt vanuit een jinja2 template

- **index.html.j2:**

```
<h1>Hello World!</h1>
<p>My IP is: {{ host_ip }}</p>
```

- **gebruik_van_een_template.yml:**

- name: Get host IP address

- set_fact:**

- host_ip: "{{ ansible_default_ipv4.address }}"**

- name: Update index.html with host IP

- template:**

- src: templates/index.html.j2**

- dest: /var/www/html/index.html**



ANSIBLE

Role: standaard directory structuur

- Een role bundelt alle elementen van een playbook
- ansible@master:~/roles\$ ansible-galaxy init nginx

```
ansible@master:~/roles$ tree nginx
```

nginx

```
└── defaults └── main.yml
└── files
└── handlers └── main.yml
└── meta └── main.yml
└── tasks └── main.yml
└── templates
└── inventory
└── vars └── main.yml
```



Referenties

■ Ansible documentation

- ◆ https://docs.ansible.com/ansible/latest/getting_started/get_started_playbook.html

■ How Ansible works

- ◆ <https://www.ansible.com/overview/how-ansible-works>

■ Playbook variables

- ◆ https://docs.ansible.com/ansible/latest/user_guide/playbooks_variables.html



ANSIBLE