

DEVOPS - 15/10/2024

ANSIBLE

ANSIBLE GEBRUIKEN

[HTTPS://GITHUB.COM/AFKIR-AYOUB/DEVOPS-ANSIBLE-2324](https://github.com/afkir-ayoub/devops-ansible-2324)

1. Indien nodig, installeer VirtualBox ([Download VirtualBox](#))
2. Installeer Ubuntu Server ([Download Ubuntu](#))

INSTALLEER VIRTUALBOX + UBUNTU SERVER (ZIE GITHUB)

WAT IS ANSIBLE?

OPEN-SOURCE CONFIGURATION MANAGEMENT SOFTWARE



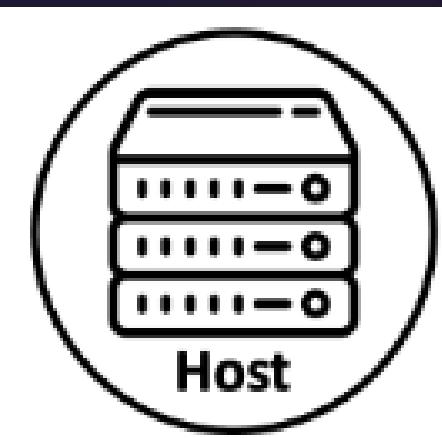
- Oorspronkelijk geschreven door Michael DeHaan in 2012
- Overgenomen door Red Hat in 2015
- Platformonafhankelijk: Werkt op Linux, MacOS & Windows
- Voor:
 - Automatisatie: automatische updates,...
 - Systeemconfiguratie beheren
 - CI/CD Pipeline



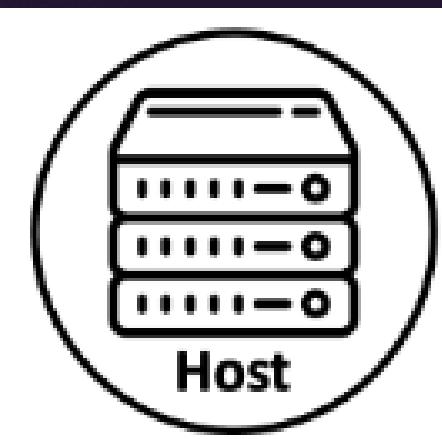
Red Hat



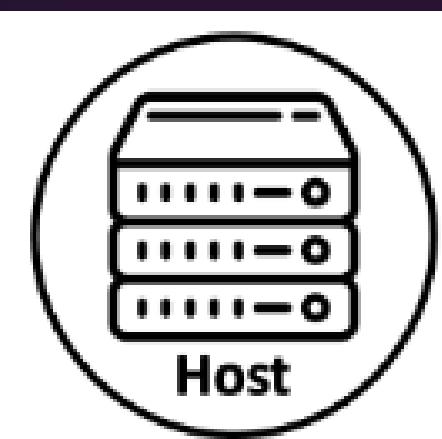
ANSIBLE USE-CASE



Installeer Python 3.18



Installeer Python 3.18



Installeer Python 3.18

Situatie:

- Je hebt 3 servers en wilt overal python installeren

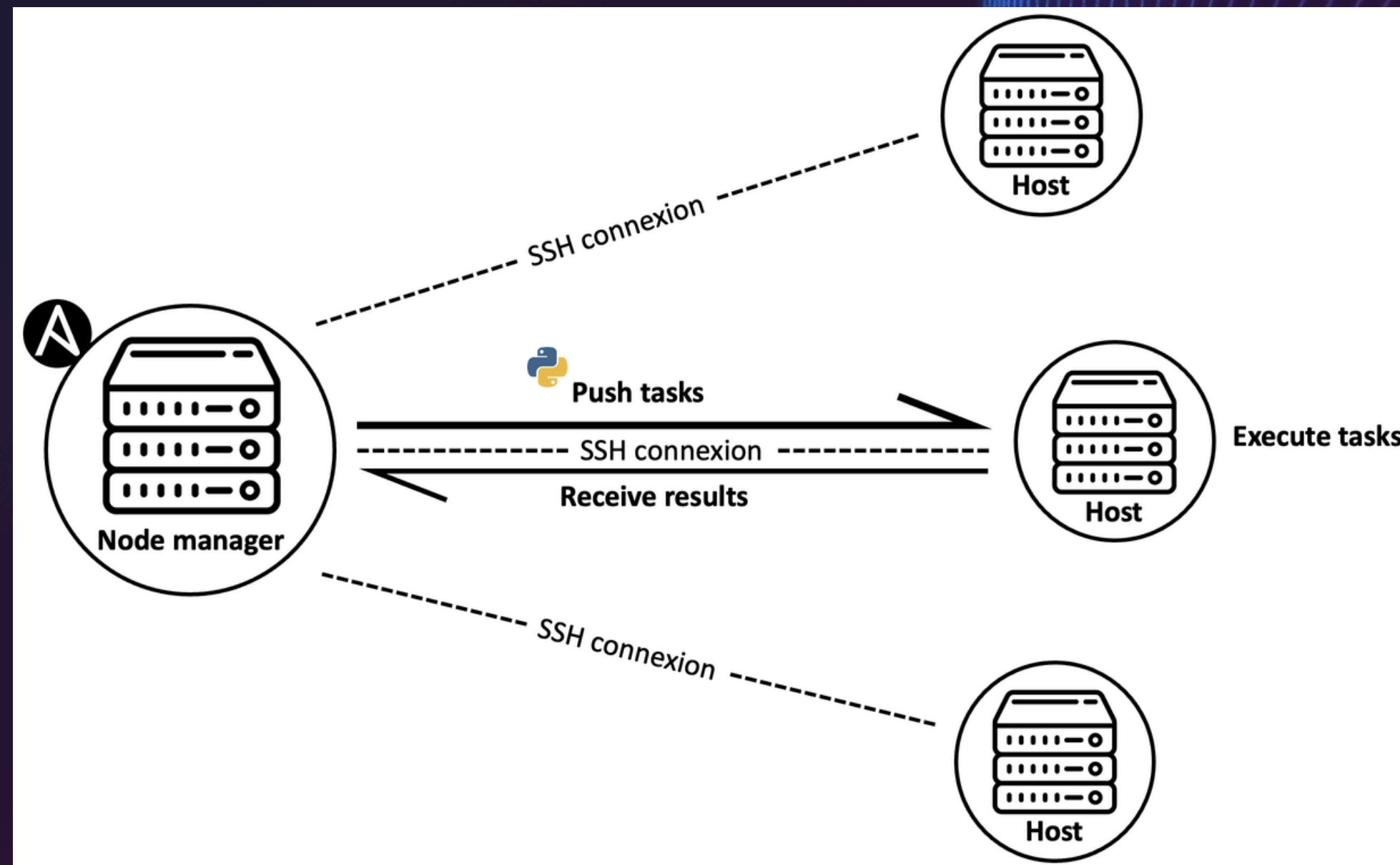
Stappenplan:

1. Elk server individueel connectie maken
2. Elk server individueel python installeren

Probleemstelling:

- Te repetitief
- Duurt te lang
- Wat met andere configuraties? Elke keer hetzelfde stappenplan met updates?

ANSIBLE USE-CASE



- 1 Master-Node die alle andere nodes configureert
- SSH-connectie
- Kan verschillende “tasks” gaan pushen op alle nodes
- Idempotent: “tasks” worden uitgevoerd alleen als nodig

ANSIBLE USE-CASE IN CI/CD PIPELINE

CI



Jenkins haalt code bij elk push

CD



ANSIBLE



Jenkins

Jenkins:

- Code halen bij elke GitHub Push
- Code Builden
- Code Testen

En dit automatisch & bij
elke server

Ansible:

- Installeert & configureert alle software (bvb; Python)
- Code naar servers kopiëren
- Servers herstarten

WAAROM ANSIBLE?



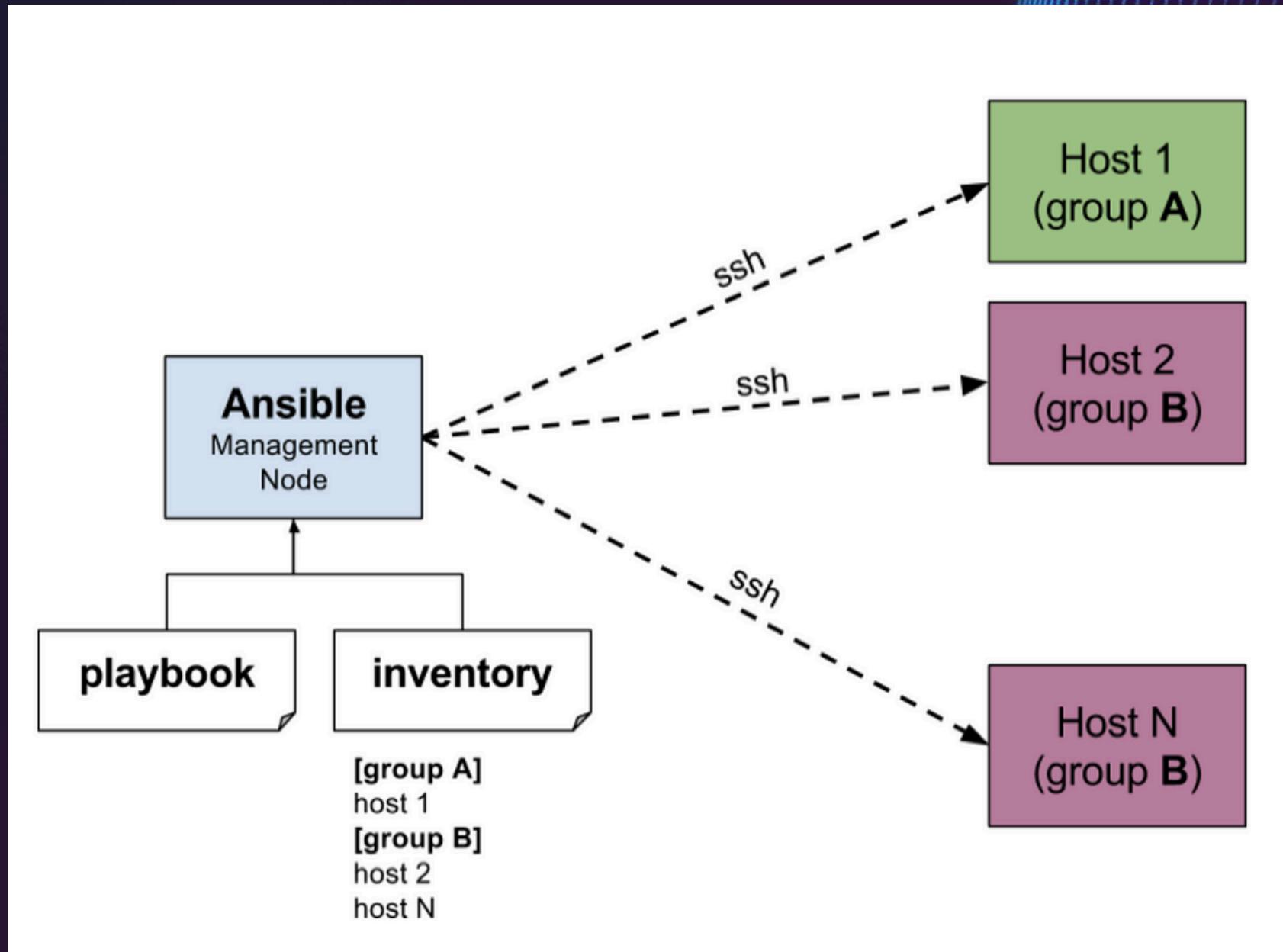
**VEREISEN AGENT OP DE
SLAVE-SERVERS**

WAAROM ANSIBLE?



**AGENT-LESS ZOALS
ANSIBLE MAAR IS WEL
COMPLEXER**

ANSIBLE & IAC (PLAYBOOK)



- Playbook sturen naar servers
- Mogelijk om servers te groeperen
- Inventory = file met alle groepen



ANSIBLE ROLLEN

FILESTRUCTUUR

```
roles/
└── my_role/
    ├── defaults/          # Hier zet je de standaardvariabelen
    ├── files/             # Bestanden die gekopieerd moeten worden naar de doelmachines
    ├── handlers/          # Handlers die door andere taken kunnen worden getriggerd
    ├── meta/              # Metadata van de rol (bijv. afhankelijkheden van andere rollen)
    ├── tasks/             # De taken die uitgevoerd moeten worden
    ├── templates/          # Sjablonen die gegenereerd worden met variabelen
    ├── vars/              # Extra variabelen die je kunt definiëren
    └── README.md           # Optioneel: beschrijving van de rol
```

- Herbruibaarheid van stukjes playbooks
- Overzichtelijker dus gemakkelijker voor onderhoud
- Community: Ansible Galaxy voor rollen van anderen te gebruiken

ANSIBLE BEVEILIGING



Vault

Om files te encrypten



Voor een passwordless connectie
met servers



Geen "root" maar users met
"sudo" rechten

VB-CODE ANSIBLE PLAYBOOK

NIEUWE USER AANMAKEN: DEEL 1

```
1  - hosts: all # Alle servers in het inventory bestand
2    become: yes # root rechten
3    tasks:
4
5      - name: Maak een nieuwe gebruiker aan (ansible_user) # De naam van de taak
6        user: # user module om een nieuwe gebruiker aan te maken
7          name: ansible_user # De naam van de gebruiker
8          state: present # Zorg ervoor dat de gebruiker bestaat
9          shell: /bin/bash
10
11     - name: Voeg gebruiker toe aan sudo-groep
12       user:
13         name: ansible_user
14         groups: sudo
15         append: yes
16
17     - name: Configureer sudo zonder wachtwoord voor ansible_user
18       copy: # copy module om een bestand te kopiëren
19         dest: /etc/sudoers.d/ansible_user
20         content: "ansible_user ALL=(ALL) NOPASSWD: ALL\n" # Voeg de regel toe aan het bestand "/etc/sudoers.d/ansible_user"
21         mode: '0440'
22
23
```

Playbook configuratie

Task

VB-CODE ANSIBLE PLAYBOOK

PYTHON INSTALLEREN

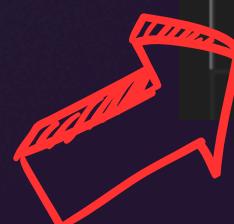
```
1  ---
2  - hosts: all # Alle servers in het inventory bestand
3    become: yes # Voer taken uit met verhoogde rechten, maar alleen waar nodig
4    become_user: ansible_user # Specificeer een niet-root gebruiker met sudo rechten
5    tasks:
6      - name: Update apt cache # Update de apt cache
7        apt:
8          update_cache: yes
9        become: yes # Alleen sudo-rechten gebruiken voor deze specifieke taak
10
11     - name: Install Python 3 # Installeer Python 3
12       apt:
13         name: python3
14         state: present
15       become: yes # sudo-rechten vereist voor installatie
16
17     - name: Verify Python installation # Verifieer de Python-installatie
18       command: python3 --version
19       register: python_version # Sla de output van het commando op in python_version
20       become: no # Geen sudo-rechten nodig voor het uitvoeren van dit commando
21
22     - name: Display Python version # Toon de Python-versie
23       debug:
24         msg: "Python versie: {{ python_version.stdout }}" # Toon de output van python3 --version
25
```

FOUTAFHANDELING PLAYBOOKS

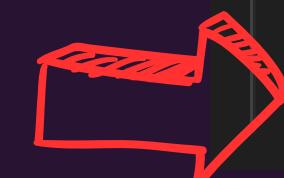
```
- name: Voorbeeld ignore_errors
  command: # comando
  ignore_errors: yes
```



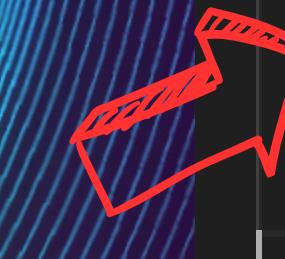
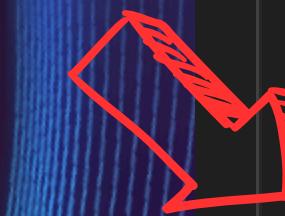
```
- name: Voorbeeld "when"
  apt:
    name: vim
    state: present # check of vim geïnstalleerd is
  when: ansible_distribution == 'Ubuntu' # voer task alleen uit als OS Ubuntu is
```



```
- name: Controleer een bestand
  stat:
    path: # pad naar bestand
  register: result # sla de resultaten op
  failed_when: result.stat.exists == false # task faalt als bestand niet bestaat
```



```
- block:
  - name: Probeer een foutgevoelige taak
    command: /bin/false
  rescue: # uitvoeren als task in het block faalt
    - name: Herstel na fout
      command: echo "Taak mislukt, herstel uitgevoerd"
  always: # altijd uitvoeren, ongeacht succes of mislukking
    - name: Print "Altijd uitvoeren"
      command: echo "Altijd uitvoeren"
```



BEST PRACTICES

```
- name: Installeer Python
  apt:
    name: python3
    state: present
  tags: python-installatie
  comment: "Task om python3 te installeren"
```

- Beschrijvende namen geven
- Variabelen gebruiken ipv hardcoding
- Ansible rollen
- Tags: om bepaalde task hergebruiken
- Comments

ANSIBLE GEBRUIKEN

[HTTPS://GITHUB.COM/AFKIR-AYOUB/DEVOPS-ANSIBLE-2324](https://github.com/afkir-ayoub/devops-ansible-2324)

INSTALLEER VIRTUALBOX + UBUNTU SERVER (ZIE GITHUB)

ANSIBLE GEBRUIKEN

MAAK 2 VM'S AAN

Name and Operating System

Name: Server1

Folder: C:\Users\Ayoub\VirtualBox VMs

ISO Image: C:\Users\Ayoub\Downloads\ubuntu-24.04.1-live-server-amd64.iso

Edition:

Type: Linux

Subtype: Ubuntu

Version: Ubuntu (64-bit)

Skip Unattended Installation

Unattended Install

Username and Password

Username: Server1

Password: password

Repeat Password: password

Additional Options

Product Key: #####-#####-#####-#####-#####

Hostname: Server1

Domain Name: server1.virtualbox.org

Install in Background

Guest Additions

Guest Additions ISO: <not selected>

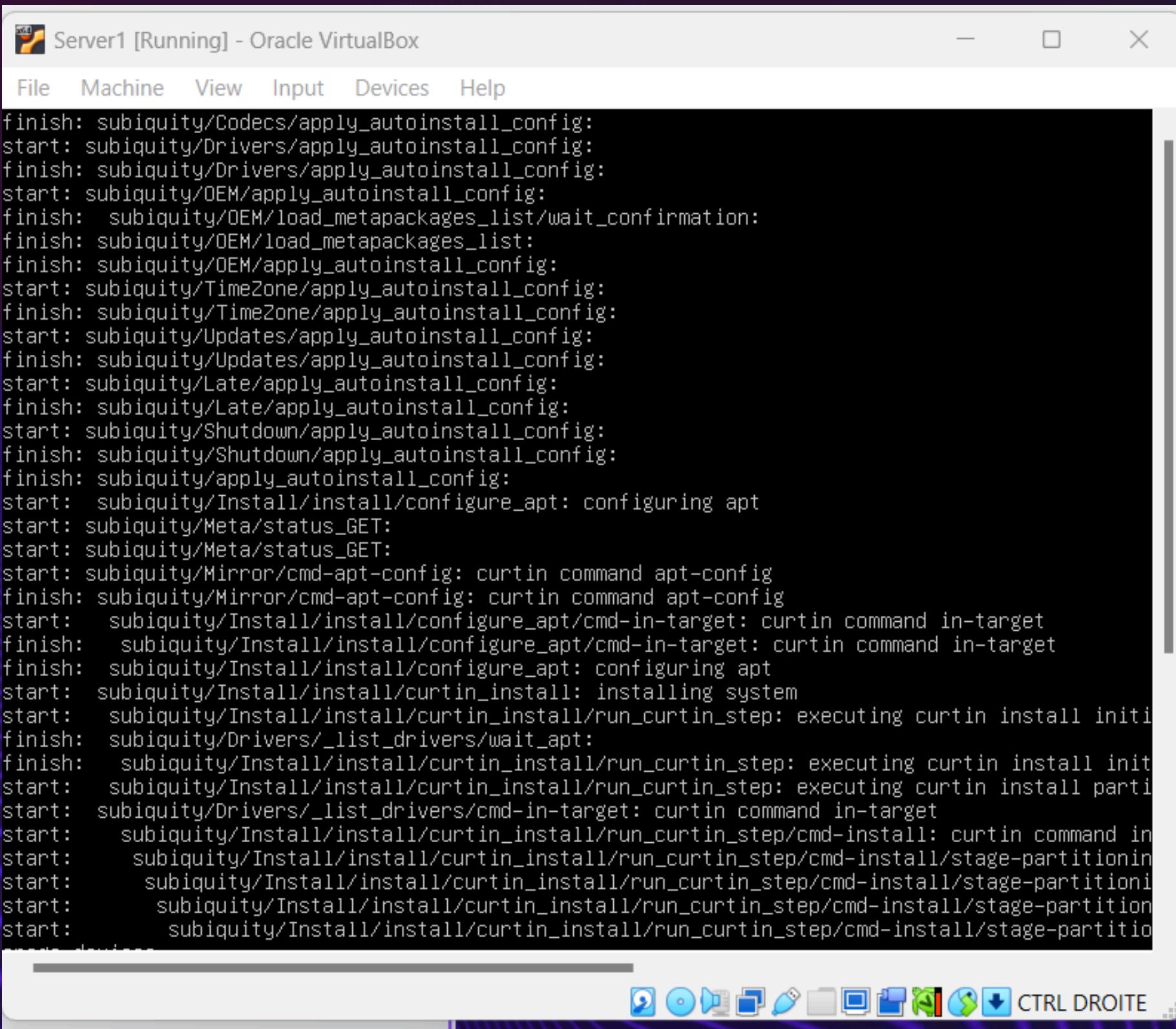
Unattended Install

Hardware

Hard Disk

ANSIBLE GEBRUIKEN

WACHT DAT DE VM OPSTART &
LOGIN



The screenshot shows a terminal window within Oracle VirtualBox. The title bar reads "Server1 [Running] - Oracle VirtualBox". The menu bar includes "File", "Machine", "View", "Input", "Devices", and "Help". The main terminal area displays a log of Ansible tasks:

```
finish: subiquity/Codecs/apply_autoinstall_config:  
start: subiquity/Drivers/apply_autoinstall_config:  
finish: subiquity/Drivers/apply_autoinstall_config:  
start: subiquity/OEM/apply_autoinstall_config:  
finish: subiquity/OEM/load_metapackages_list/wait_confirmation:  
finish: subiquity/OEM/load_metapackages_list:  
finish: subiquity/OEM/apply_autoinstall_config:  
start: subiquity/TimeZone/apply_autoinstall_config:  
finish: subiquity/TimeZone/apply_autoinstall_config:  
start: subiquity/Updates/apply_autoinstall_config:  
finish: subiquity/Updates/apply_autoinstall_config:  
start: subiquity/Late/apply_autoinstall_config:  
finish: subiquity/Late/apply_autoinstall_config:  
start: subiquity/Shutdown/apply_autoinstall_config:  
finish: subiquity/Shutdown/apply_autoinstall_config:  
finish: subiquity/apply_autoinstall_config:  
start: subiquity/Install/install/configure_apt: configuring apt  
start: subiquity/Meta/status_GET:  
start: subiquity/Meta/status_GET:  
start: subiquity/Mirror/cmd-apt-config: curtin command apt-config  
finish: subiquity/Mirror/cmd-apt-config: curtin command apt-config  
start: subiquity/Install/install/configure_apt/cmd-in-target: curtin command in-target  
finish: subiquity/Install/install/configure_apt/cmd-in-target: curtin command in-target  
finish: subiquity/Install/install/configure_apt: configuring apt  
start: subiquity/Install/install/curtin_install: installing system  
start: subiquity/Install/install/curtin_install/run_curtin_step: executing curtin install init  
finish: subiquity/Drivers/_list_drivers/wait_apt:  
finish: subiquity/Install/install/curtin_install/run_curtin_step: executing curtin install init  
start: subiquity/Install/install/curtin_install/run_curtin_step: executing curtin install parti  
start: subiquity/Drivers/_list_drivers/cmd-in-target: curtin command in-target  
start: subiquity/Install/install/curtin_install/run_curtin_step/cmd-install: curtin command in  
start: subiquity/Install/install/curtin_install/run_curtin_step/cmd-install/stage-partitionin  
start: subiquity/Install/install/curtin_install/run_curtin_step/cmd-install/stage-partitioni  
start: subiquity/Install/install/curtin_install/run_curtin_step/cmd-install/stage-partition  
start: subiquity/Install/install/curtin_install/run_curtin_step/cmd-install/stage-partitio
```

At the bottom of the terminal window, there is a toolbar with various icons.

QWERTY!

ANSIBLE GEBRUIKEN

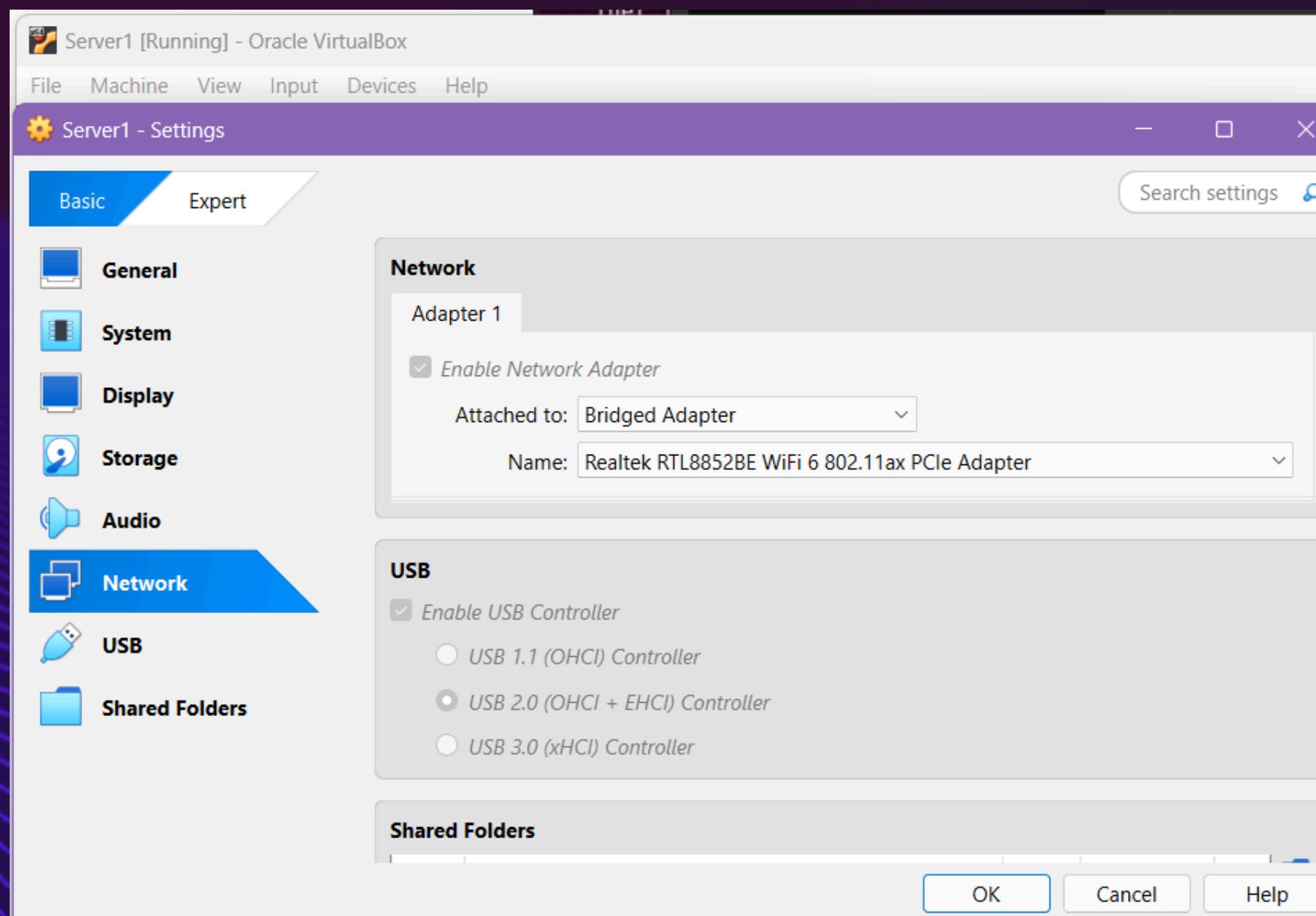
INSTALLEER SSH

```
sudo apt install openssh-server
```

```
Server1@Server1:~$ systemctl enable ssh --now
Synchronizing state of ssh.service with SysV service script with /usr/lib/systemd/systemd-sysv-install.
Executing: /usr/lib/systemd/systemd-sysv-install enable ssh
==== AUTHENTICATING FOR org.freedesktop.systemd1.reload-daemon ====
Authentication is required to reload the systemd state.
Authenticating as: Server1
Password:
==== AUTHENTICATION COMPLETE ====
==== AUTHENTICATING FOR org.freedesktop.systemd1.reload-daemon ====
Authentication is required to reload the systemd state.
Authenticating as: Server1
Password:
==== AUTHENTICATION COMPLETE ====
==== AUTHENTICATING FOR org.freedesktop.systemd1.manage-unit-files ====
Authentication is required to manage system service or unit files.
Authenticating as: Server1
Password:
==== AUTHENTICATION COMPLETE ====
Created symlink /etc/systemd/system/sshd.service → /usr/lib/systemd/system/ssh.service.
Created symlink /etc/systemd/system/multi-user.target.wants/ssh.service → /usr/lib/systemd/system/ssh.service.
Server1@Server1:~$
```

ANSIBLE GEBRUIKEN

KRIJG EEN IP VAN DE SERVERS



```
Server1@Server1:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
        inet 127.0.0.1/8 scope host lo
            valid_lft forever preferred_lft forever
        inet6 ::1/128 scope host noprefixroute
            valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_
    link/ether 08:00:27:12:30:24 brd ff:ff:ff:ff:ff:ff
        inet 192.168.68.63/22 metric 100 brd 192.168.71.255 scope global
            valid_lft 7178sec preferred_lft 7178sec
        inet6 fe80::a00:27ff:fe12:3024/64 scope link
            valid_lft forever preferred_lft forever
Server1@Server1:~$
```

```
C:\Users\Ayoub>ping 192.168.68.63

Pinging 192.168.68.63 with 32 bytes of data:
Reply from 192.168.68.63: bytes=32 time=1ms TTL=64
Reply from 192.168.68.63: bytes=32 time<1ms TTL=64
Reply from 192.168.68.63: bytes=32 time<1ms TTL=64
Reply from 192.168.68.63: bytes=32 time<1ms TTL=64

Ping statistics for 192.168.68.63:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 1ms, Average = 0ms
```

ANSIBLE GEBRUIKEN

TEST: CONNECT NAAR SERVERS VIA SSH

```
C:\Users\Ayoub>ssh Server1@192.168.68.63
Server1@192.168.68.63's password:
Welcome to Ubuntu 24.04.1 LTS (GNU/Linux 6.8.0-45-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/pro

System information as of Sun Oct 13 04:47:59 PM UTC 2024

      System load:  0.0          Processes:          97
      Usage of /:   9.6% of 24.44GB  Users logged in:     1
      Memory usage: 9%           IPv4 address for enp0s3: 192.168.68.63
      Swap usage:   0%

Expanded Security Maintenance for Applications is not enabled.

35 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

Server1@Server1:~$
```

```
C:\Users\Ayoub>ssh Server2@192.168.68.64
The authenticity of host '192.168.68.64 (192.168.68.64)' can't be established.
ED25519 key fingerprint is SHA256:L/1L3C8qeJFnJccGDS09c/nedhGmld5VWFpjuZW///I.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.68.64' (ED25519) to the list of known hosts.
Server2@192.168.68.64's password:
Welcome to Ubuntu 24.04.1 LTS (GNU/Linux 6.8.0-45-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/pro

System information as of Sun Oct 13 04:51:40 PM UTC 2024

      System load:  0.04         Processes:          98
      Usage of /:   9.6% of 24.44GB  Users logged in:     1
      Memory usage: 9%           IPv4 address for enp0s3: 192.168.68.64
      Swap usage:   0%

Expanded Security Maintenance for Applications is not enabled.

35 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

Server2@Server2:~$
```

ANSIBLE GEBRUIKEN

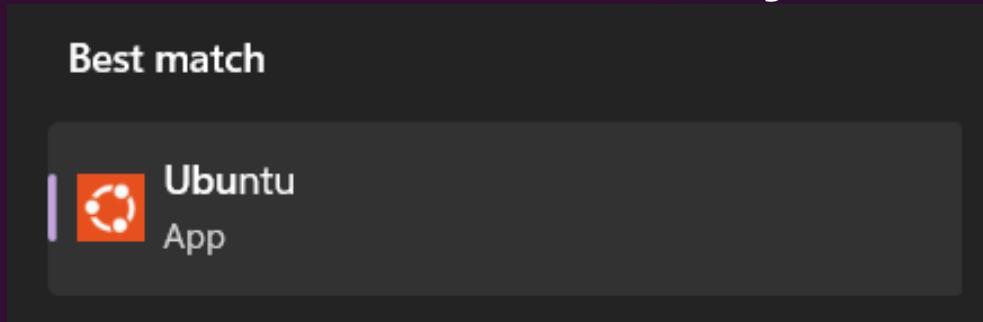
WSL INSTALLEREN OP WINDOWS

CMD IN ADMIN MODE

```
C:\Windows\System32>wsl --install
Installing: Ubuntu
Ubuntu has been installed.
Launching Ubuntu...
Installing, this may take a few minutes...
Please create a default UNIX user account. The username does not need to match your Windows username.
For more information visit: https://aka.ms/wslusers
Enter new UNIX username: ayoub
New password:
Retype new password:
passwd: password updated successfully
Installation successful!
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.
C:
Welcome to Ubuntu 24.04.1 LTS (GNU/Linux 5.15.153.1-microsoft-standard-WSL2 x86_64)
```

ANSIBLE GEBRUIKEN

ANSIBLE INSTALLEREN (OP JOUW LAPTOP)



```
ayoub@DESKTOP-28N60FK: ~ $ sudo apt-get install ansible  
[sudo] password for ayoub:
```

ANSIBLE GEBRUIKEN

PASSWORDLESS SSH CONFIGUREREN (OP JOUW LAPTOP)

```
ayoub@DESKTOP-28N60FK:~$ ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/home/ayoub/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/ayoub/.ssh/id_rsa
Your public key has been saved in /home/ayoub/.ssh/id_rsa.pub
-----
```

```
ayoub@DESKTOP-28N60FK:~$ ssh-copy-id Server1@192.168.68.63
```

IP VERANDEREN

```
ayoub@DESKTOP-28N60FK:~$ ssh-copy-id Server2@192.168.68.64
```

ANSIBLE GEBRUIKEN

TEST: WERKT SSH ZONDER PASSWORD?

IP VERANDEREN

```
ayoub@DESKTOP-28N60FK:~$ ssh Server1@192.168.68.63  
Welcome to Ubuntu 24.04.1 LTS (GNU/Linux 6.8.0-45-generic x86_64)
```

```
ayoub@DESKTOP-28N60FK:~$ ssh Server2@192.168.68.64  
Welcome to Ubuntu 24.04.1 LTS (GNU/Linux 6.8.0-45-generic x86_64)
```

ANSIBLE GEBRUIKEN

ANSIBLE PROJECT MAKEN

```
| ayoub@DESKTOP-28N60FK:~$ mkdir ansible-voorbeeld  
ayoub@DESKTOP-28N60FK:~$ cd ansible-voorbeeld/  
ayoub@DESKTOP-28N60FK:~/ansible-voorbeeld$ |
```

ANSIBLE GEBRUIKEN

INVENTORY.YML FILE SCHRIJVEN

```
ayoub@DESKTOP-28N60FK:~/ansible-voorbeeld$ nano inventory.yml
```

```
GNU nano 7.2                                         inventory.yml *
all:
  hosts:
    s1: # naam van de 1ste VM
      ansible_host: 192.168.68.63 # IP-adres van de 1ste VM
      ansible_user: Server1 # gebruikersnaam van de 1ste VM
    s2: # naam van de 2de VM
      ansible_host: 192.168.68.64 # IP-adres van de 2de VM
      ansible_user: Server2 # gebruikersnaam van de 2de VM

^G Help   ^O Write Out  ^W Where Is   ^K Cut      ^T Execute   ^C Location   M-U Undo
^X Exit   ^R Read File  ^\ Replace   ^U Paste    ^J Justify   ^/ Go To Line M-E Redo
```

IP VERANDEREN

ANSIBLE GEBRUIKEN

PLAYBOOK.YML MAKEN

```
ayoub@DESKTOP-28N60FK:~/ansible-voorbeeld$ nano playbook.yml
```

```
GNU nano 7.2                                         playbook.yml *
```

```
---
```

```
- name: Installeer Python 3 op alle VM's
hosts: all # alle VM's in de inventory
become: yes
```

```
tasks:
- name: Update de apt cache
  apt:
    update_cache: yes
```

```
- name: Installeer Python 3
  apt:
    name: python3
    state: present # installeert Python 3 als het nog niet geïnstalleerd is
```

```
^G Help      ^O Write Out   ^W Where Is     ^K Cut        ^T Execute   ^C Location   M-U Undo
^X Exit      ^R Read File   ^\ Replace      ^U Paste      ^J Justify   ^/ Go To Line M-E Redo
```

ANSIBLE GEBRUIKEN

ANSIBLE RUNNEN

```
ayoub@DESKTOP-28N60FK:~/ansible-voorbeeld$ ansible-playbook -i inventory.yml playbook.yml --ask-become-pass
BECOME password:

PLAY [Installeer Python 3 op alle VM's] ****
TASK [Gathering Facts] ****
ok: [s1]
ok: [s2]

TASK [Update de apt cache] ****
changed: [s1]
changed: [s2]

TASK [Installeer Python 3] ****
ok: [s1]
ok: [s2]

PLAY RECAP ****
s1                  : ok=3    changed=1    unreachable=0    failed=0     skipped=0    rescued=0    ignor
ed=0
s2                  : ok=3    changed=1    unreachable=0    failed=0     skipped=0    rescued=0    ignor
ed=0

ayoub@DESKTOP-28N60FK:~/ansible-voorbeeld$ █
```

THANK YOU
FOR YOUR ATTENTION