

Homework Sesi 14 Bootcamp DevOps Engineer DigitalSkola

SOAL

1. Siapkan 3 server untuk ansible di virtualbox, 1 sebagai Control Node, 2 Managed Nodes.
2. Install nginx menggunakan ansible

Sudah disiapkan 3 server di GCP yaitu **server1**, **server2**, dan **server3**. Ditentukan bahwa **server1** menjadi control node yang akan diinstal ansible dan menjalankan peng-install-an nginx ke **server2** dan **server3**.

VM instances

Filter

Enter property name or value

<input type="checkbox"/>	Status	Name ↑	Zone	Recommendations	In use by	Internal IP	External IP	Connect
<input type="checkbox"/>	✓	server1	asia-southeast1-b			10.148.0.2 (nic0)	34.142.140.71 (nic0)	SSH ▾ ⋮
<input type="checkbox"/>	✓	server2	asia-southeast1-b	💡 Save \$2 / mo		10.148.0.3 (nic0)	34.124.156.51 (nic0)	SSH ▾ ⋮
<input type="checkbox"/>	✓	server3	asia-southeast1-b			10.148.0.4 (nic0)	34.124.158.219 (nic0)	SSH ▾ ⋮

SSH tanpa password telah disetting sehingga **server1** dapat mengakses **server2** dan **server3** menggunakan ssh-key. Sehingga pada homework ini dapat langsung dijelaskan step by step untuk menginstall nginx menggunakan ansible.

Berikut langkah-langkah install nginx di **server2** dan **server3** menggunakan ansible:

1. Instal Ansible di server1 (Control Node)

Karena server1 akan menjadi control node, server1 perlu diinstal Ansible. Berikut cara menginstal Ansible di Ubuntu server:

- **Update daftar paket:** Buka terminal di server1 dan jalankan perintah **sudo apt update** untuk memastikan daftar paket terbaru.

```
ubuntu_admin@server1:~$ sudo apt update
```
- **Instal Ansible:** Setelah daftar paket diperbarui, instal Ansible dengan perintah **sudo apt install ansible**

```
ubuntu_admin@server1:~$ sudo apt install ansible
```
- **Verifikasi Instalasi Ansible:** Setelah instalasi selesai, verifikasi Ansible telah terinstal dengan benar dengan memeriksa versi Ansible dengan perintah **ansible --version**. Output yang ditampilkan adalah informasi versi Ansible, lokasi konfigurasi, dan lain-lain.

2. Konfigurasi Inventory (inventory.ini atau hosts) di Server1

Buat file inventory di **server1** untuk mendefinisikan server target (**server2** dan **server3**). File ini bisa bernama inventory.ini atau hosts. Biasanya, file ini diletakkan di direktori yang sama dengan playbook. Isi file **inventory.ini** untuk homework ini adalah sebagai berikut:

```
[webservers]
server2 ansible_host=34.124.156.51
server3 ansible_host=34.124.158.219

[webservers:vars]
ansible_user=ubuntu_admin
ansible_ssh_private_key_file=/home/ubuntu_admin/.ssh/to_server2_key
```

Penjelasan:

- **[webservers]** → Nama grup server yang dikelola.
- **server2 ansible_host=34.124.156.51** dan **server3 ansible_host=34.124.158.219** → Definisi host dalam grup webservers.
- **[webservers:vars]** → Variabel yang berlaku untuk semua host dalam grup webservers.
 - **ansible_user=ubuntu_admin** → Username SSH untuk login ke server2 dan server3.
 - **ansible_ssh_private_key_file=/home/ubuntu_admin/.ssh/to_server2_key** → Path ke private key SSH. Karena sudah konfigurasi SSH tanpa password, Ansible akan pakai key ini untuk autentikasi ke **server2** dan **server3**. Penamaan file key SSH yang seakan hanya untuk ke **server2** tidak masalah karena hanya sebuah penamaan. Pelajaran yang bisa diambil adalah sebaiknya beri nama file yang lebih umum atau default saja.

File **inventory.ini** akan dibuat di dalam direktori **ansible_nginx**. Langkah yang dilakukan adalah sebagai berikut:

```
ubuntu_admin@server1:~$ mkdir ansible_nginx
ubuntu_admin@server1:~$ cd ansible_nginx/
ubuntu_admin@server1:~/ansible_nginx$ nano inventory.ini
```

Di dalam file **inventory.ini** isi konfigurasi tersebut.

```
ubuntu_admin@server1:~/ansible_nginx$ cat inventory.ini
[webservers]
server2 ansible_host=34.124.156.51
server3 ansible_host=34.124.158.219

[webservers:vars]
ansible_user=ubuntu_admin
ansible_ssh_private_key_file=/home/ubuntu_admin/.ssh/to_server2_key
```

3. Buat Playbook (nginx_install.yml) di Server1

Buat file playbook **nginx_install.yml** di **server1**, di direktori yang sama dengan **inventory.ini** yaitu **ansible_nginx/**. Isi file **nginx_install.yml** adalah sebagai berikut:

```
---
- hosts: webservers
  become: yes
  tasks:
    - name: Update apt cache
      apt:
        update_cache: yes

    - name: Install nginx
      apt:
        name: nginx
        state: present

    - name: Start and enable nginx
      service:
        name: nginx
        state: started
        enabled: yes
```

Penjelasan Playbook:

- **---** → Awal file YAML.
- **- hosts: webservers** → Playbook dijalankan pada host di grup webservers (dari inventory).
- **become: yes** → Izinkan Ansible jalankan task dengan hak akses sudo di server target (perlu untuk instalasi paket dan service).
- **tasks:** → Daftar task yang dijalankan berurutan.
 - **- name: Update apt cache** → Task pertama: update cache APT.
 - **apt: update_cache: yes** → Modul apt untuk kelola paket di Debian/Ubuntu. update_cache: yes → jalankan apt update.
 - **- name: Install nginx** → Task kedua: instal paket Nginx.
 - **apt: name: nginx state: present** → Modul apt lagi.
 - **name: nginx** → Paket yang diinstal adalah nginx.
 - **state: present** → Pastikan nginx terinstal. Jika sudah, tidak ada aksi. Jika belum, akan diinstal.
 - **- name: Start and enable nginx** → Task ketiga: pastikan service Nginx jalan dan aktif saat boot.

- **service: name: nginx state: started enabled: yes** → Modul service untuk kelola service.
 - **name: nginx** → Service yang dikelola nginx.
 - **state: started** → Pastikan service nginx berjalan. Jika belum, dijalankan. Jika sudah, tidak ada aksi.
 - **enabled: yes** → Pastikan nginx otomatis jalan saat sistem reboot.

```
ubuntu_admin@server1:~/ansible_nginx$ nano nginx_install.yml
ubuntu_admin@server1:~/ansible_nginx$ cat nginx_install.yml
---
- hosts: webserver
  become: yes
  tasks:
    - name: Update apt cache
      apt:
        update_cache: yes

    - name: Install nginx
      apt:
        name: nginx
        state: present

    - name: Start and enable nginx
      service:
        name: nginx
        state: started
        enabled: yes
ubuntu_admin@server1:~/ansible_nginx$
```

4. Jalankan Playbook dari Server1

Setelah buat **inventory.ini** dan **nginx_install.yml** di **server1**, jalankan playbook dari **server1** dengan perintah **ansible-playbook -i inventory.ini nginx_install.yml**.

Penjelasan Perintah:

- **ansible-playbook**: Perintah untuk jalankan playbook Ansible.
- **-i inventory.ini**: Opsi **-i** atau **--inventory** untuk tentukan file inventory. Jika inventory bernama **hosts** dan di direktori sama, bisa pakai **-i hosts** atau hilangkan **-i** (karena **hosts** nama default inventory).
- **nginx_install.yml**: Nama file playbook yang dijalankan.

Pastikan perintah ini dijalankan dari direktori tempat menyimpan **inventory.ini** dan **nginx_install.yml** yaitu di direktori **ansible_nginx/**. Ansible-playbook yang berhasil dijalankan akan menampilkan output seperti gambar berikut.

```

ubuntu_admin@server1:~/ansible_nginx$ ansible-playbook -i inventory.ini nginx_install.yml

PLAY [webservers] *****

TASK [Gathering Facts] *****
The authenticity of host '34.124.156.51 (34.124.156.51)' can't be established.
ECDSA key fingerprint is SHA256:csU9Iu5Fd8J3qUiY4yf+7YJT0cHWWhs5+HSmILDwLN9Q.
Are you sure you want to continue connecting (yes/no/[fingerprint])? ok: [server3]
yes
ok: [server2]

TASK [Update apt cache] *****
changed: [server3]
changed: [server2]

TASK [Install nginx] *****
changed: [server3]
changed: [server2]

TASK [Start and enable nginx] *****
ok: [server3]
ok: [server2]

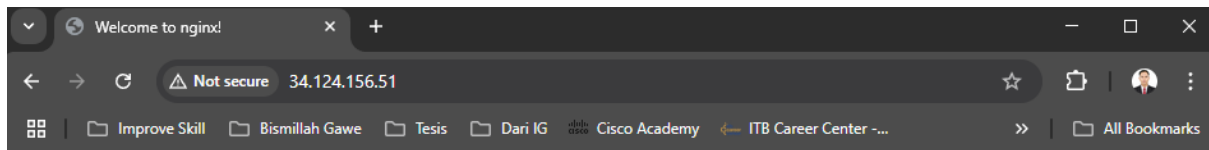
PLAY RECAP *****
server2      : ok=4    changed=2    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
server3      : ok=4    changed=2    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

```

5. Verifikasi Instalasi Nginx di Server2 dan Server3

Setelah playbook berhasil dijalankan, verifikasi instalasi Nginx:

1. **Akses Browser:** Buka browser, arahkan ke IP **server2** dan **server3** yang firewall-nya sudah disetup mengizinkan akses HTTP (port 80). Halaman default Nginx akan muncul jika berhasil.

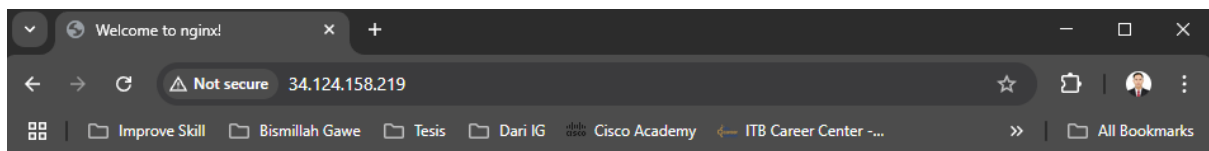


Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Thank you for using nginx.



Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Thank you for using nginx.