## Tugas:

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

1. Buat 3 server dan gunakan 1 key untuk ketiganya. Setelah itu simpan key di tempat yang aman dan chmod 400

| | | | |
|---|---|---|---|
| mysql-node1 | i-06f17cd9e62707f56 | Running | t3.micro |
| ansible-control | i-063dfc02af8c179b7 | Running | t3.micro |
| mysql-node2 | i-06fd4a179e5dc78ce | Running | t3.micro |

chmod 400

```
Valin@DESKTOP-VMJSNLD MINGW64 ~
$ cd Downloads/

Valin@DESKTOP-VMJSNLD MINGW64 ~/Downloads
$ chmod 400 valin2.pem
```

2. SSH ke control node :
ssh -i valin2.pem ubuntu@52.221.228.123

3. Didalam control node, generate SSH key
ssh-keygen -t ed25519 -N "" -f ~/.ssh/id_ed25519

4. Copy public key ke kedua managed nodes:
ssh-copy-id -i ~/.ssh/id_ed25519.pub ubuntu@13.212.186.110
ssh-copy-id -i ~/.ssh/id_ed25519.pub ubuntu@47.129.238.137

dan tenyata permission denied, jadi move to the next step.

```
ubuntu@ip-172-31-35-247:~$ ssh-copy-id -i ~/.ssh/id_ed25519.pub ubuntu@13.212.18
6.110
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/ubuntu/.ssh
/id_ed25519.pub"
The authenticity of host '13.212.186.110 (13.212.186.110)' can't be established.
ED25519 key fingerprint is SHA256:wq/yaalnmwHiwoU/SGxC+mxK7JQqdFNjNPok/4QEdVY.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter
out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompt
ed now it is to install the new keys
ubuntu@13.212.186.110: Permission denied (publickey).
```

5. Tampilkan public key di control node nya dan catat.
   cat ~/.ssh/id_ed25519.pub

6. SSH ke managed node dari lokal laptop (untuk managednode1 dan managednode2)
   ssh -i valin2.pem ubuntu@13.212.186.110
   ssh -i valin2.pem ubuntu@47.129.238.137

7. Setelah masuk, copy code dibawah ini
   mkdir -p ~/.ssh
   chmod 700 ~/.ssh
   echo " ssh-ed25519
   AAAAC3NzaC1lZDI1NTE5AAAAINKTl27+kWmTAgoB3ZjmD/8Ll01/3FjUKqg4co/p7
   mRR ubuntu@ip-172-31-35-247" >> ~/.ssh/authorized_keys
   chmod 600 ~/.ssh/authorized_keys

8. Test dari control node (tanpa pakai .pem lagi)
   ssh -i valin2.pem ubuntu@IP_CONTROL_NODE
   ssh ubuntu@13.212.186.110
   ssh ubuntu@47.129.238.137

   Kalau berhasil, lanjut ke step selanjutnya mengenai Ansible

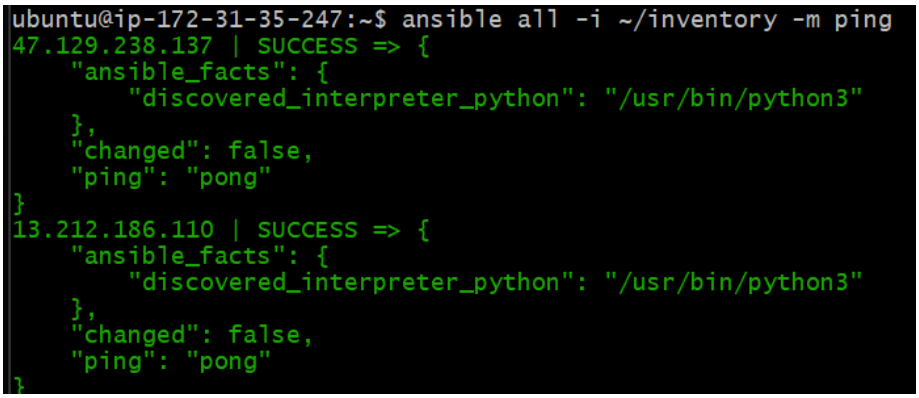
9. Di control node:
   sudo apt update
   sudo apt install ansible -y

   Buat inventory sederhana dulu untuk test
   **nano ~/inventory**

   Copy ini :
   **[node1]**
   **13.212.186.110 ansible_user=ubuntu**

   **[node2]**
   **47.129.238.137 ansible_user=ubuntu**

   Lalu tes ping:
   **ansible all -i ~/inventory -m ping**




Kalau sudah pong, lanjut install MySQL pakai playbook (karna berarti Ansible (control node) sudah terkoneksi dengan baik ke managed nodes)

Notes : langkah2 selanjutnya dilakukan di control node.

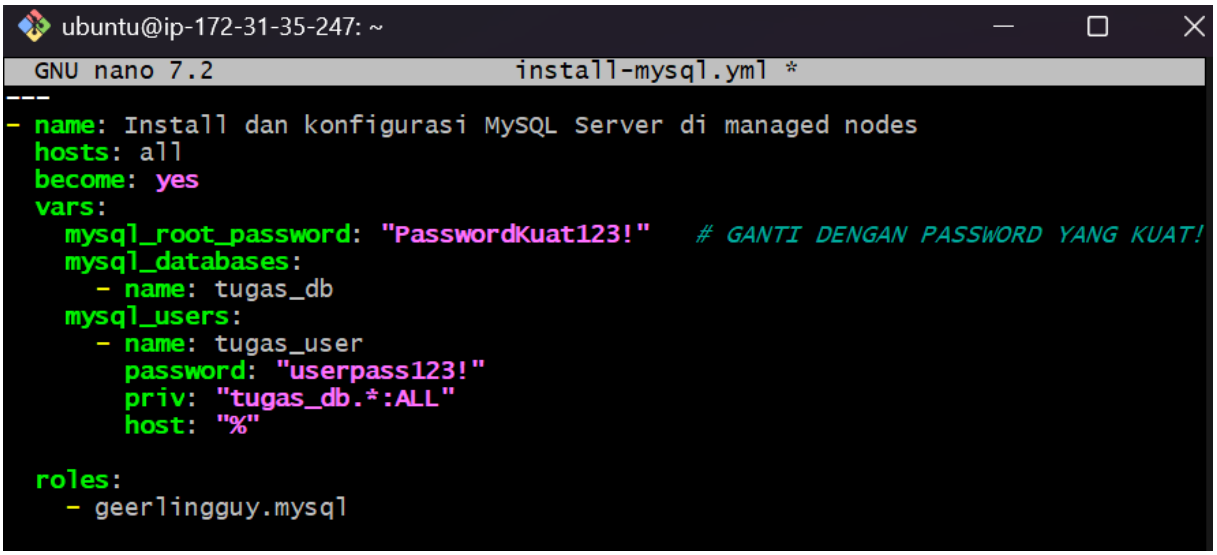
10. Install collection yang dibutuhkan
    ansible-galaxy collection install community.mysql

11. Install role MySQL yang bagus dan ter-maintain
    Role geerlingguy.mysql adalah yang paling populer dan stabil untuk Ubuntu.
    ansible-galaxy install geerlingguy.mysql

12. Buat file playbook
    Buat file baru bernama install-mysql.yml di home kamu:
    nano install-mysql.yml

    Paste isi playbook ini :

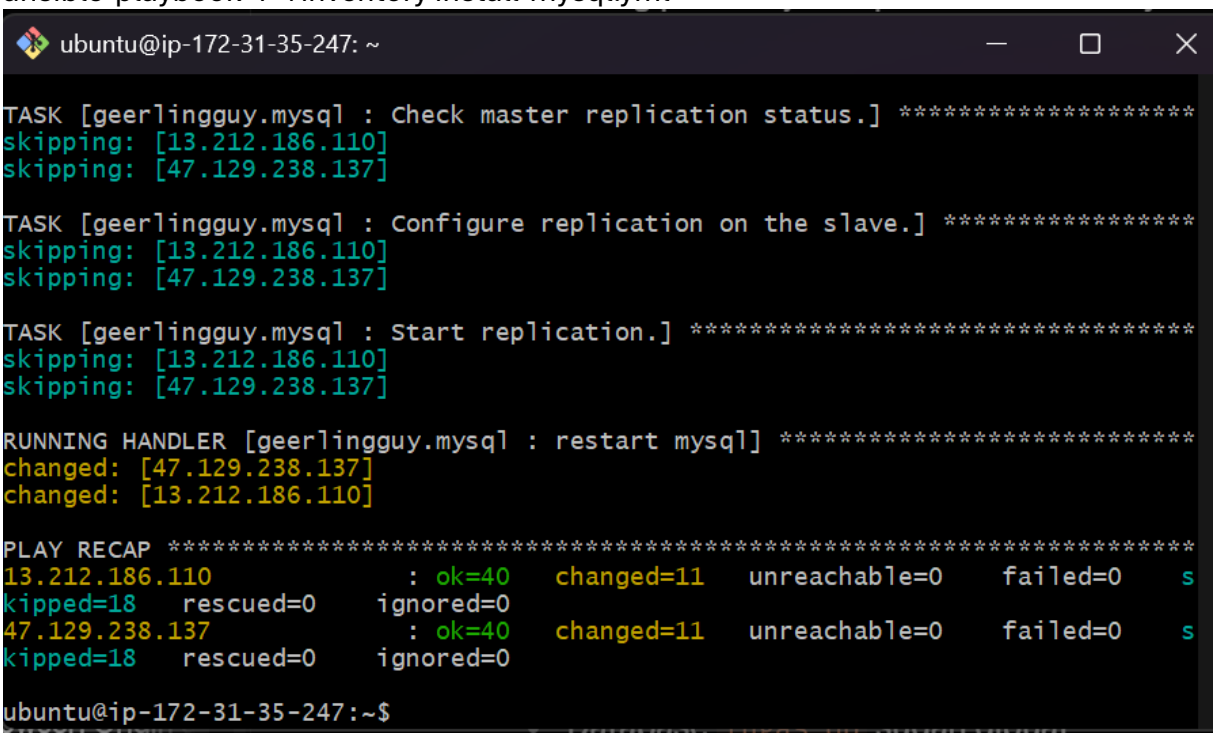
```
---
- name: Install dan konfigurasi MySQL Server di managed nodes
 hosts: all
 become: yes
 vars:
  mysql_root_password: "PasswordKuat123!"   # GANTI DENGAN PASSWORD YANG KUAT!
  mysql_databases:
   - name: tugas_db
  mysql_users:
   - name: tugas_user
     password: "userpass123!"
     priv: "tugas_db.*:ALL"
     host: "%"

 roles:
  - geerlingguy.mysql
```




Notes : hosts: all --- tapi kalo mau sesuaiin group namenya boleh juga, pokoknya nama groupnyya disesuaikan dengan y gada di dalam **nano ~/inventory**

Sedikit penjelasan :
hosts: dbservers → pakai group yang ada di inventory kita
mysql_root_password → password root MySQL (wajib diganti!)
Otomatis buat database tugas_db dan user tugas_user dengan akses full ke database itu
Role ini juga otomatis jalankan mysql_secure_installation (hapus anonymous user, test db, dll)

13. Jalankan playbook
Kalau inventory kamu masih di ~/inventory, jalankan dengan:
ansible-playbook -i ~/inventory install-mysql.yml



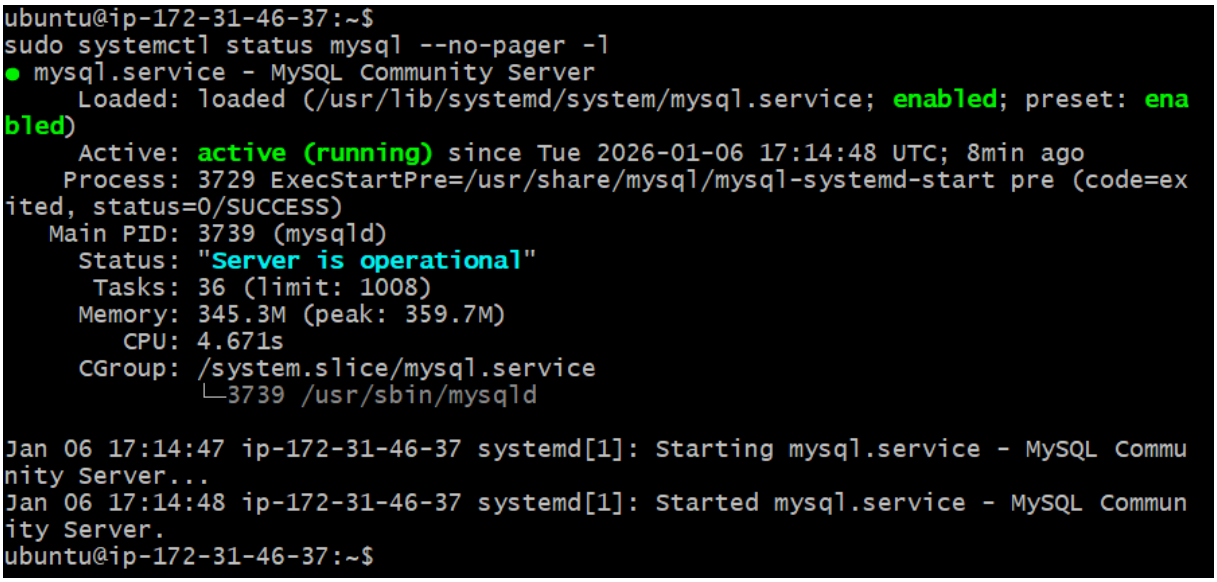
14. Verifikasi akhir

    Cek status MySQL di kedua node (SSH ke masing2 managed node):
    **sudo systemctl status mysql --no-pager -l**

    Kalo gamau SSH ke masing2 managed node, bisa paste ini di control node:
    **ansible all -i ~/inventory -m shell -a "sudo systemctl status mysql --no-pager -l" –become**

    Notes : all diatas, disesuaikan dengan nama groupnya, lihat di step nomor 12

    Screenshot SSH ke masing2 managed node:




    Screenshot dari control node: