**BÁO CÁO TUẦN 2**

*Nguyễn Ngọc Minh Trí*

**MỤC LỤC**

[**MỤC LỤC** 2](#_Toc190685422)

[1. NAT 3](#_Toc190685423)

[2. PORT FORWARDING 8](#_Toc190685424)

[3. ROUTING 16](#_Toc190685425)

## **NAT**

*Username VM1: mntri*

*Username VM2: minhtri*

Mỗi bạn tạo 2 VM:

**VM1:** 1/1/20/vmbr0/vmbr1

• IP WAN: 45.122.223.120

• IP LAN: 10.0.x.1/24

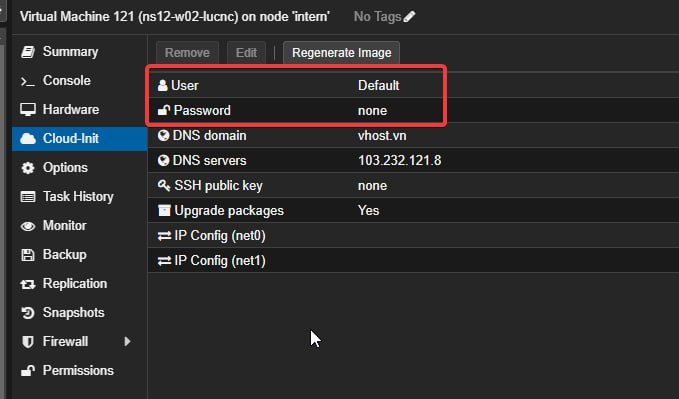
**VM2:** 1/1/20/vmbr1

• IP LAN: 10.0.x.2/24 - GW: 10.0.x.1

*Yêu cầu:*

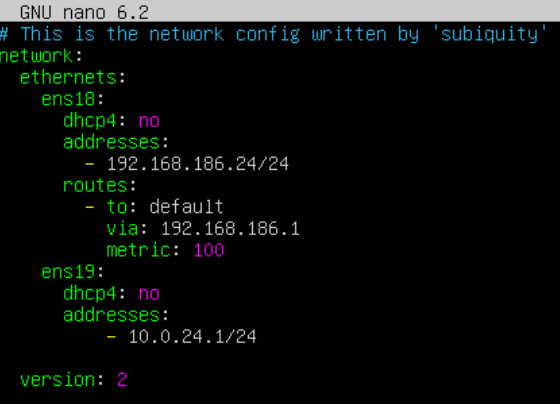
* *Giá trị của x: Lấy theo giá trị cuối của IP card mạng 1 ra Internet. Ví dụ 192.168.186.24 thì x là 24*
* *Sử dụng iptables cấu hình VM1 NAT masquerade để VM2 có thể đi ra internet được thông qua VM1.*
* *Reboot VM1 và sau khi boot vào OS thì VM2 vẫn có thể đi ra internet được thông qua VM1*

Đầu tiên, chỉnh username & password máy **VM2** ở phần Cloud-Init:



Ở máy **VM1**:

* Mở file cấu hình netplan:   
  **sudo nano /etc/netplan/00-installer-config.yaml**

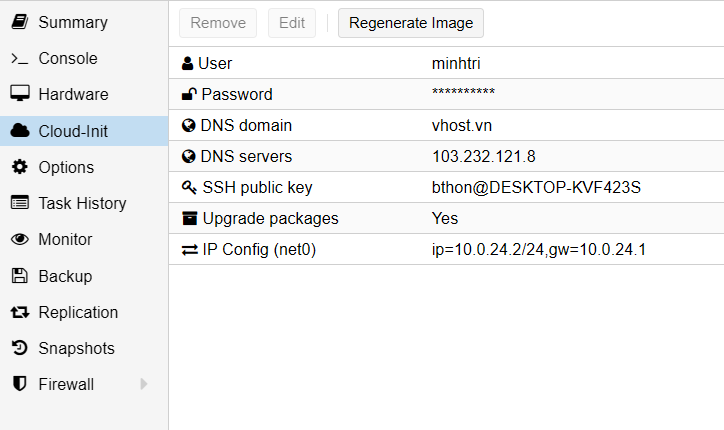


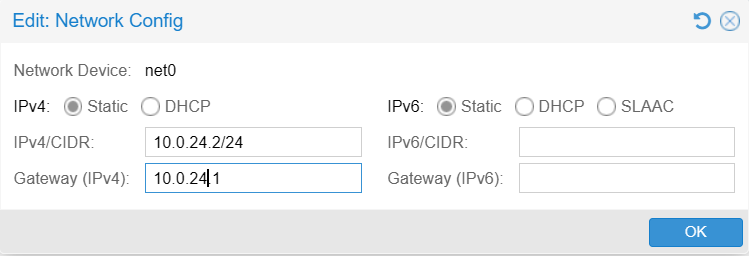
* IP 192.168.186.24 ở lab tuần 1 vẫn giữ nguyên không xóa.
* Gõ **sudo netplan apply** để apply changes.
* Gõ **ip a** để kiểm tra (hình cũ):



Ở máy **VM2**:

* Máy này có thể lên Cloud-Init và chỉnh nó như sau:



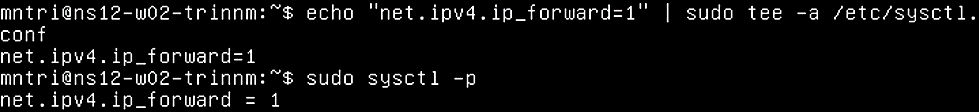


**\* Bật NAT trên VM1:**

Bây giờ, chúng ta cần cấu hình NAT để **VM2** có thể truy cập internet thông qua **VM1**:

a) Bật chuyển tiếp gói IPv4 (IP Forwarding) ở VM1:

* Chỉnh cấu hình sysctl:  
  **echo "net.ipv4.ip\_forward=1" | sudo tee -a /etc/sysctl.conf**
* Apply thay đổi: **sudo sysctl -p**



* Giải thích: Đảm bảo VM1 có thể route gói tin giữa các interface

b) Thiết lập NAT với iptables ở VM1:

**sudo iptables -t nat -A POSTROUTING -o ens18 -j MASQUERADE**

**sudo iptables -A FORWARD -i ens19 -o ens18 -j ACCEPT**

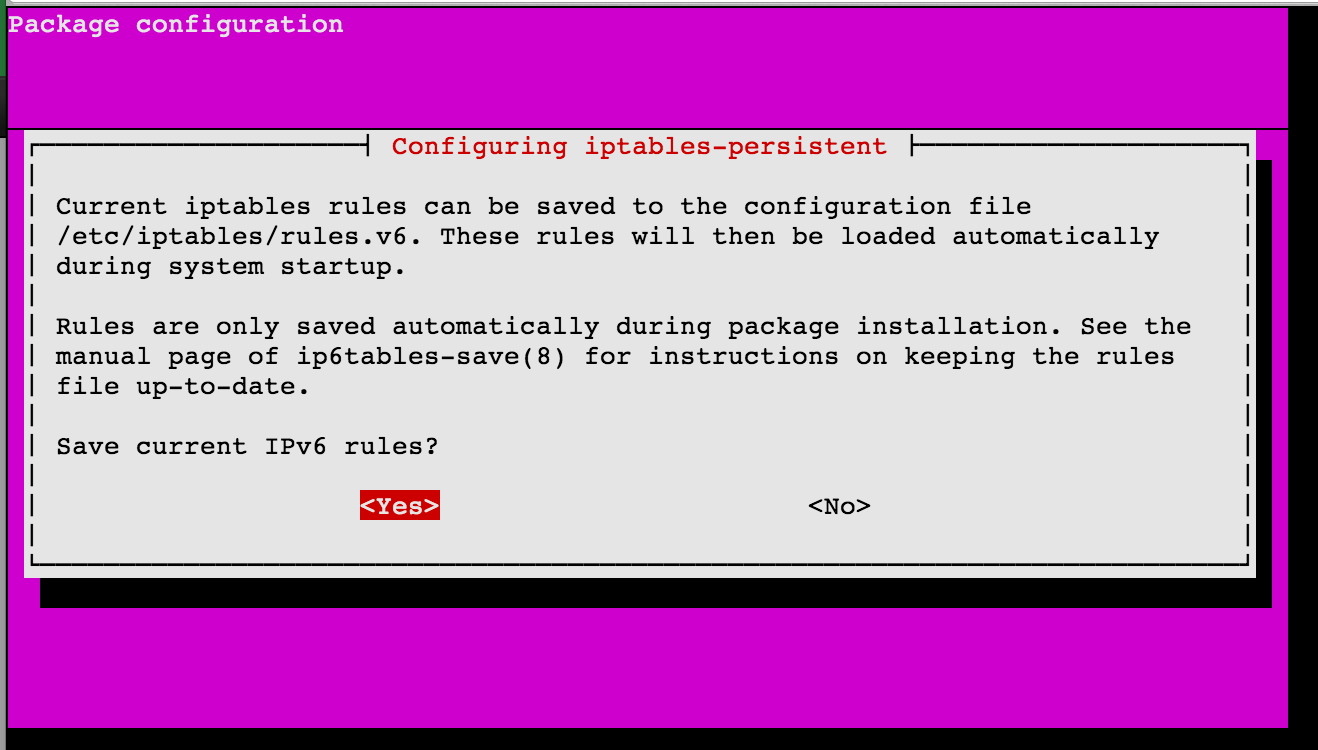
**sudo iptables -A FORWARD -i ens18 -o ens19 -m state --state RELATED,ESTABLISHED -j ACCEPT**

**\* Cài đặt tường lửa Persistent:**

Bây giờ tiến hành cài đặt iptables-persistent để lưu rules và config của mình sau khi reboot:

**sudo apt update**

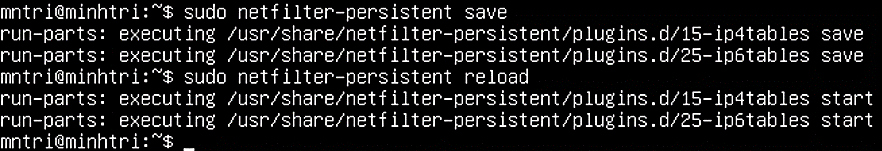
**sudo apt install iptables-persistent**



Lưu rules:

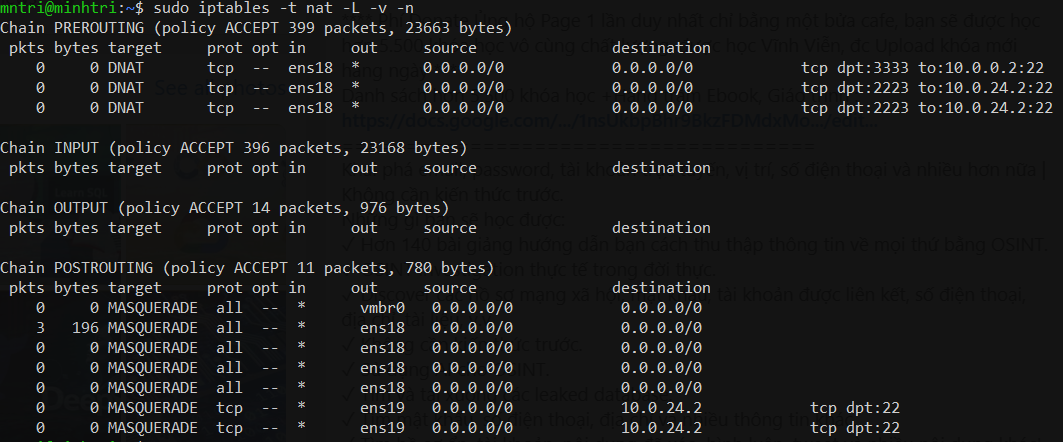
**sudo netfilter-persistent save**

**sudo netfilter-persistent reload**



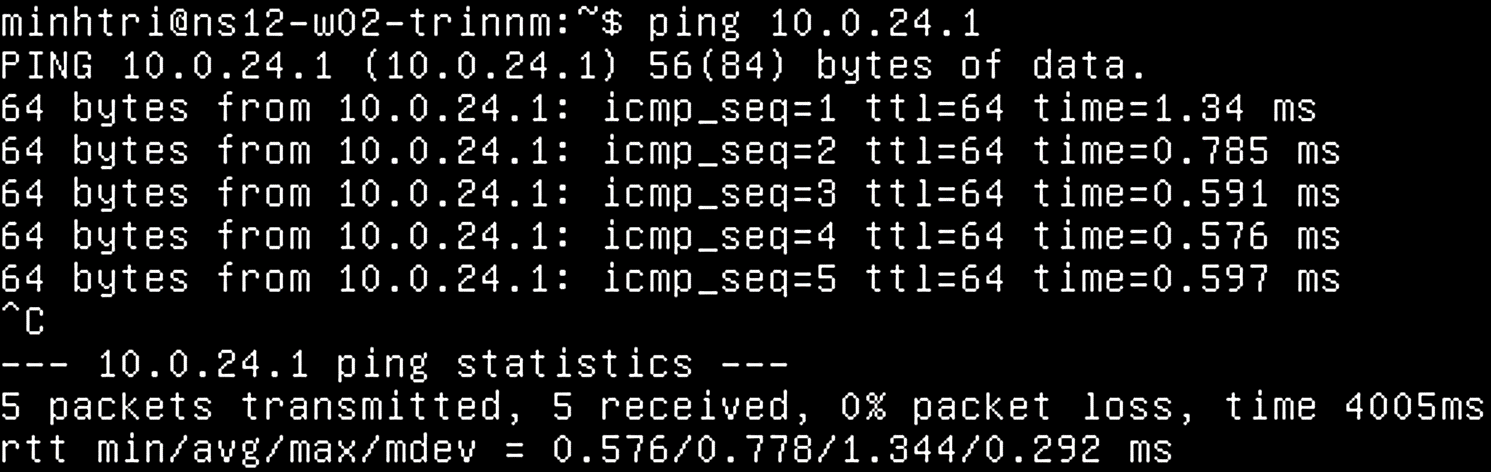
Kiểm tra lại:

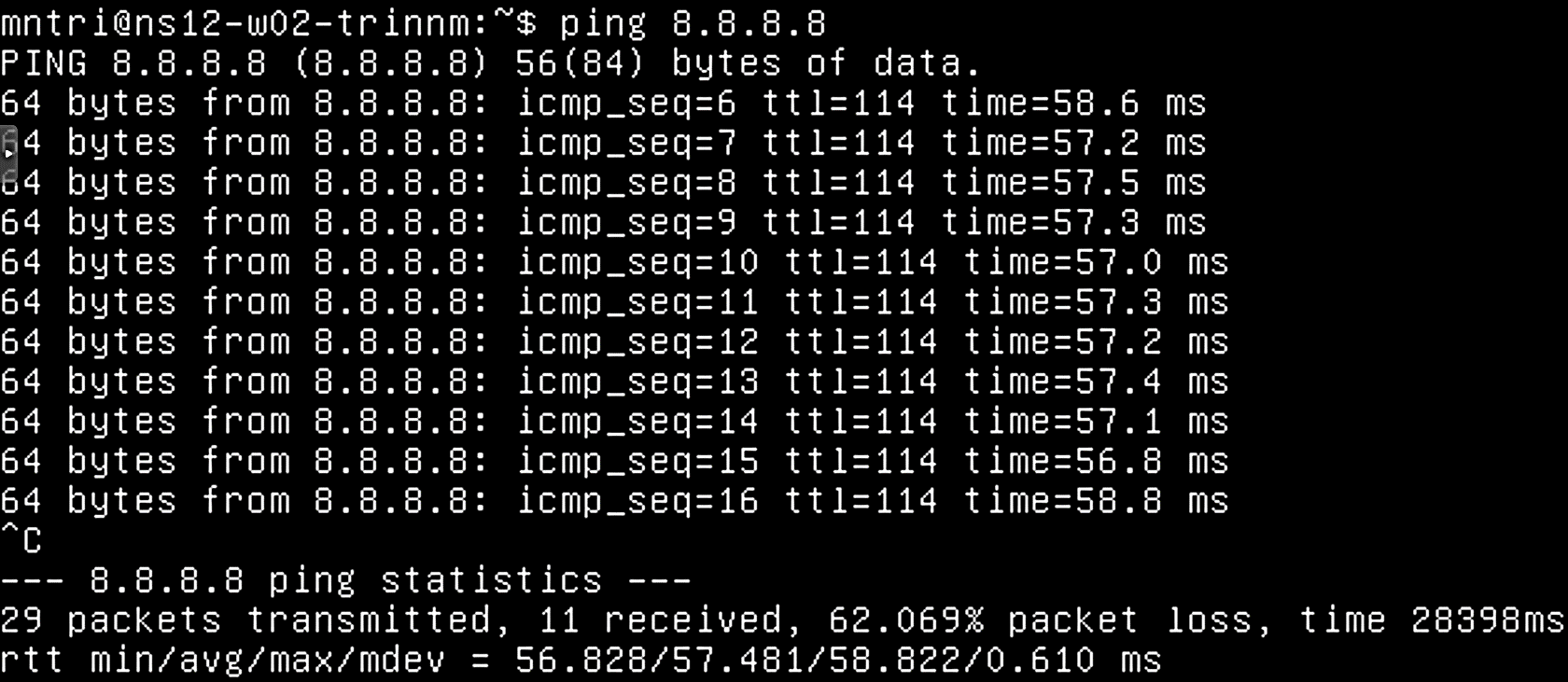
**sudo iptables -t nat -L -v -n**



**\*** **Test kết nối từ máy VM2:**

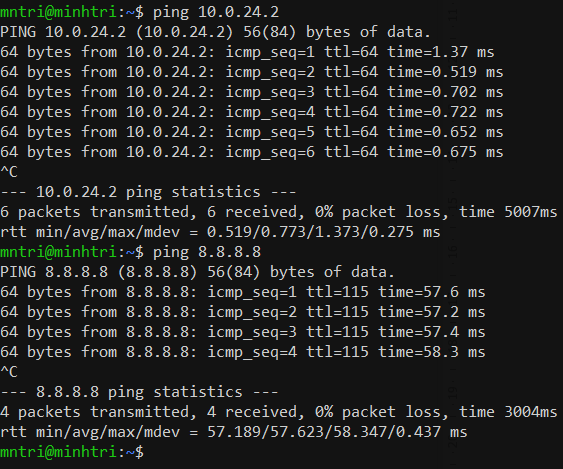
* Ping máy **VM1**: **ping 10.0.24.1**
* Ping 8.8.8.8

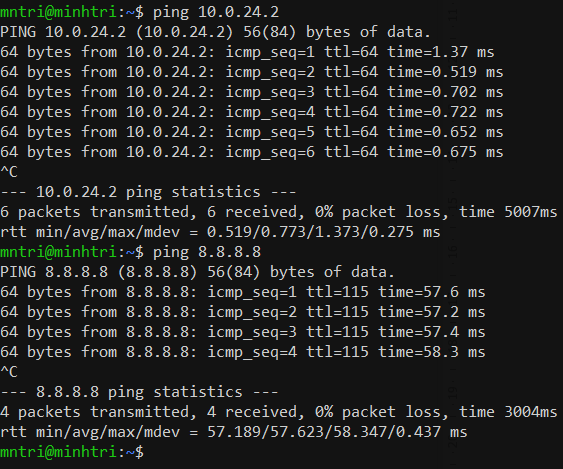




**\*** **Test kết nối từ máy VM1:**

* Ping máy **VM2**: **ping 10.0.24.2**
* Ping 8.8.8.8





## **PORT FORWARDING**

*Cấu hình port forwarding trên* ***VM1*** *để khi SSH vào IP WAN của* ***VM1*** *port 2222 thì có thể truy cập được SSH được thẳng vào* ***VM2***

a) Cấu hình Port Forwarding với iptables:

Ở máy **VM1**, gõ các lệnh sau:

# Enable NAT forwarding for port 2222 to VM2 (10.0.0.2:22)

**sudo iptables -t nat -A PREROUTING -i ens18 -p tcp --dport 2222 -j DNAT --to-destination 10.0.0.2:22**

# Allow forwarding packets to VM2

**sudo iptables -A FORWARD -p tcp -d 10.0.0.2 --dport 22 -m state --state NEW,ESTABLISHED,RELATED -j ACCEPT**

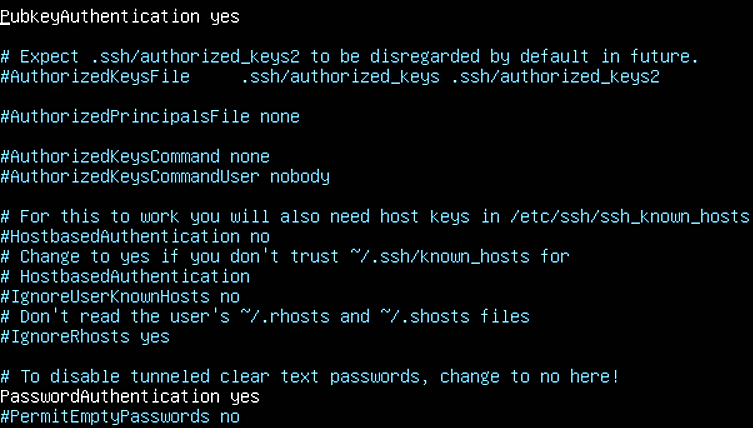
b) Save rule lại để tránh mất sau reboot:

**sudo netfilter-persistent save**

**sudo netfilter-persistent reload**

c) Sửa PubkeyAuthentication và PasswordAuthentication

Ở cả 2 máy **VM1** và **VM2** hãy để chúng thành yes:

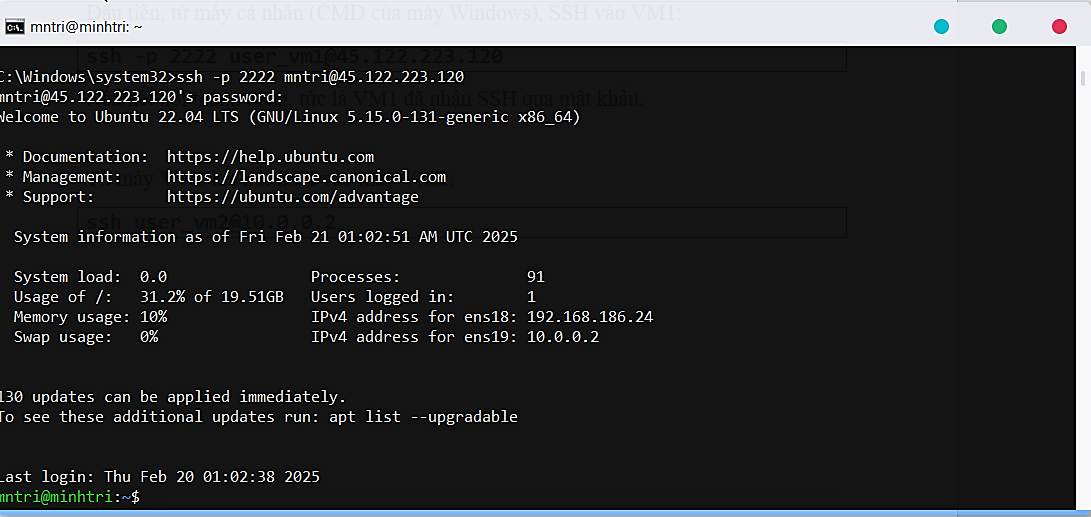


d) Test kết nối SSH bằng mật khẩu trước:

Đầu tiên, từ máy cá nhân (CMD của máy Windows), SSH vào **VM1**:

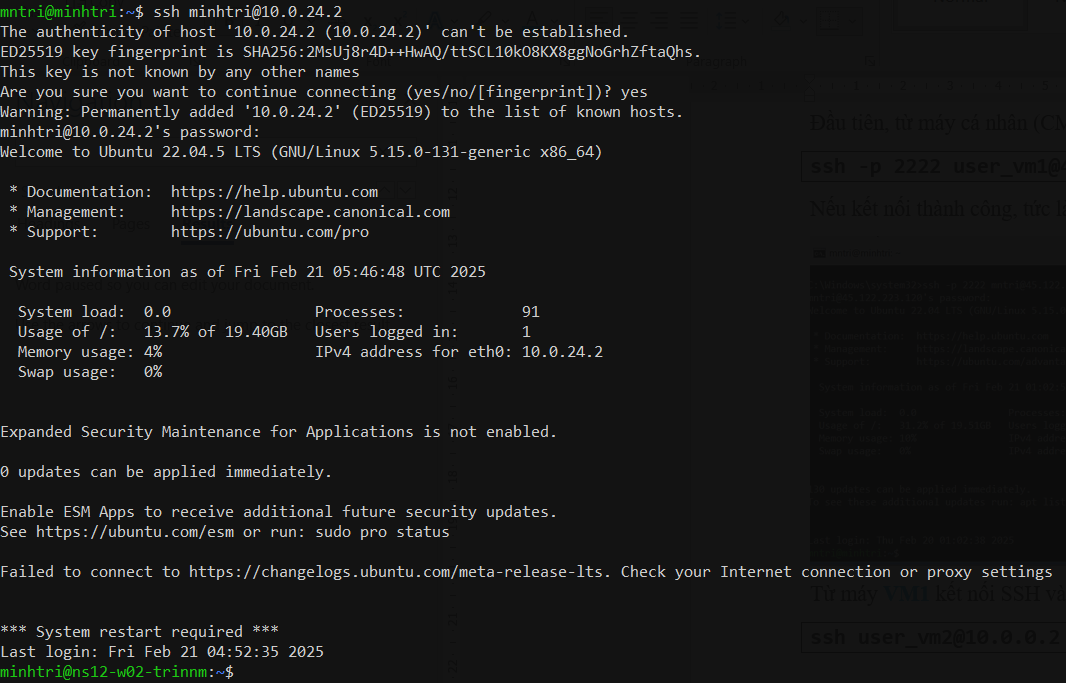
**ssh -p 2222 user\_vm1@45.122.223.120**

(Port 2222 là của VM1) Nếu kết nối thành công, tức là **VM1** đã nhận SSH qua mật khẩu.



Từ máy **VM1** kết nối SSH vào máy **VM2**:

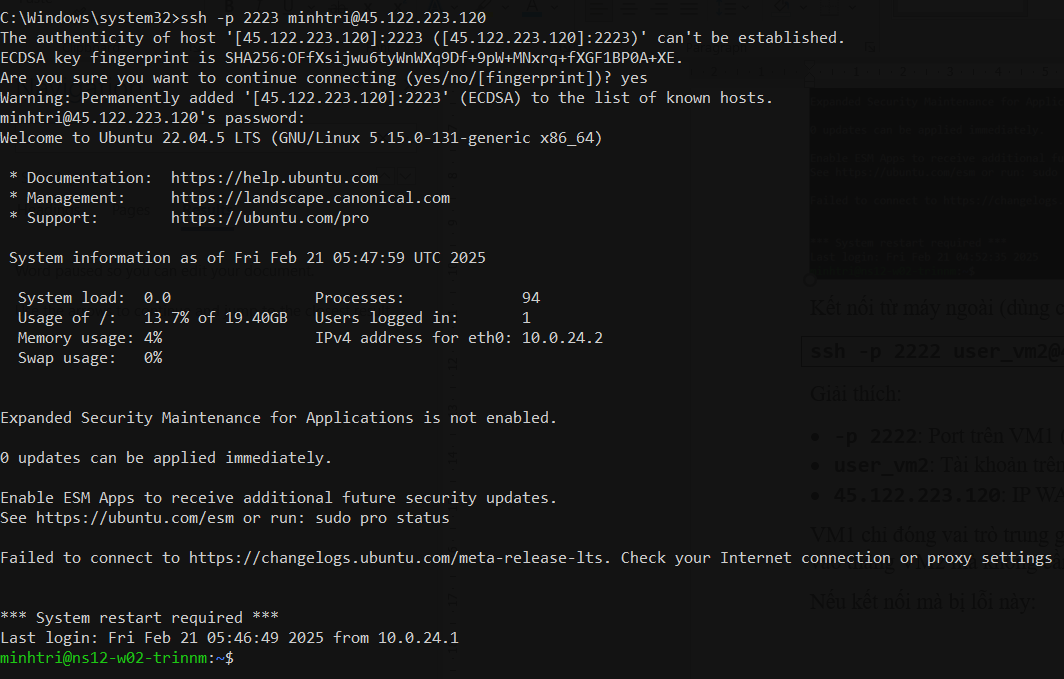
**ssh user\_vm2@10.0.24.2**



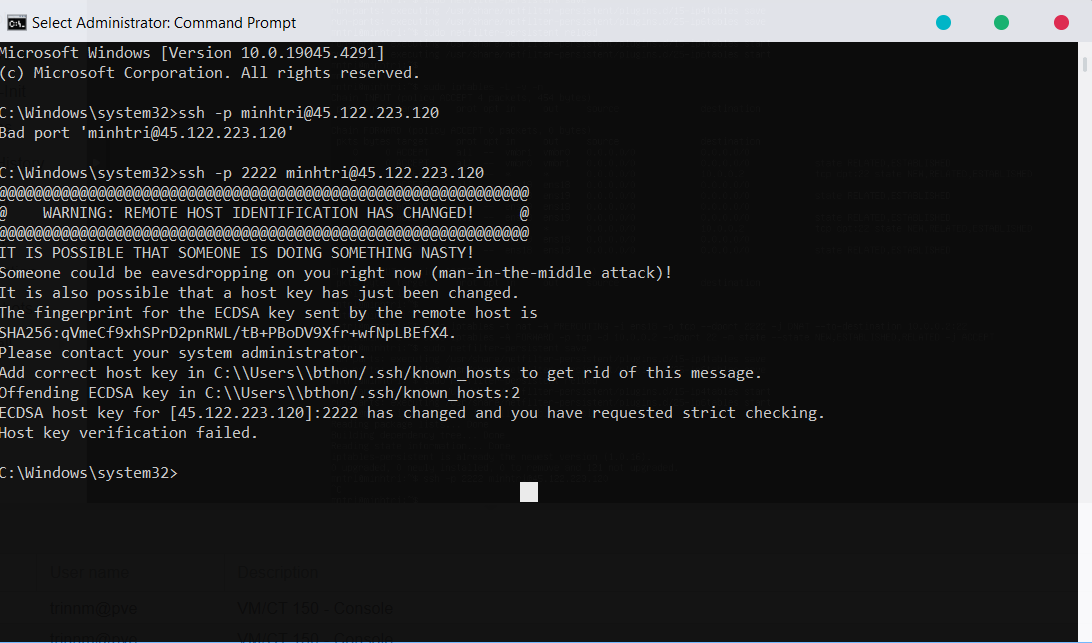
Kết nối từ máy ngoài (dùng cmd của Windows) vào VM2 thông qua VM1:

**ssh -p 2223 user\_vm2@45.122.223.120**

VM1 chỉ đóng vai trò trung gian chuyển tiếp (port forwarding). Sau khi kết nối, bạn sẽ vào thẳng VM2 mà không cần SSH vào VM1 trước.



Nếu kết nối mà bị lỗi này:



Lỗi này xảy ra vì host key của máy đích (VM2) khác với host key cũ mà máy client của bạn đã lưu.

Trên CMD của Windows, gõ

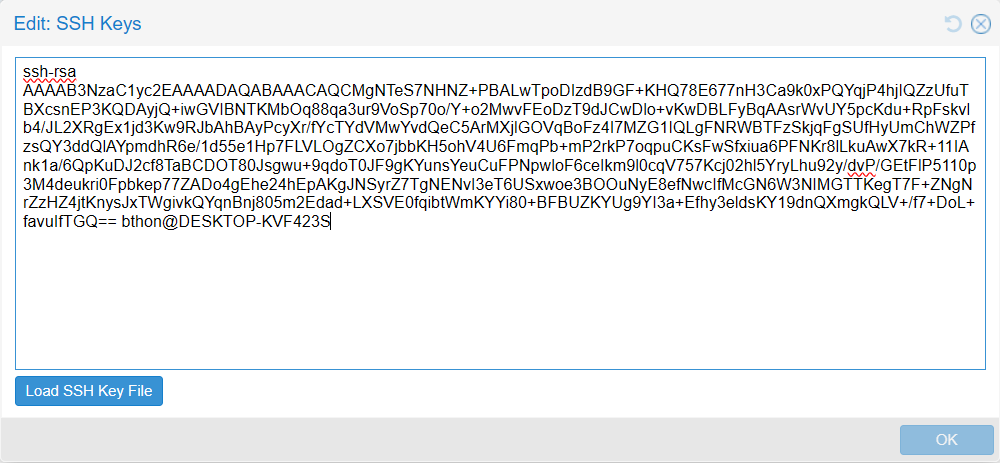
**ssh-keygen -R "[45.122.223.120]:2222"**

Sau đó thử lại SSH.

e) Test kết nối SSH bằng pubkey:

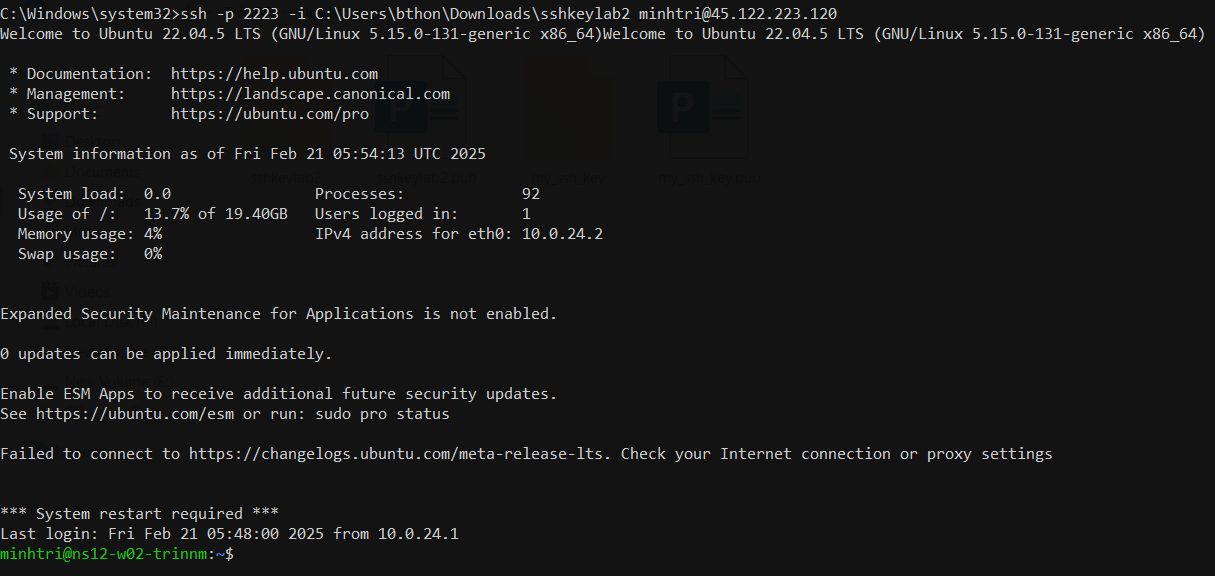
Dựa vào lab tuần 1:

* Tạo key:  
  **ssh-keygen -t rsa -b 4096 -f ~/.ssh/sshkeylab2**
* Có thể nhập key ở Cloud-Init > SSH Public Key:



**Kết quả:** Dùng CMD của Windows gõ

**ssh -p 2223 -i C:\Users\<user\_name>\Downloads\sshkeylab2 user\_vm2@<IP WAN>**



**TROUBLESHOOTING KHÁC:**

**a/ Lúc đang làm mà phát hiện bị hết bộ nhớ cần extend:**

Đầu tiên hãy kiểm tra hệ thống file của phân vùng:

**df -Th**

* Nếu /dev/vda1 là ext4, bạn có thể resize trực tiếp.
* Nếu là XFS, thì cần cách khác.

Mở rộng phân vùng bằng parted:

**sudo parted /dev/vda**

Bên trong parted, nhập:

**print**

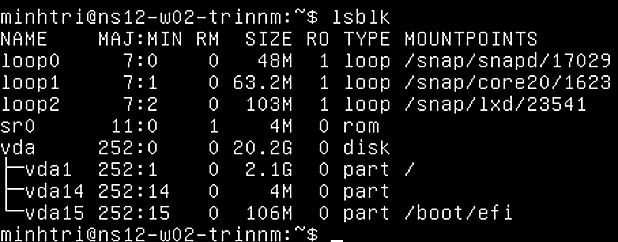
**resizepart 1 100%**

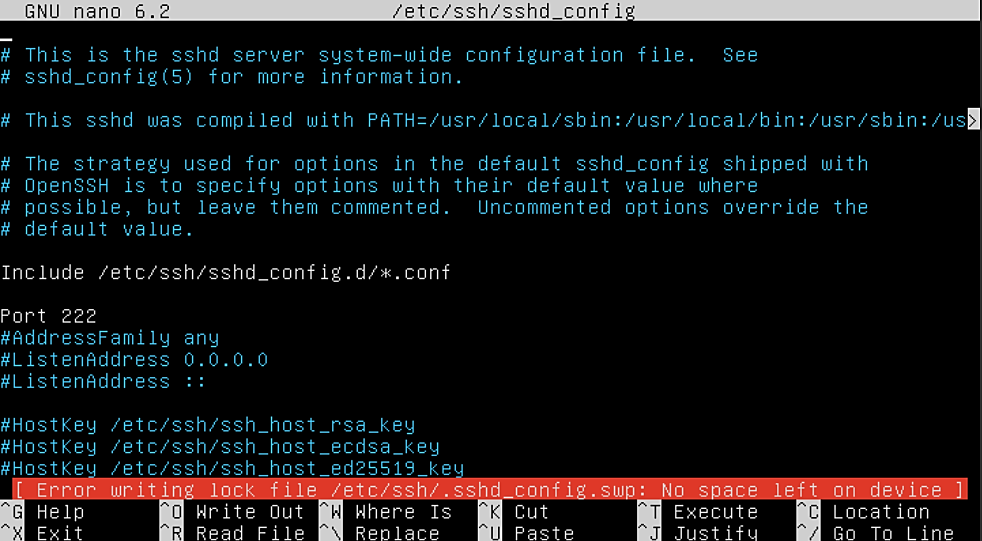
**quit**

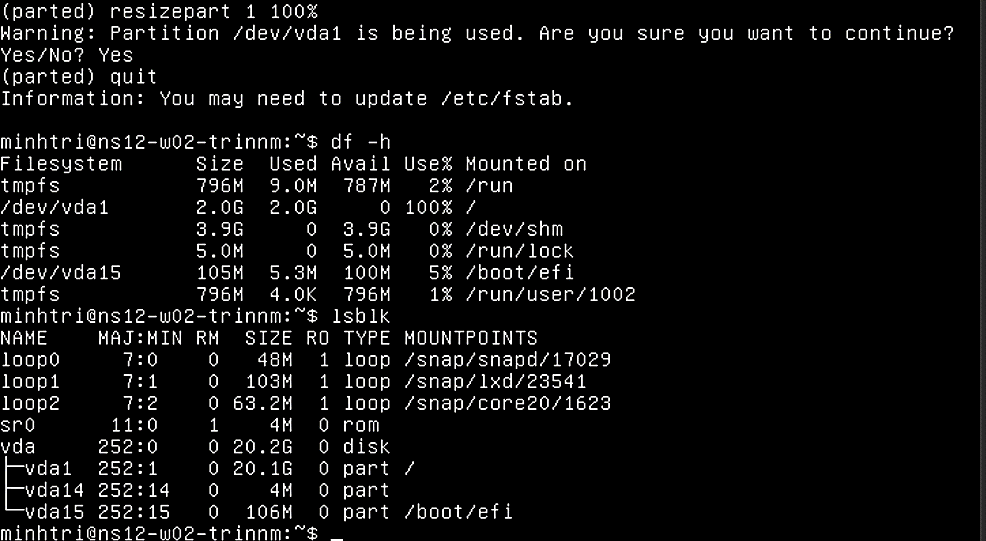
Kiểm tra lại dung lượng mới

**df -h**

**lsblk**

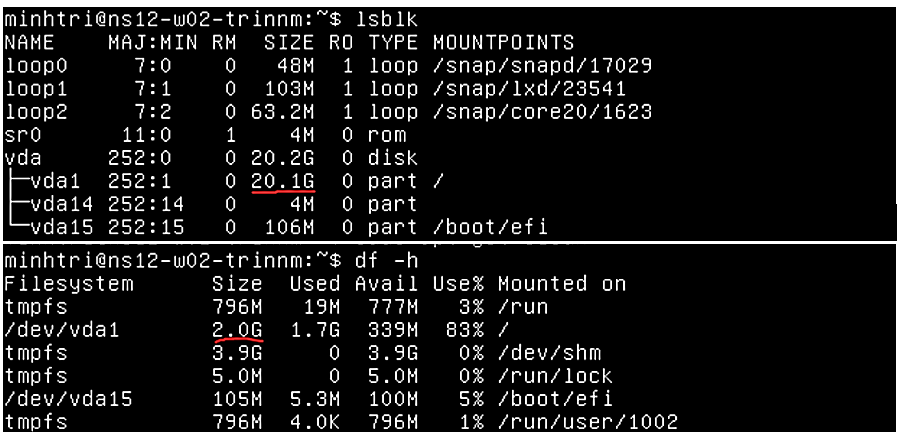


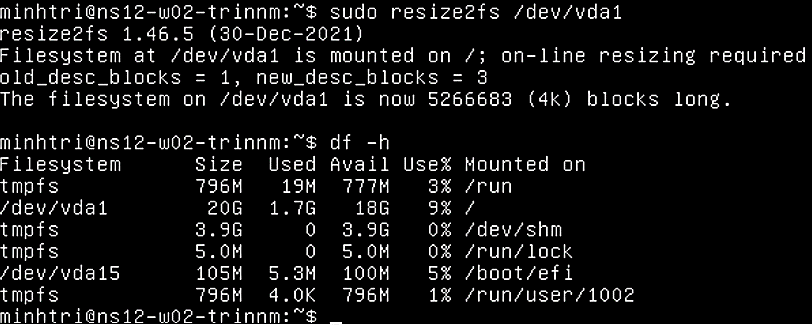




Mình đã mở rộng phân vùng vật lý, giờ mình phải mở rộng filesystem. (resize filesystem)

**sudo resize2fs /dev/vda1**





**b/ Nếu file sshd\_config bị lỗi (Cách khôi phục sshd\_config)**

***Bước 1: Kiểm tra file backup (nếu có)***

**ls -l /etc/ssh/sshd\_config\***

Nếu có file như sshd\_config.old hoặc sshd\_config.bak, bạn có thể khôi phục bằng:

**sudo mv /etc/ssh/sshd\_config.bak /etc/ssh/sshd\_config**

Rồi mở lại để kiểm tra:

**sudo nano /etc/ssh/sshd\_config**

***Bước 2: Kiểm tra file swap (.swp)***

**ls -l /etc/ssh/.sshd\_config.swp**

Nếu file này tồn tại, hãy xóa nó và thử mở lại:

**sudo rm /etc/ssh/.sshd\_config.swp**

**sudo nano /etc/ssh/sshd\_config**

***Bước 3: Khôi phục sshd\_config mặc định***

Nếu file hoàn toàn mất, hãy tạo lại với nội dung mặc định:

**sudo cp /usr/share/openssh/sshd\_config /etc/ssh/sshd\_config**

Rồi chỉnh sửa theo nhu cầu:

**sudo nano /etc/ssh/sshd\_config**

Restart lại ssh:

**sudo systemctl restart ssh**

**c/ Sửa lỗi “Bad lock file is ignored: /etc/ssh/.sshd\_config.swp”**



(là nếu như file .sshd\_config.swp bị corrupt)

Thì cứ xóa nó rồi mở lại ssh config để kiểm tra

**sudo rm /etc/ssh/.sshd\_config.swp**

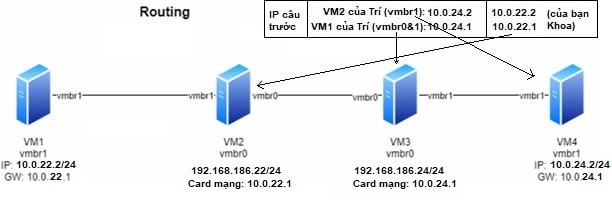
**sudo nano /etc/ssh/sshd\_config**

## **ROUTING**

Môi trường: đề tài yêu cầu 2 bạn làm chung 1 bài lab. Mỗi bạn được cấp 2 VM và cấu hình mạng như hình sau:

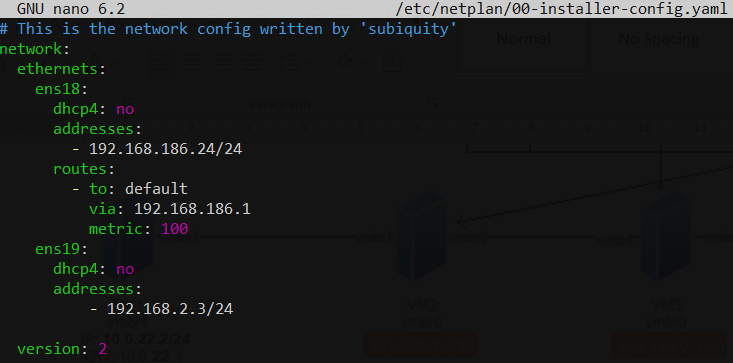
• Bạn 1: VM2 (có vmbr0 và vmbr1) và VM1 (only vmbr1)

• Bạn 2: VM3 (có vmbr0 và vmbr1) và VM4 (only vmbr1)

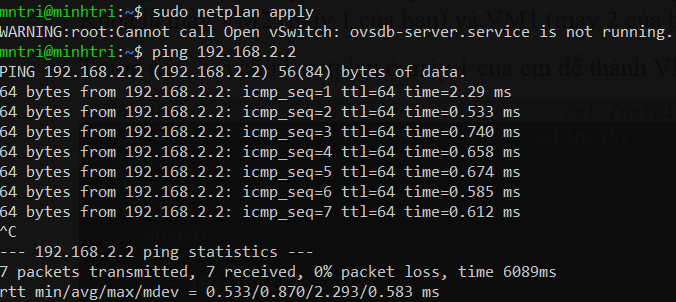


Bài này em sẽ là máy VM3 (máy 1 của em) và VM4 (máy 2 của em), còn bạn Nguyễn Khoa là máy VM2 (máy 1 của bạn) và VM1 (máy 2 của bạn).

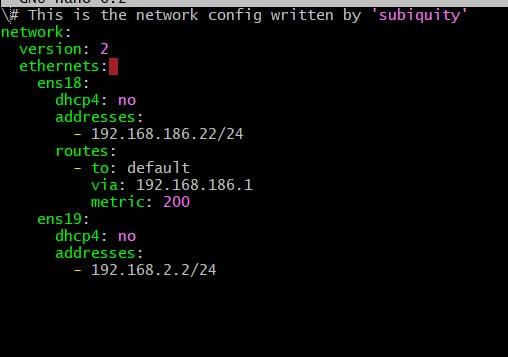
Trước tiên, em chỉnh netplan ở máy 1 của em để thành VM3.



Test ping 192.168.2.2



Bạn Khoa chỉnh netplan ở máy 1 của bạn để thành VM2.



Ở máy VM1, thêm static route để trỏ về mạng của VM2 qua IP của VM2:

**sudo ip route add 192.168.2.0/24 via 10.0.22.2 dev eth0**

Ở VM2, trỏ về mạng của VM1 qua IP của VM1:

**sudo ip route add 10.0.22.0/24 via 192.168.2.2 dev ens19**

Ở VM3:

**sudo ip route add 10.0.24.0/24 via 192.168.2.3 dev ens19**

Ở VM4:

**sudo ip route add 192.168.2.0/24 via 10.0.24.3 dev eth0**

Bảng cấu hình ở từng máy:

VM1:

Ảnh có chứa văn bản, ảnh chụp màn hình, Phông chữ

Nội dung do AI tạo ra có thể không chính xác.

VM2:

Ảnh có chứa văn bản, Phông chữ, ảnh chụp màn hình, màu đen

Nội dung do AI tạo ra có thể không chính xác.

VM3:

Ảnh có chứa văn bản, ảnh chụp màn hình, Phông chữ, màu đen

Nội dung do AI tạo ra có thể không chính xác.

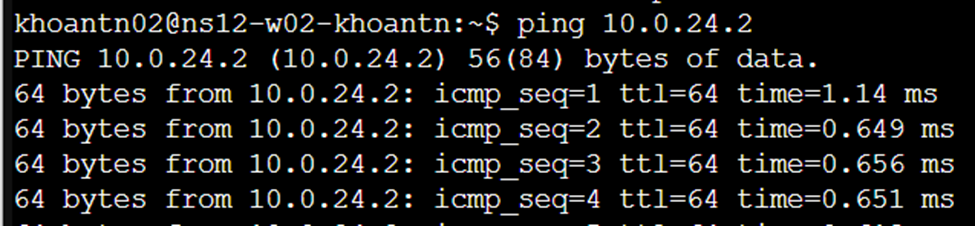
VM4:

Ảnh có chứa văn bản, ảnh chụp màn hình, Phông chữ

Nội dung do AI tạo ra có thể không chính xác.

Test ping từ máy VM1 sáng VM4 và ngược lại:

Từ máy VM1 sang VM4:



Từ máy VM4 sang VM1:

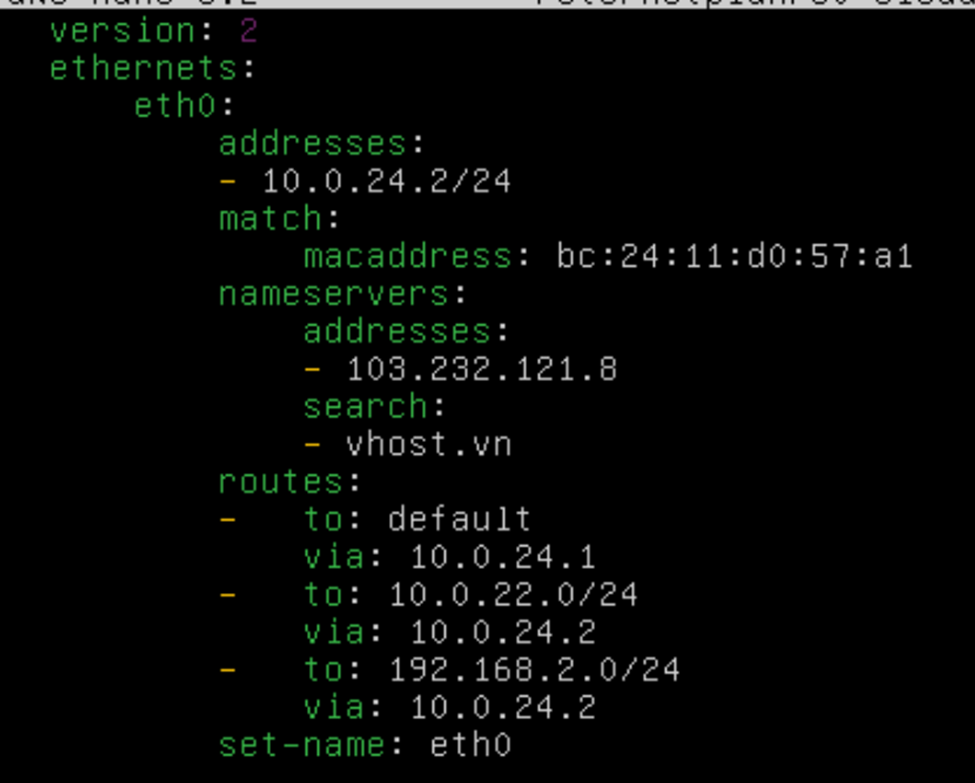
Ảnh có chứa văn bản, ảnh chụp màn hình, Phông chữ

Nội dung do AI tạo ra có thể không chính xác.

Vấn đề lưu lại các cấu hình route thì ta lưu ở file mạng luôn:

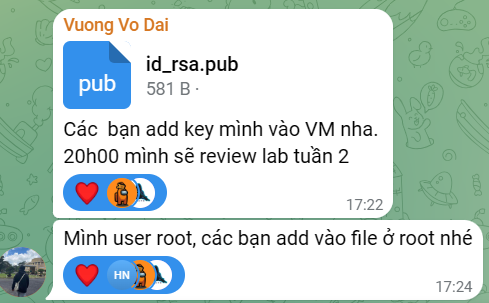
**sudo nano /etc/netplan/(file mạng)**

Thì lúc mở lên nó sẽ tự động cấu hình luôn



*(Các máy khác tương tự)*

BONUS KHÔNG CÓ TRONG BÀI



Nếu anh hỏi cần đưa key vào root user của 2 máy thì:

**sudo -i**

**sudo nano /root/.ssh/authorized\_keys**

**chmod 600 /root/.ssh/authorized\_keys**

**chmod 700 /root/.ssh**