**BÁO CÁO TUẦN 1**

*Nguyễn Ngọc Minh Trí*

**MỤC LỤC**

[**MỤC LỤC** 2](#_Toc190240067)

[1. OS (UBUNTU SERVER 22.04 - PROXMOX) 3](#_Toc190240068)

[2. SỬ DỤNG LVM ĐỂ QUẢN LÝ Ổ CỨNG 9](#_Toc190240069)

[3. SỬ DỤNG LVM ĐỂ MỞ RỘNG PHÂN VÙNG Ổ CỨNG 11](#_Toc190240070)

[4. SECURE SSH 15](#_Toc190240071)

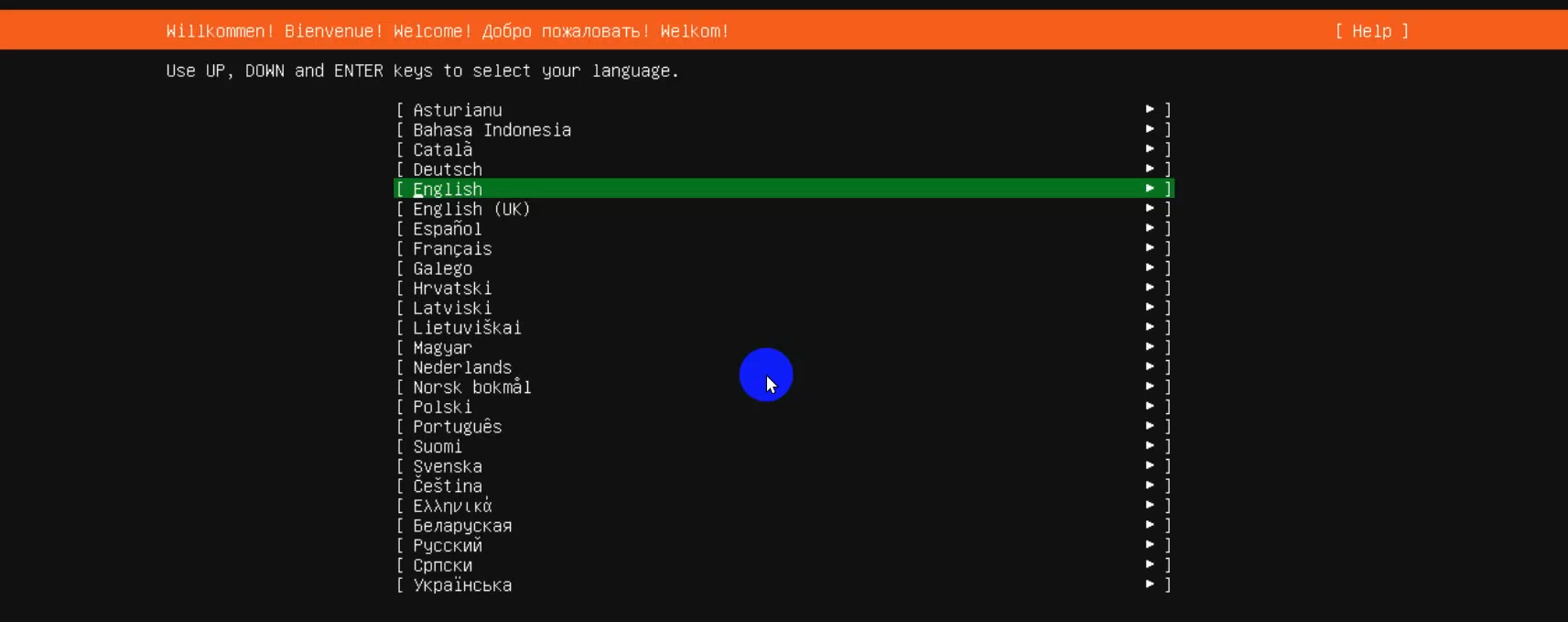
## **OS (UBUNTU SERVER 22.04 - PROXMOX)**

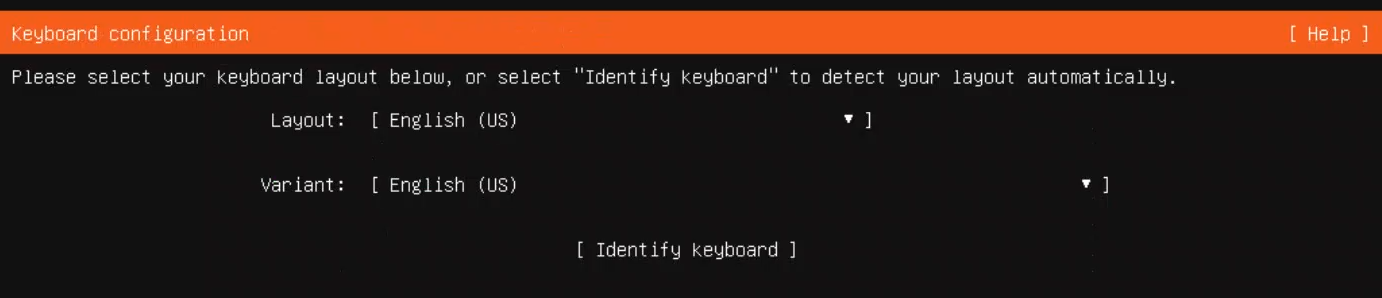
Cài đặt hệ điều hành Ubuntu 22.04 với các yêu cầu sau:

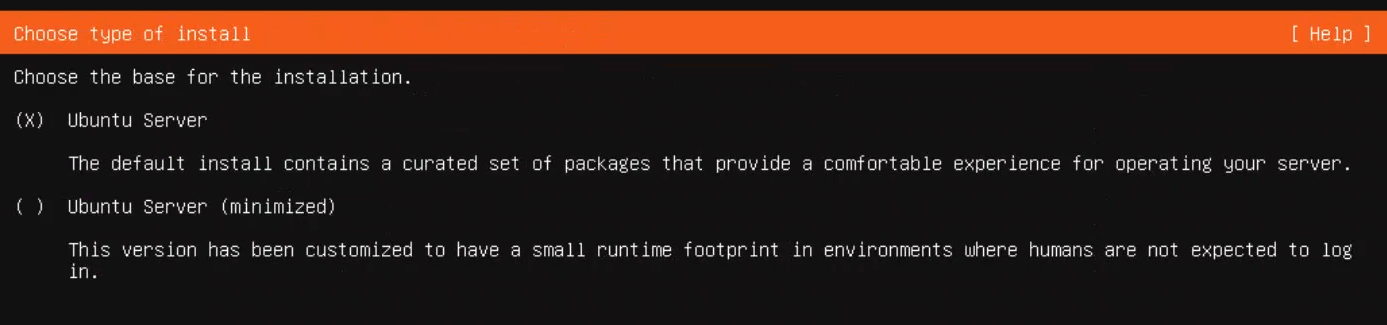
a/ Partition: chỉ 1 partition /

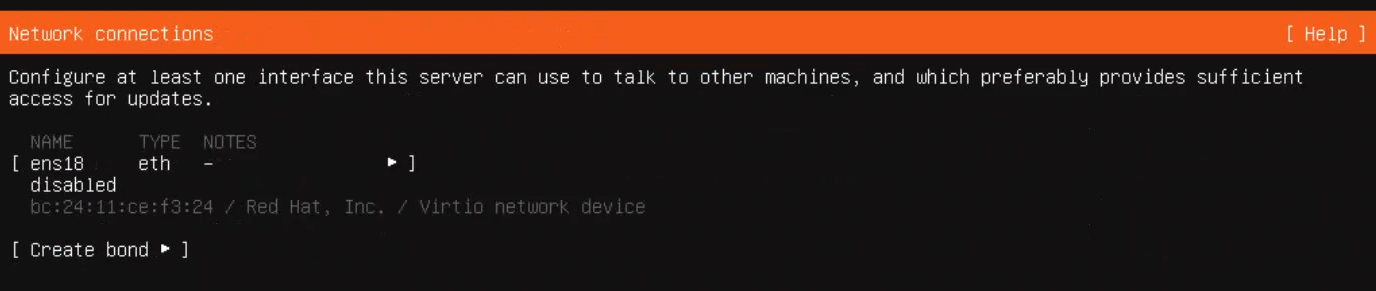
b/ SWAP: không.

c/ File system: EXT4

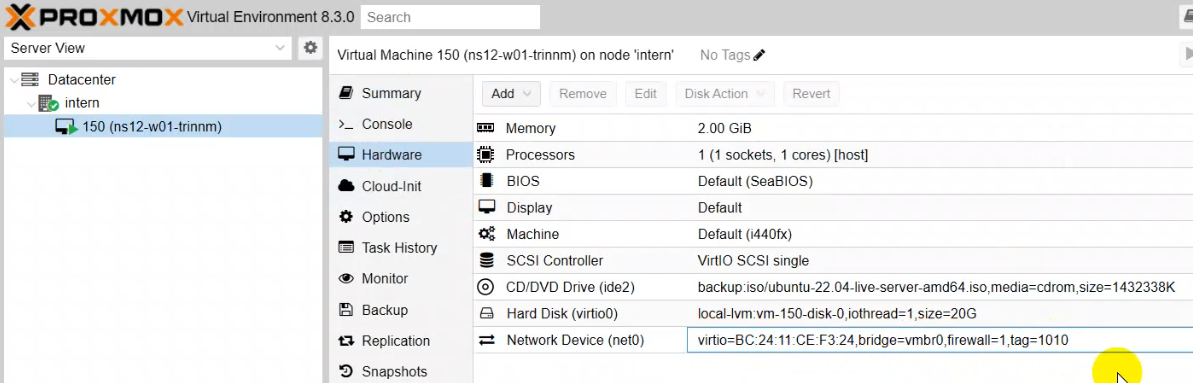




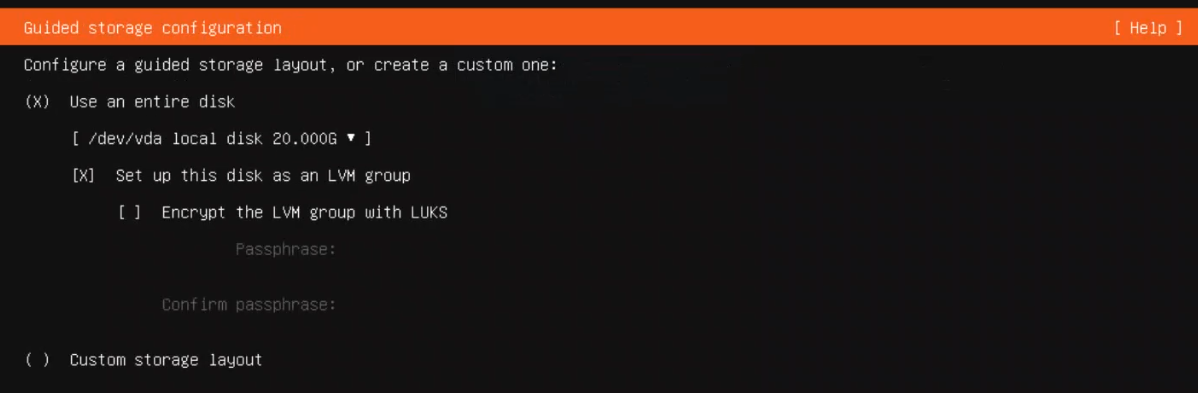




Mình add VLAN tag ở ngoài hardware (ở ngoài Settings Proxmox)



Bấm next tiếp đến phần Storage Configuration



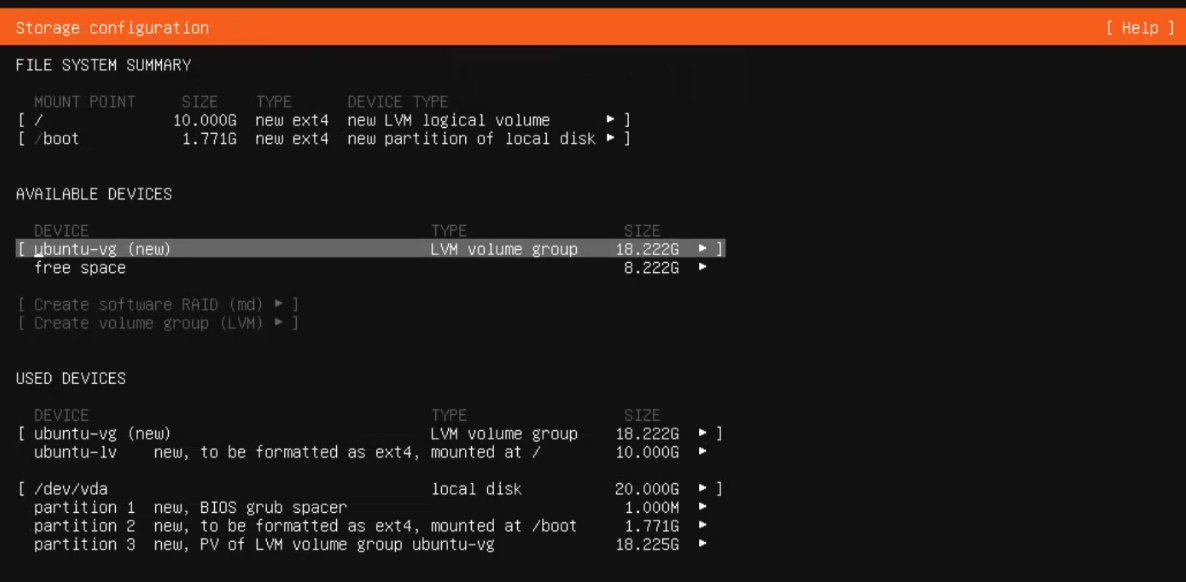
Giữ nguyên settings: Use an entire disk & Set up this disk as an LVM group. Theo yêu cầu của đề bài là:

a/ Partition: chỉ 1 partition /

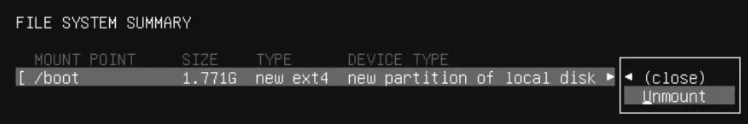
b/ SWAP: không.

c/ File system: EXT4

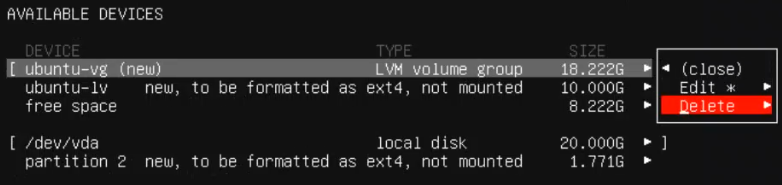
Nên là em sẽ xóa hết tất cả các mount point, partition và devices, chỉ giữ lại partition 1 (bootloader partition)

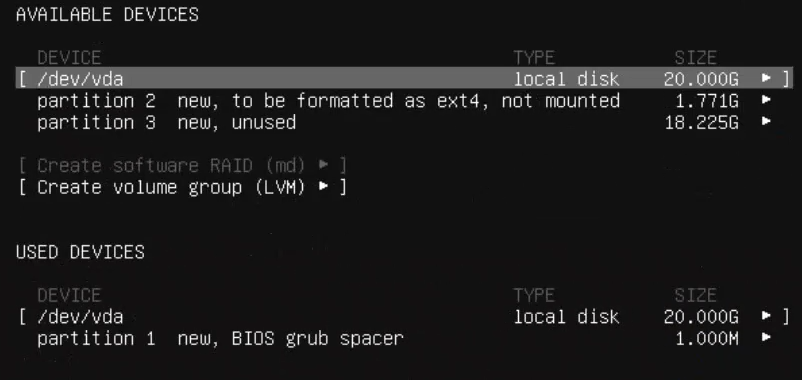


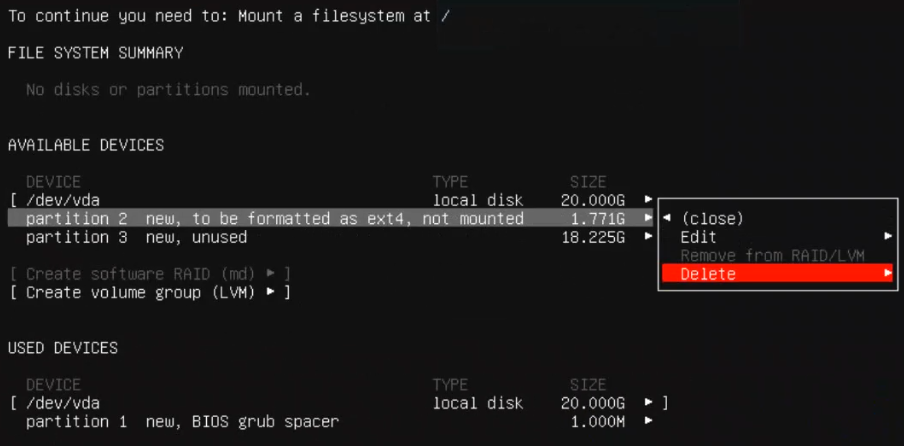
Ở phần FILE SYSTEM SUMMARY > MOUNT POINT: Unmount hết tất cả



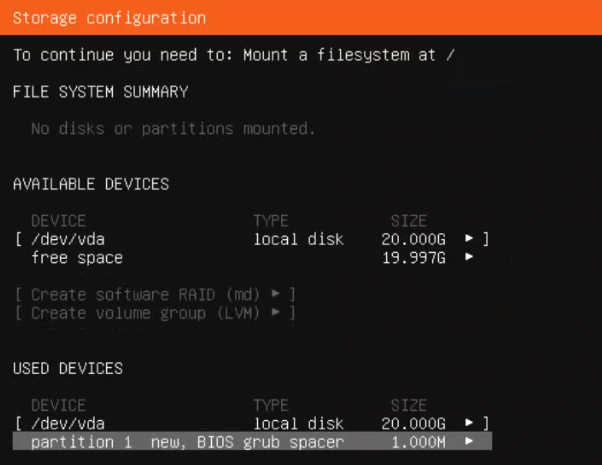
Ở phần AVAILABLE DEVICES: Xóa hết tất cả các DEVICE







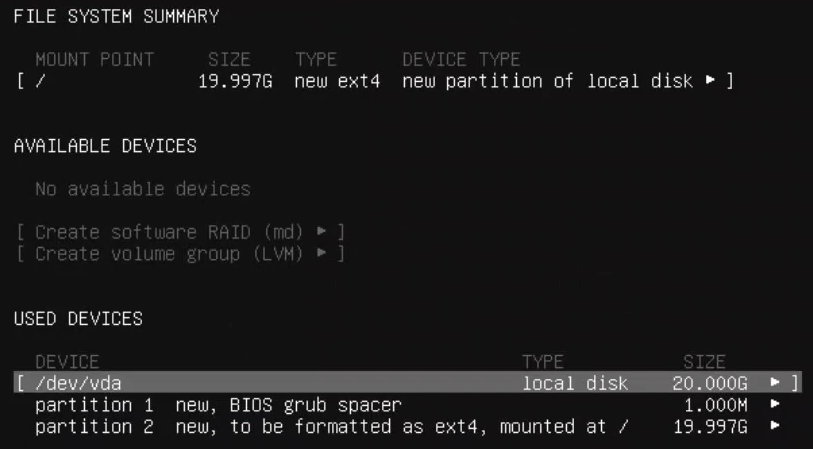
Sau khi xóa hết tất cả thì cuối cùng sẽ ra như sau:



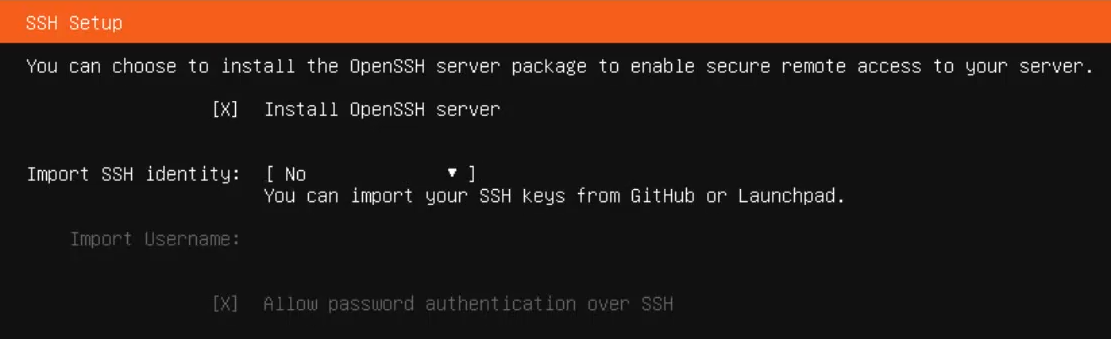
Tạo Partition mới:

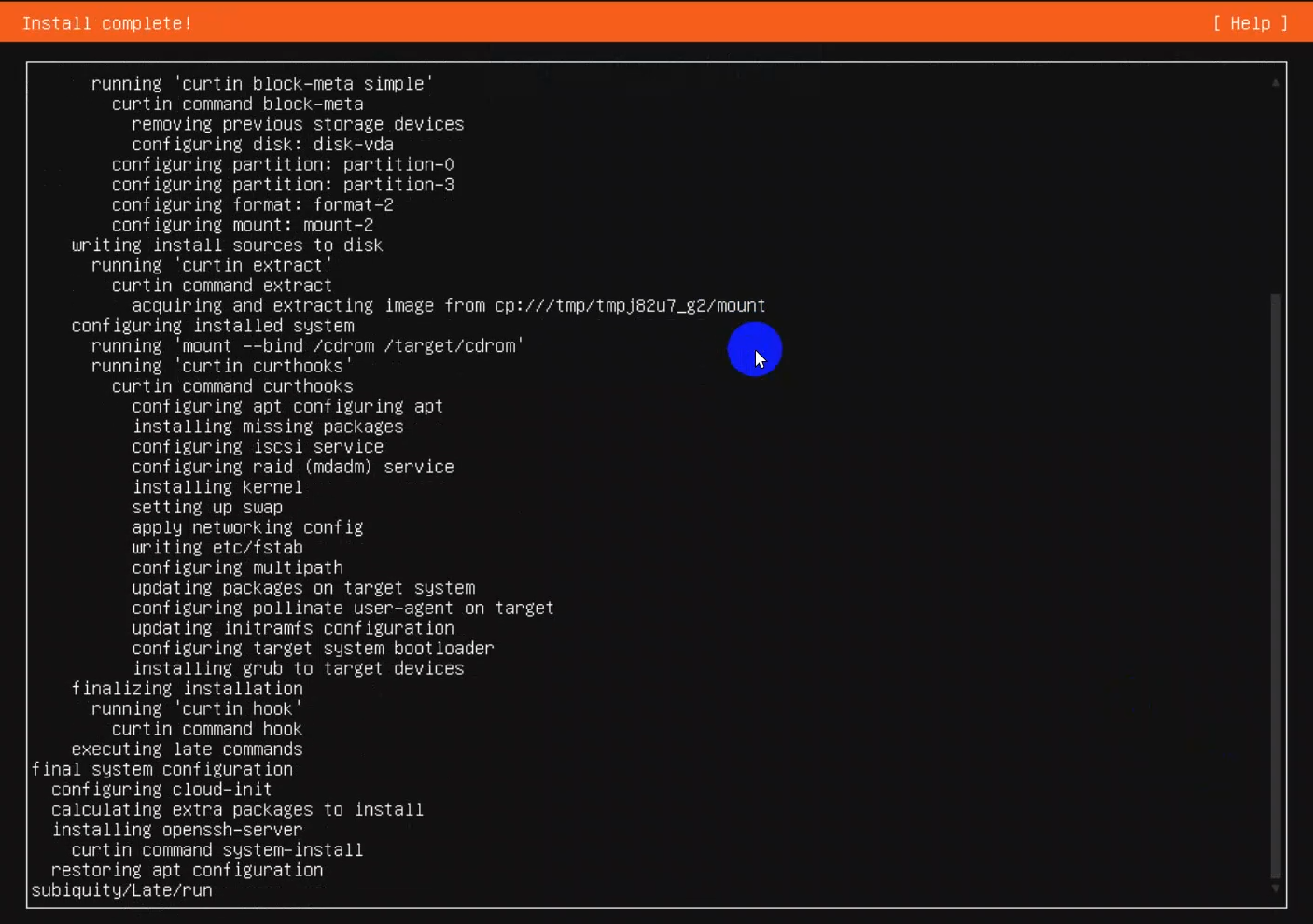






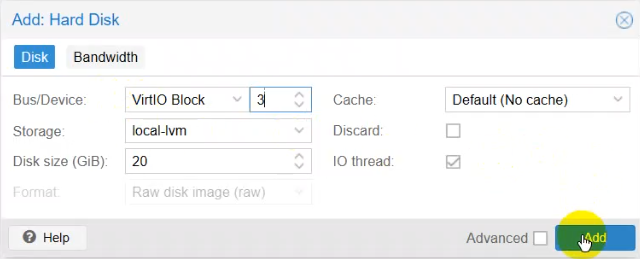


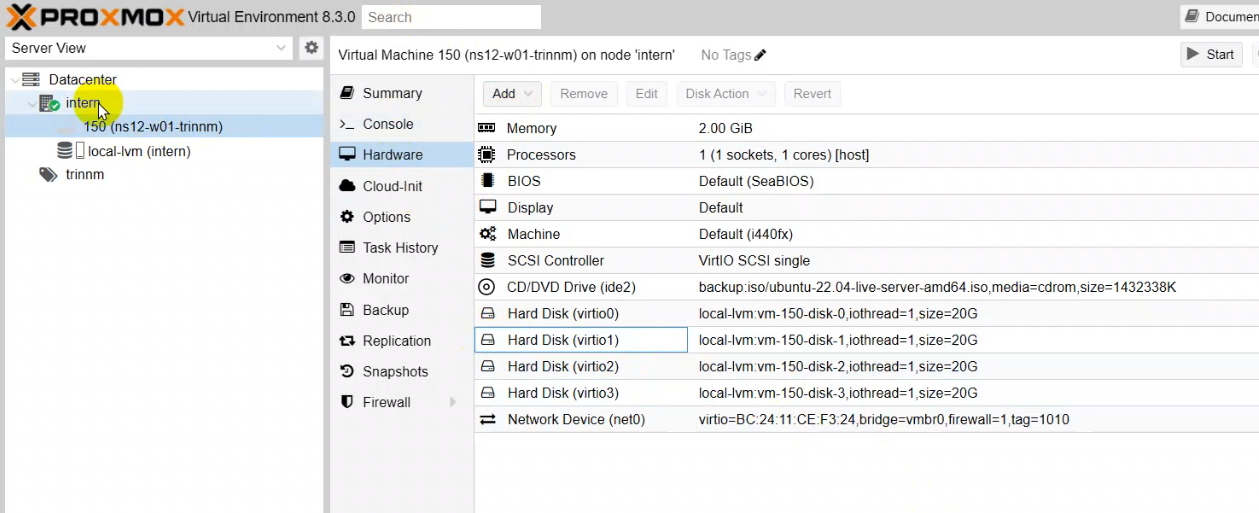




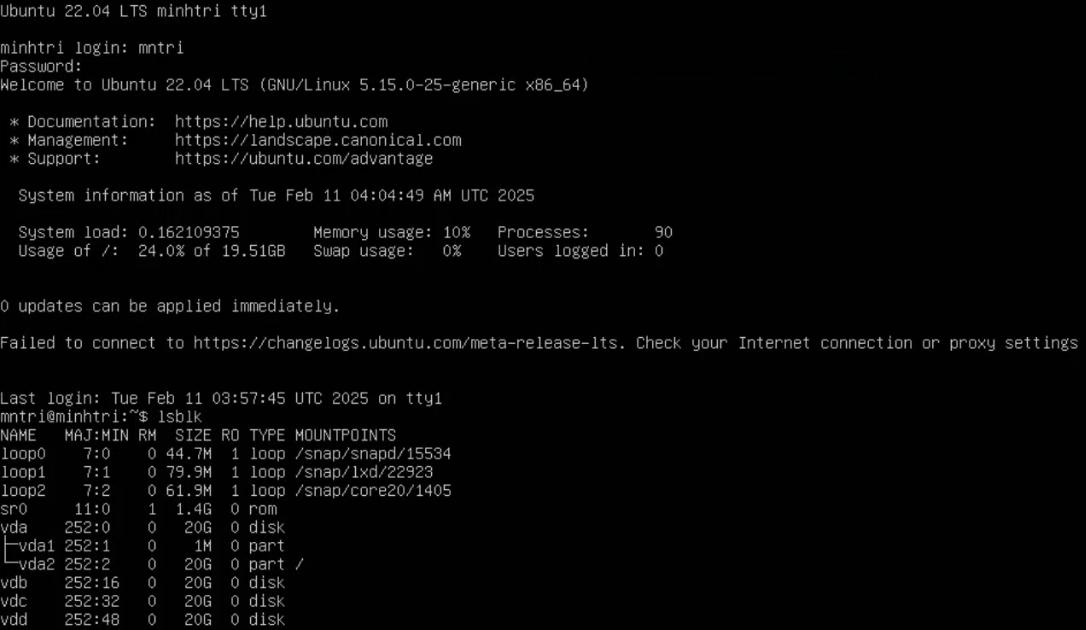
## **SỬ DỤNG LVM ĐỂ QUẢN LÝ Ổ CỨNG**

#### **2.1. Add thêm 3 ổ cứng 20GB cho VM ở phần 1 ở trên**



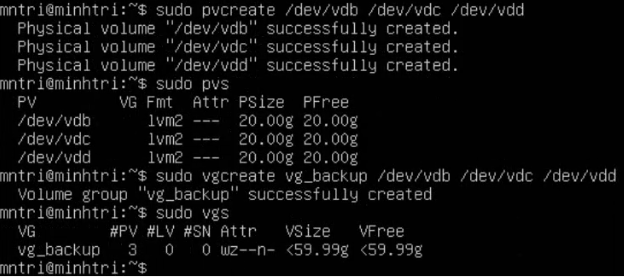


Dùng lệnh **lsblk** để kiểm tra



#### **2.2. Sử dụng LVM để tạo 1 Logical Volume kết nối 3 ổ cứng này thành 1 ổ 60GB**

* Chạy lệnh **sudo pvcreate /dev/vdb /dev/vdc /dev/vdd** để tạo Physical Volume cho từng ổ cứng mới. Sau đó dùng lệnh **sudo pvs** để kiểm tra lại.
* Tạo một Volume Group (gọi là **vg\_backup**) từ 3 ổ cứng:  
  **sudo vgcreate vg\_backup /dev/sdb /dev/sdc /dev/sdd**
* Kiểm tra lại bằng: **sudo vgs**
* Tạo một Logical Volume (tên **lv\_backup**) có dung lượng 60GB:  
  **sudo lvcreate -L 59G -n lv\_backup vg\_backup**
* Kiểm tra lại: **sudo lvs**





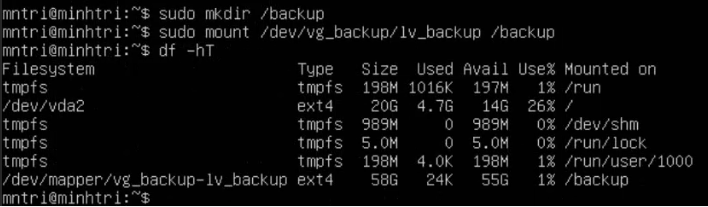
#### **2.3. Tạo format Logical Volume với file system EXT4**

Chạy lệnh sau để format Logical Volume **lv\_backup** thành hệ thống file EXT4:  
**sudo mkfs.ext4 /dev/vg\_backup/lv\_backup**



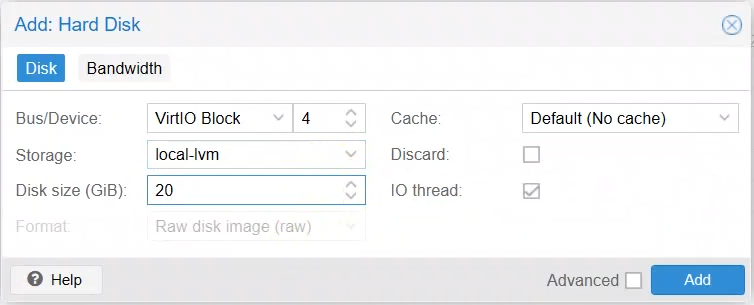
#### **2.4. Mount phân vùng này vào folder /backup**

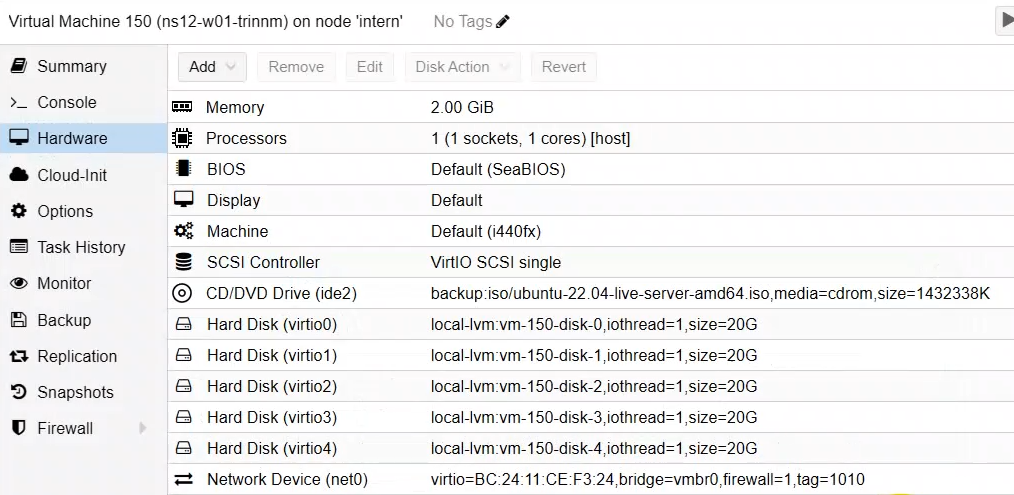
* Tạo thư mục /backup:  
  **sudo mkdir /backup**
* Mount Logical Volume vào /backup:   
  **sudo mount /dev/vg\_backup/lv\_backup /backup**
* Kiểm tra lại bằng: **df -hT**



## **SỬ DỤNG LVM ĐỂ MỞ RỘNG PHÂN VÙNG Ổ CỨNG**

#### **3.1. Add thêm 1 ổ cứng 20GB cho VM ở phần 1 ở trên**

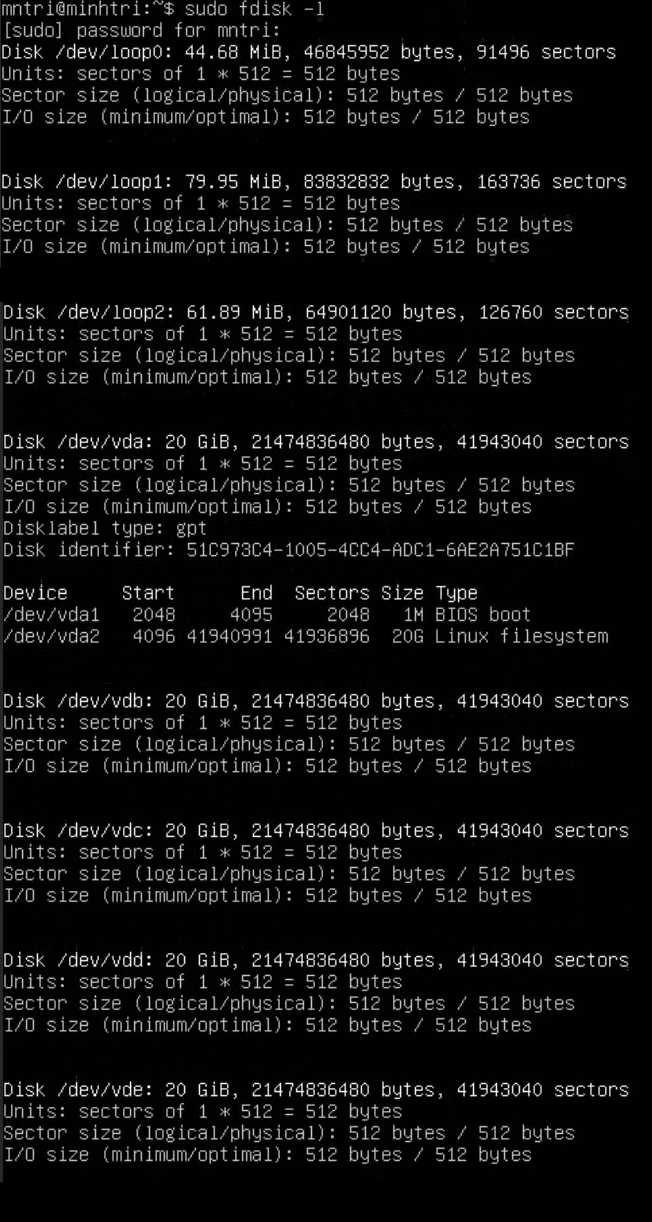


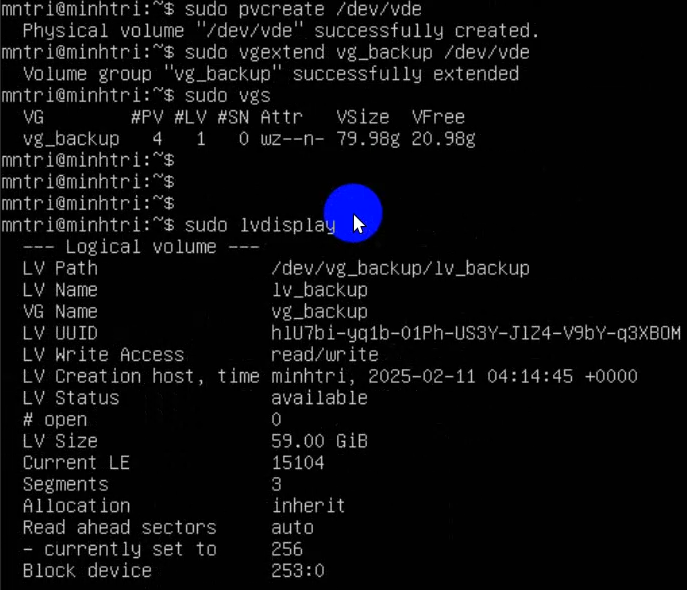


#### **3.2. Sử dụng LVM để add thêm ổ 20GB này vào LVM 60GB ở trên để tăng dung lượng thành 80GB**

Thêm ổ 20GB vào Volume Group (VG):

* Xác định ổ đĩa mới: **sudo fdisk -l**
* Tạo Physical Volume (PV) trên ổ mới: **sudo pvcreate /dev/vde**
* Thêm PV này vào Volume Group **vg\_backup**:   
  **sudo vgextend vg\_backup /dev/vde**
* Kiểm tra lại VG: **sudo vgs**

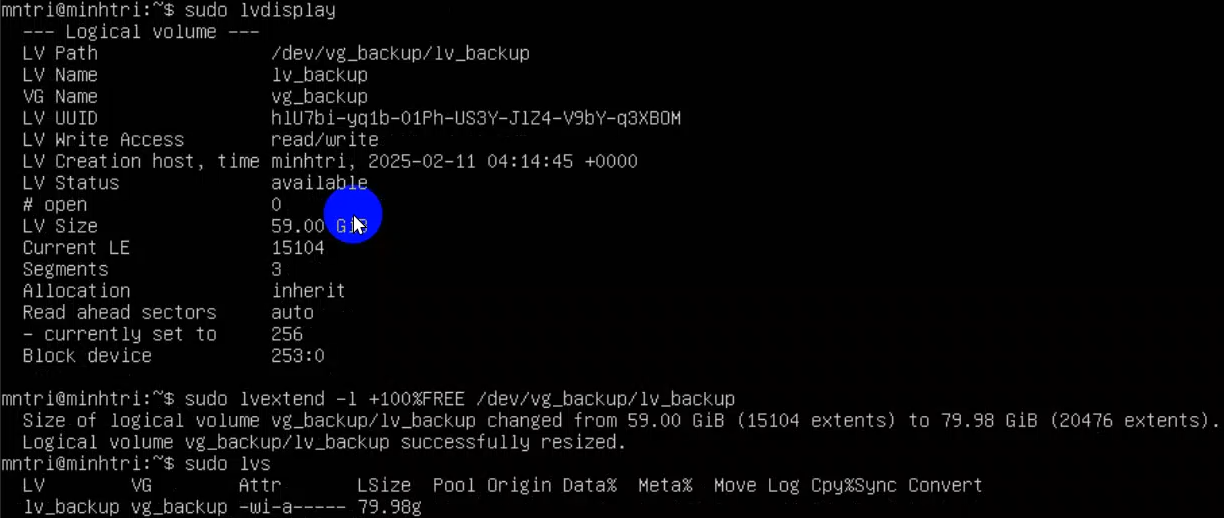




#### **3.3. Live resize mount point /backup thành 80GB mà không cần reboot lại server**

Mở rộng Logical Volume (LV) lên 80GB:

* Xác nhận tên của Logical Volume: **sudo lvdisplay**  
  /dev/vg\_backup/lv\_backup
* Mở rộng LV lên 80GB:  
  **sudo lvextend -l +100%FREE /dev/vg\_backup/lv\_backup**
* Kiểm tra lại LV: **sudo lvs**



Live Resize Filesystem EXT4 mà không cần reboot:

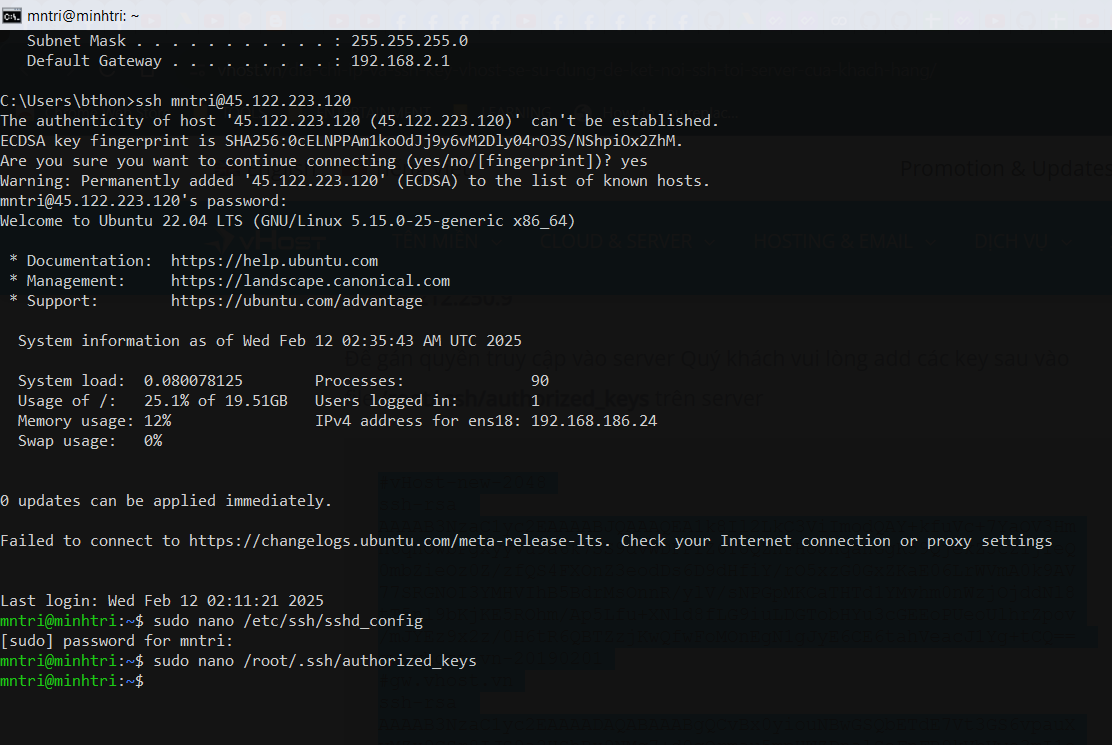
* Chạy lệnh sau để mở rộng filesystem mà không cần unmount /backup:   
  **sudo resize2fs /dev/vg\_backup/lv\_backup**
* Kiểm tra dung lượng mới của /backup: **df -hT**

## **SECURE SSH**

Yêu cầu:

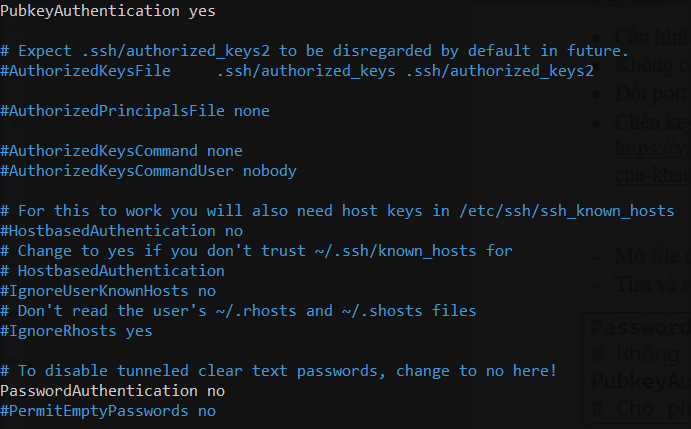
* Cấu hình bảo mật SSH chỉ cho phép login bằng SSH public key.
* Không cho phép login bằng password.
* Đổi port SSH từ 22 thành 2222.
* Chèn key SSH mặc định của vHost vào server sau khi đã cài đặt hoàn tất: <https://vhost.vn/dia-chi-ip-va-ssh-key-vhost-se-su-dung-de-ket-noi-ssh-toi-server-cua-khach-hang>

Để cho dễ thì em sẽ tạm thời kết nối SSH bằng CMD của máy laptop em trước

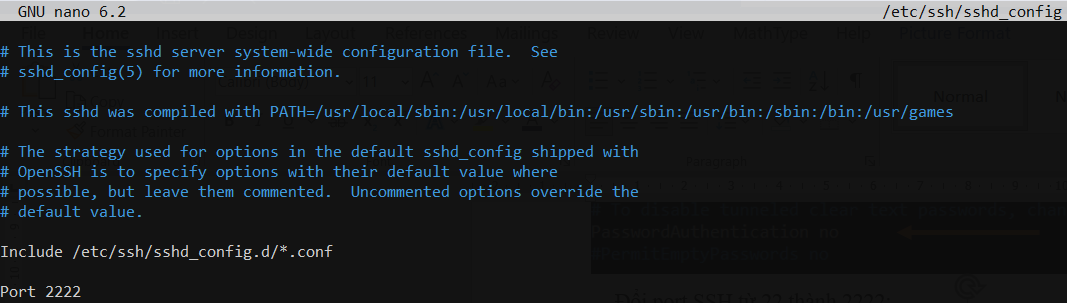


* Mở file cấu hình SSH: **sudo nano /etc/ssh/sshd\_config**
* Tìm và sửa các dòng sau:

PasswordAuthentication no   
# Không cho phép đăng nhập bằng mật khẩu  
PubkeyAuthentication yes   
# Cho phép đăng nhập bằng SSH Key

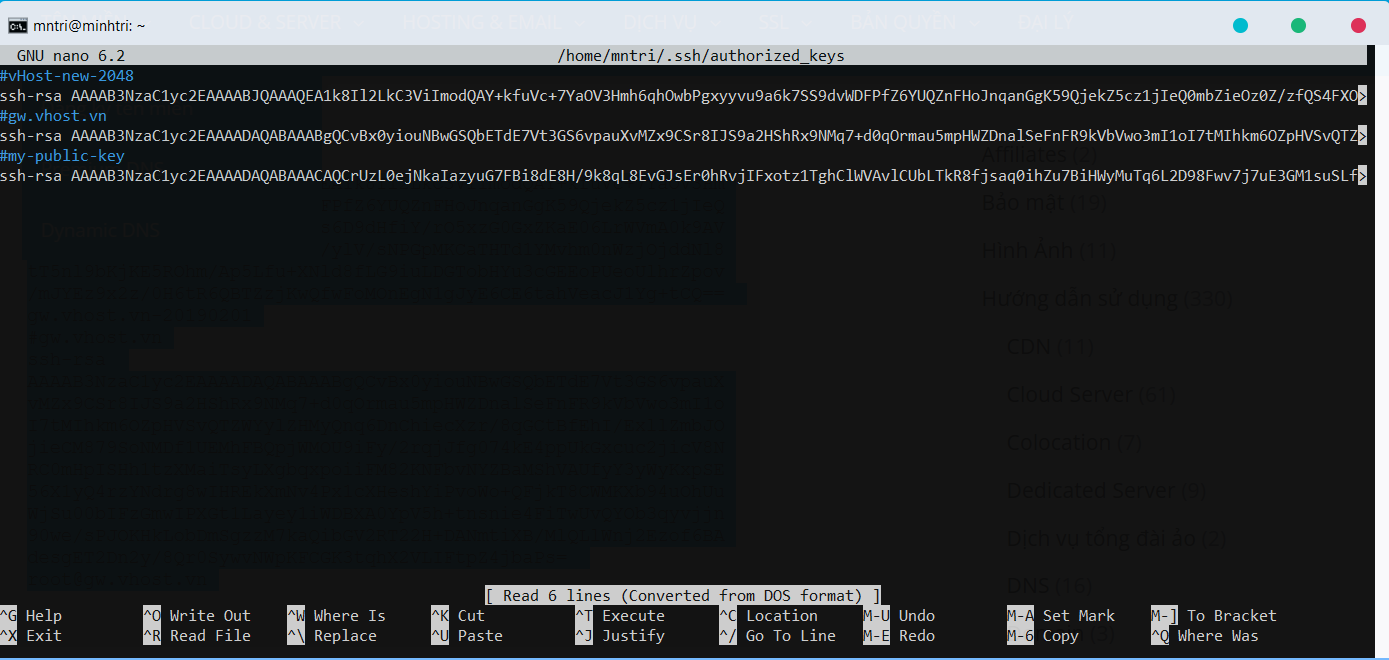


* Đổi port SSH từ 22 thành 2222:



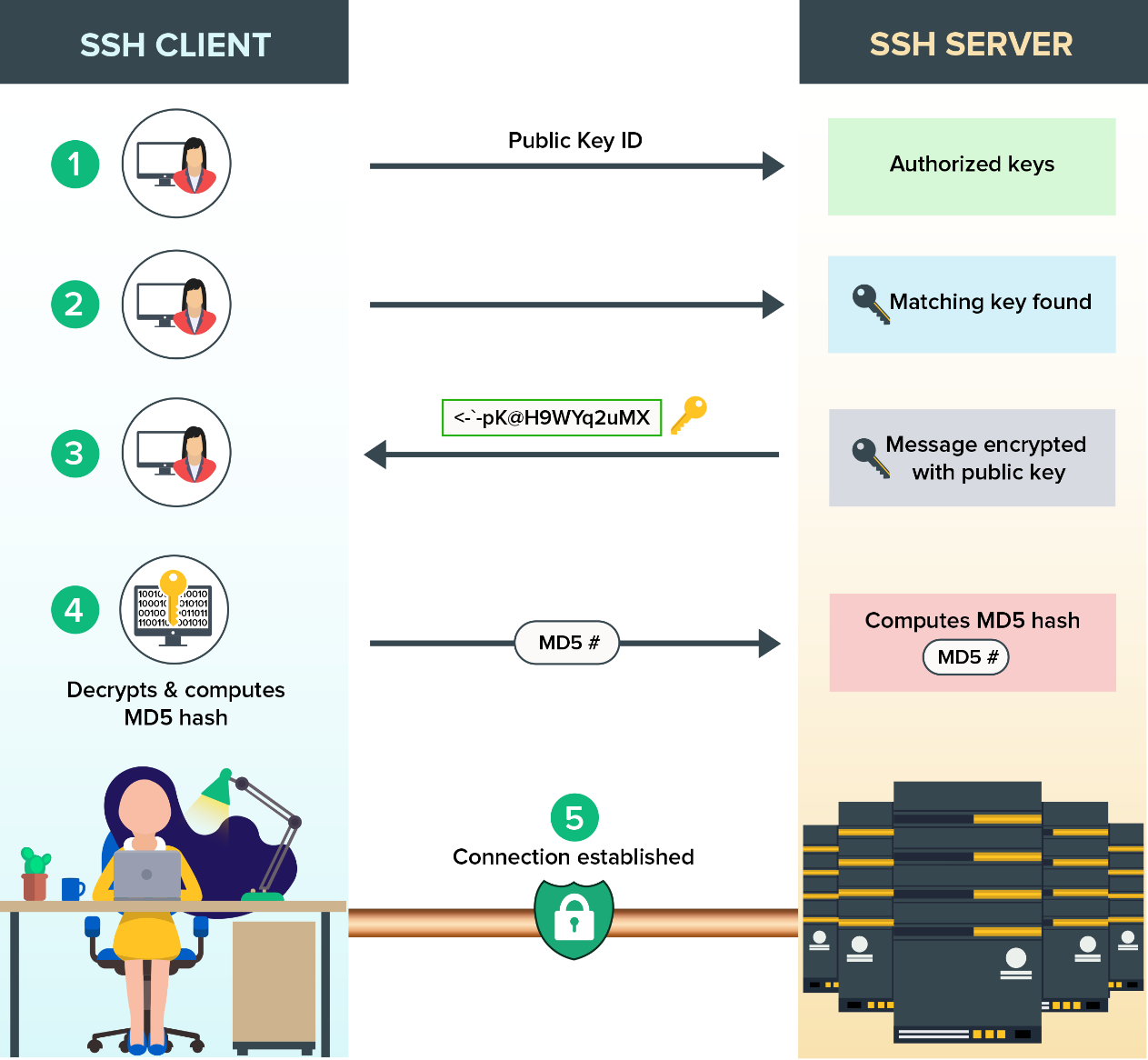
* Chèn key SSH mặc định của vHost vào server sau khi đã cài đặt hoàn tất:  
  **sudo nano /home/mntri/.ssh/authorized\_keys**

(Copy paste cả cái key trên đường link <https://vhost.vn/dia-chi-ip-va-ssh-key-vhost-se-su-dung-de-ket-noi-ssh-toi-server-cua-khach-hang/> vào)



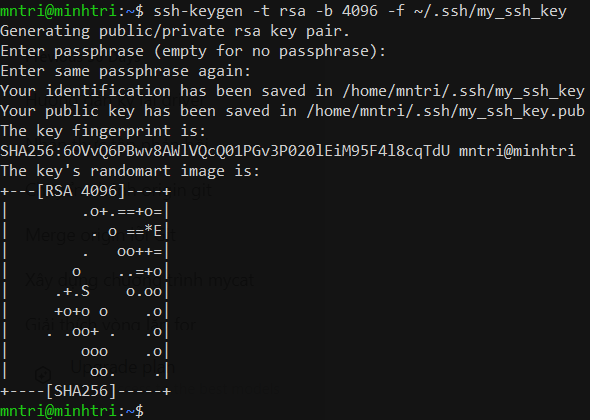
🔹 Nếu chỉ cần vHost SSH vào server, thì chỉ cần chèn key vHost vào ~/.ssh/authorized\_keys và không cần tạo key mới.

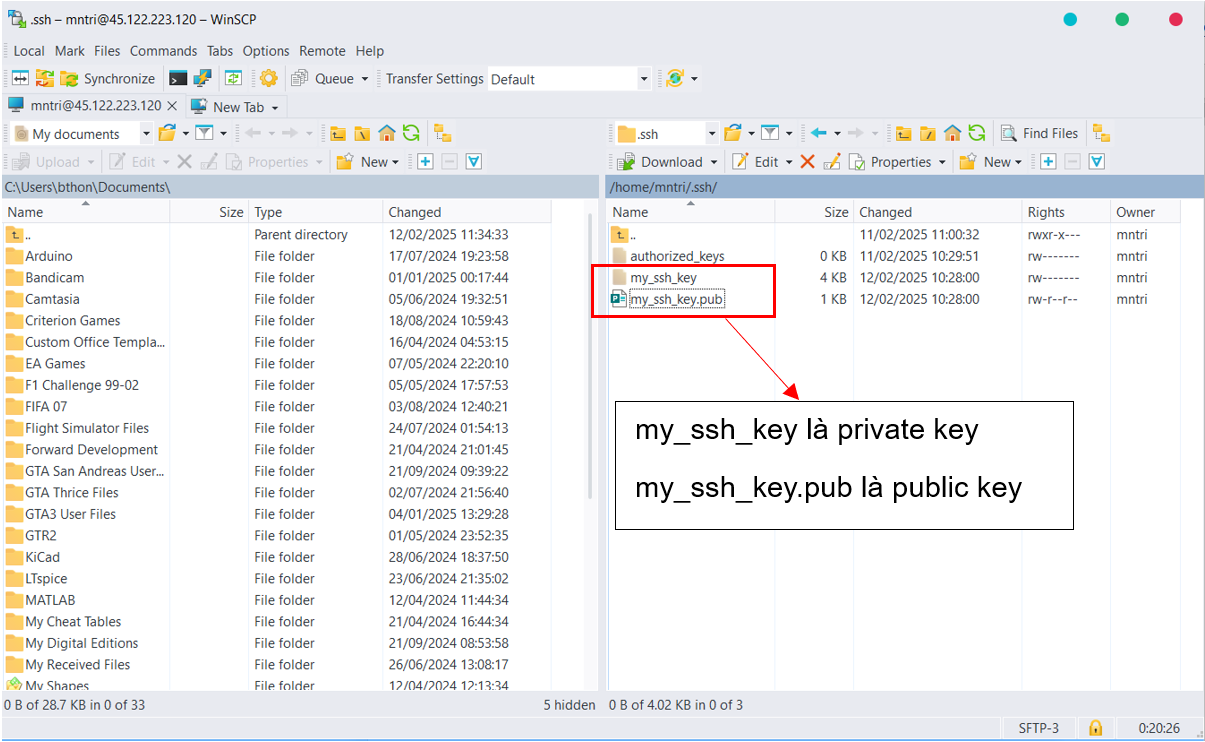
🔹 Nếu em muốn đăng nhập từ máy cá nhân, thì cần phải thêm public key của em vào server và em giữ 1 cái private key

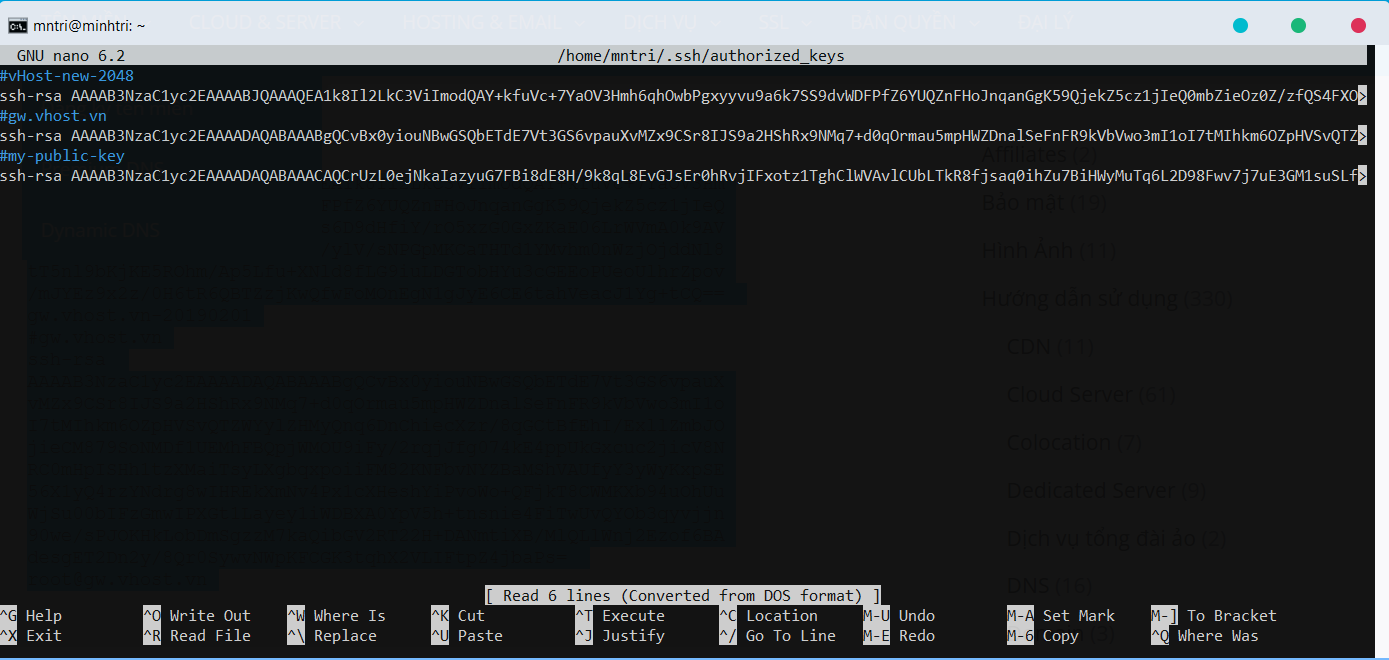


Tạo SSH Key trên máy cá nhân:

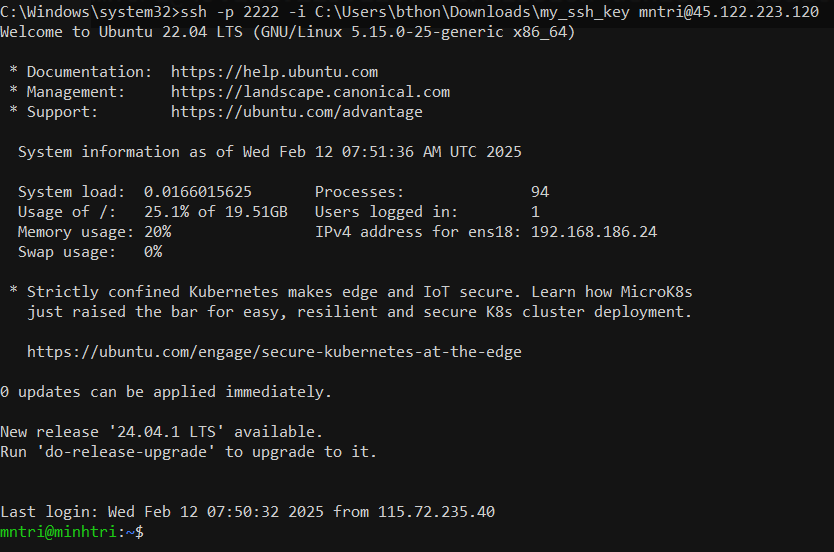
* Chạy lệnh: **ssh-keygen -t rsa -b 4096 -f ~/.ssh/my\_ssh\_key**
* Sau đó, copy my\_ssh\_key.pub (public key) vào server:  
  **sudo nano /home/mntri/.ssh/authorized\_keys**
* Ở máy laptop Windows của em, sử dụng WinSCP và truy cập vào máy ảo Ubuntu dùng SSH (PasswordAuthentication với PubkeyAuthentication tạm thời em disable nó bằng cách thêm dấu # phía trước lại). Sau đó em kiếm file my\_ssh\_key.pub (public key) & my\_ssh\_key (private key) rồi em copy qua máy Windows ở C:\Users\<user\_name>\Downloads
* Qua máy ảo, xóa dấu # để bật lại cái **PasswordAuthentication no** & **PubkeyAuthentication yes**
* Restart lại SSH: **sudo systemctl restart ssh**
* Kiểm tra lại:   
  **ssh -p 2222 -i C:\Users\<your\_windows\_user>\Downloads\my\_ssh\_key <user>@<server\_ip>**





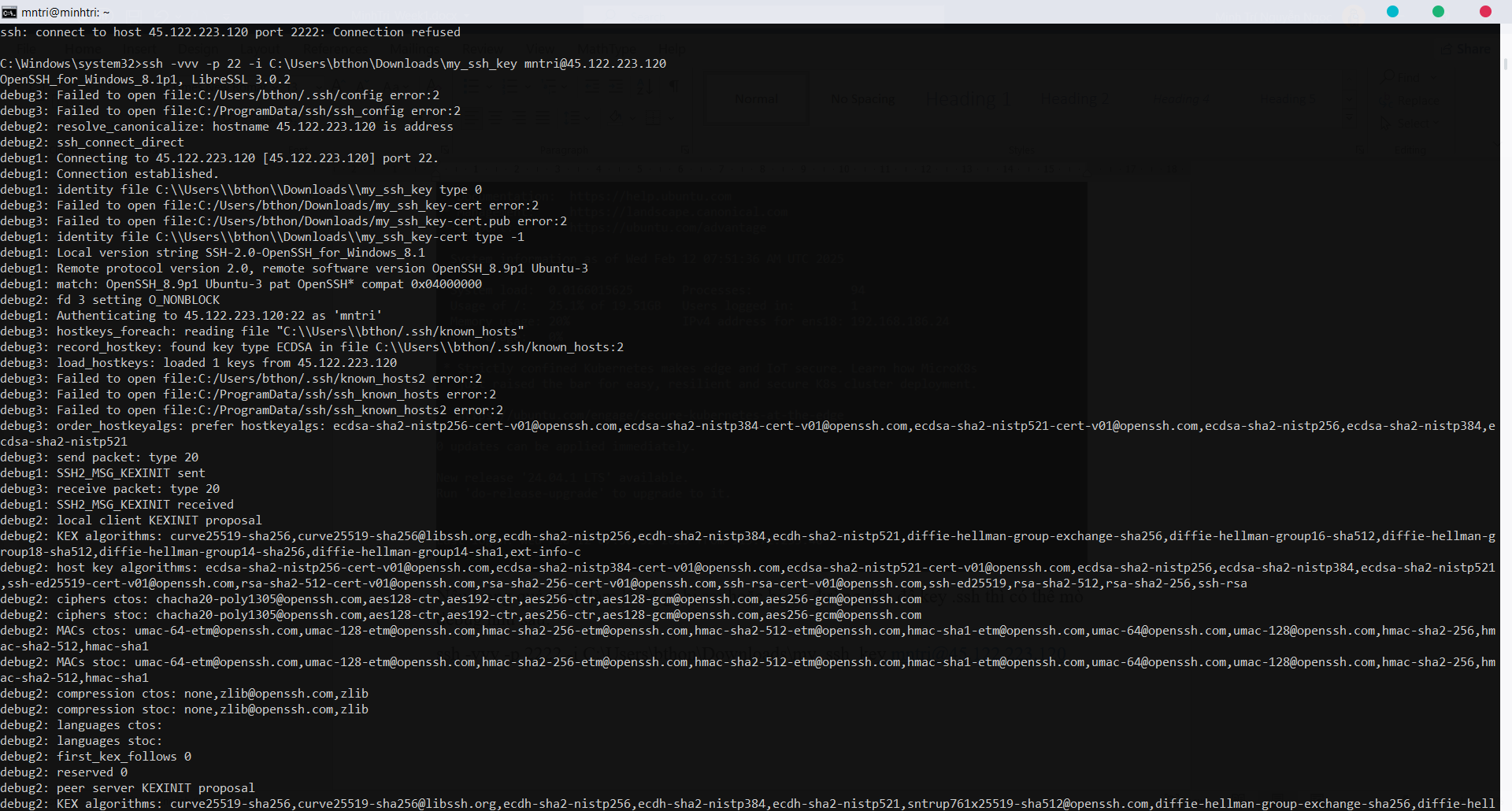






Trong trường hợp nếu làm lỡ có gì bị trục trặc hoặc nhập public key vào máy ảo bị nhầm đường dẫn (ví dụ như **/home/mntri/.ssh/authorized\_keys** mà nhập lộn key qua bên **/root/.ssh/authorized\_keys**) thì có thể mò debug như sau:

**ssh -vvv -p 2222 -i C:\Users\bthon\Downloads\my\_ssh\_key <user>@<server\_ip>**



Check cái dòng debug1: Offering public key… Ví dụ của em lúc làm bị:

debug1: Offering public key: C:\\Users\\bthon\\Downloads\\my\_ssh\_key RSA SHA256:6OVvQ6PBwv8AWlVQcQ01PGv3P020lEiM95F4l8cqTdU explicit

but the next lines are

debug3: send packet: type 50

debug2: we sent a publickey packet, wait for reply

debug3: receive packet: type 51

debug1: Authentications that can continue: publickey

debug2: we did not send a packet, disable method

debug1: No more authentication methods to try

Nghĩa là:

✅ Your SSH client is correctly offering the key.

❌ The server is rejecting it.

Tiếp đến, vô cái máy ảo Proxmox em gõ:

**ls -ld /home/mntri/.ssh**

**ls -l /home/mntri/.ssh/authorized\_keys**

Kết quả nên hiện ra như vậy

drwx------ 2 mntri mntri 4096 Feb 12 12:00 .ssh

-rw------- 1 mntri mntri 750 Feb 12 12:00 authorized\_keys

Nhưng khi em gõ ls -l /home/mntri/.ssh/authorized\_keys thì kết quả lại hiện ra như này:

-rw------- 1 mntri mntri 0 Feb 11 03:29 /home/mntri/.ssh/authorized\_keys

Nghĩa là authorized\_keys file đang bị trống rỗng (0 bytes) tức là không có file đó ấy.