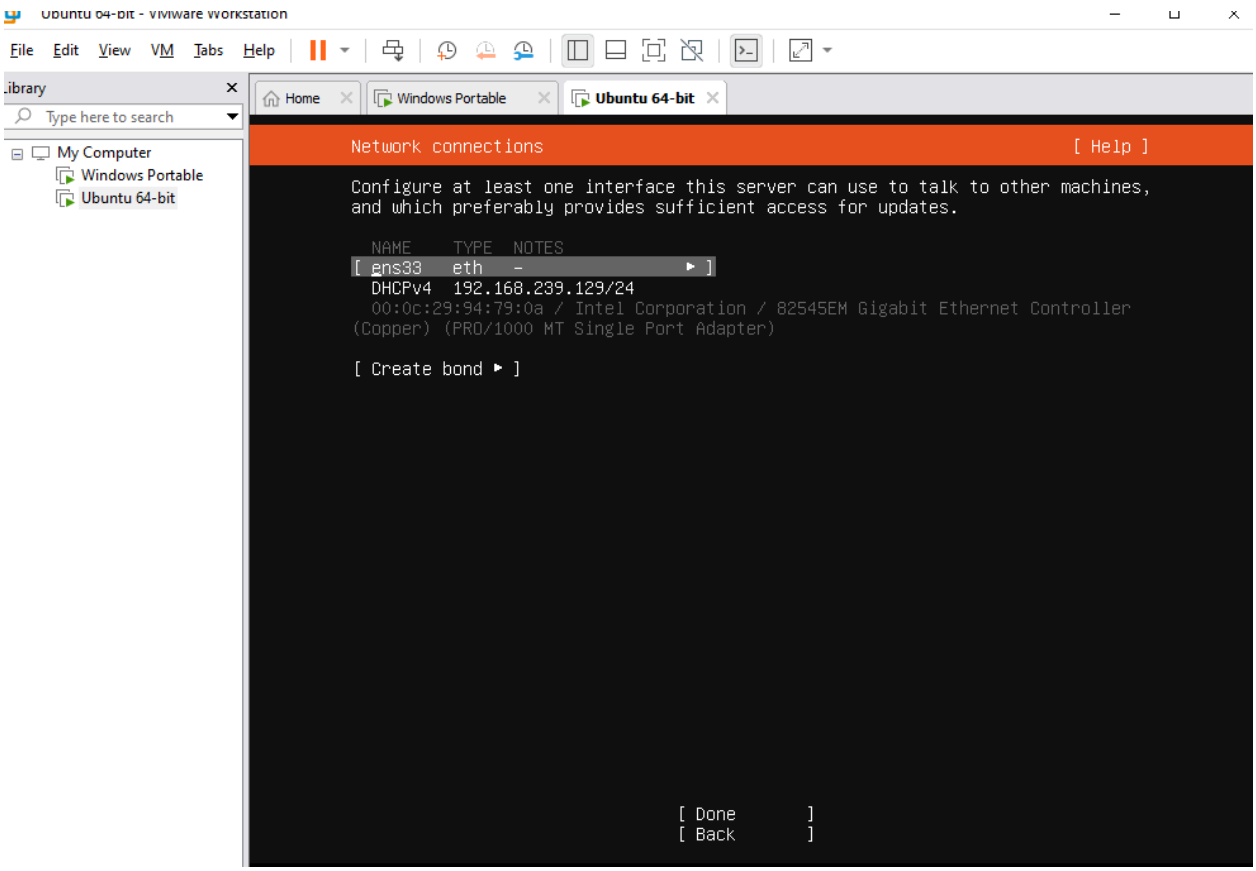
Set up your virtual environment

Download and install Ubuntu into a new VM

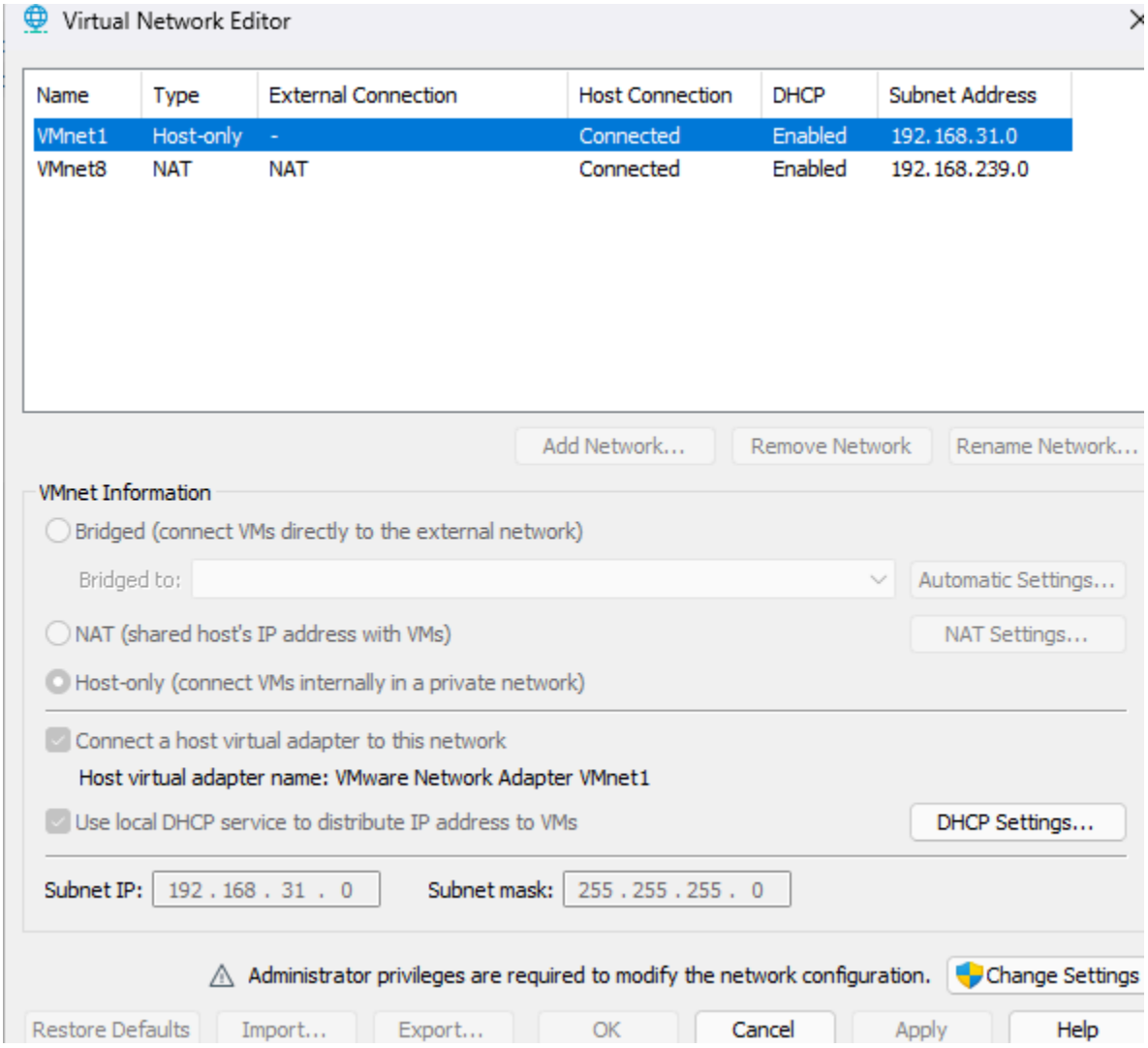
a. Download the Ubuntu Server 22.04.1 installer ISO.
    i. NOTE: I am specifying the **SERVER** version of Ubuntu because it comes preinstalled with necessary packages. If you choose the Desktop flavor, you will have issues, and you are wasting unnecessary resources.
b. Once downloaded, create a new VM in Workstation with the following specs
    i. Use the downloaded ISO as the installer image
    ii. 14GB Disk size
    iii. Customize Hardware
        - 2 CPU cores
        - 2GB RAM
    iv. During OS install, leave defaults unless otherwise specified
        - Use Tab to navigate, Space to check boxes, Enter to confirm
        - “Installer update available”
            a. “Continue without updating”



When you get to “Network connections” section above, we need to take a few steps to set a static IP address for this VM so that it doesn’t change throughout the lab or beyond it.

a. Find out the gateway IP of your VMware Workstation NAT network
    i. In VMware Workstation, click “Edit” menu at top
    ii. Click “Virtual Network Editor”
    iii. Select the “Type: NAT” network

iv. Click “NAT Settings…”

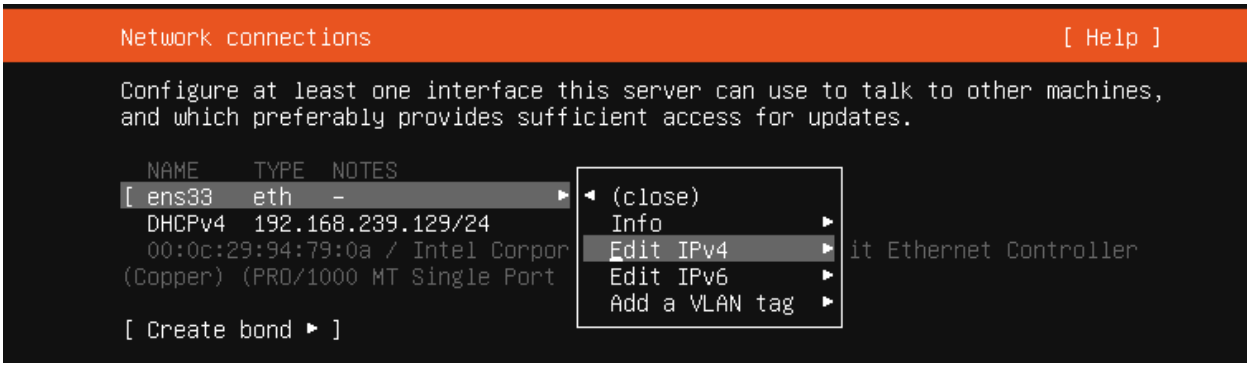


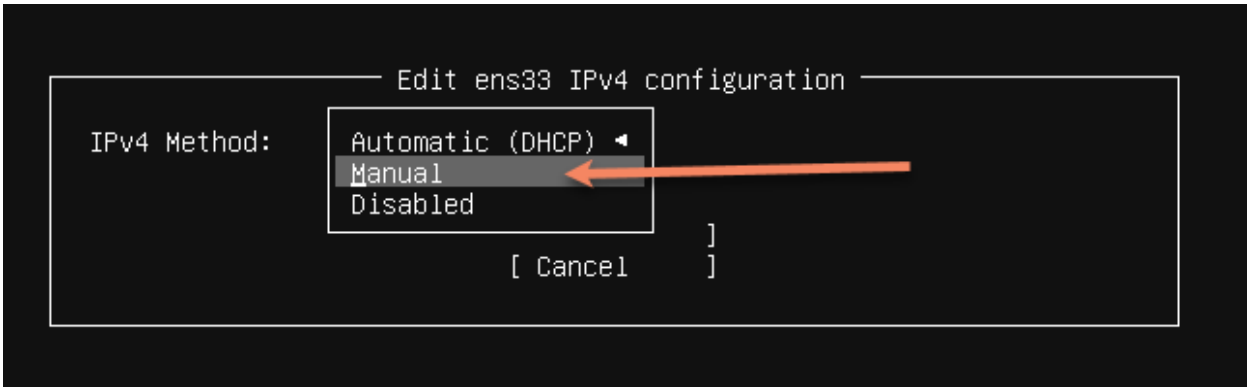
v. Copy down the “Subnet IP” & “Gateway IP”, we’ll need it in the next step

vi. 

vii. Close the NAT Settings and Virtual Network Editor windows.

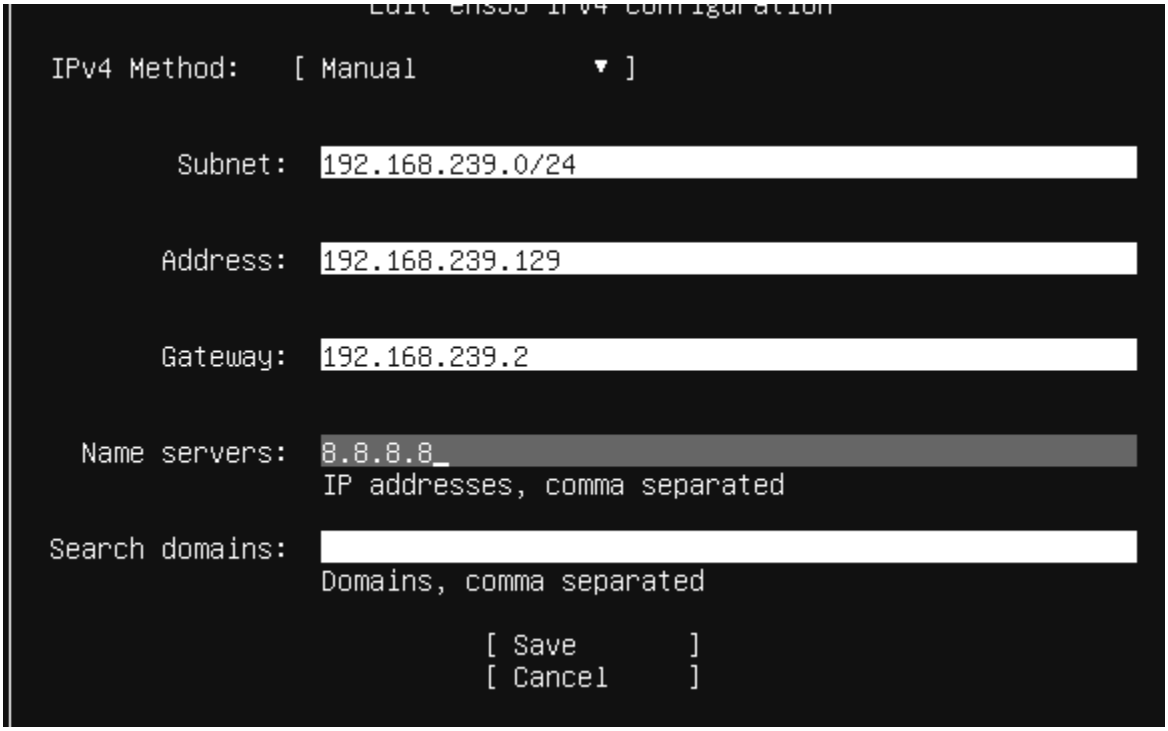
b. Now, back in the Ubuntu installer, let’s change the interface from DHCPv4 to Manual.

i. 

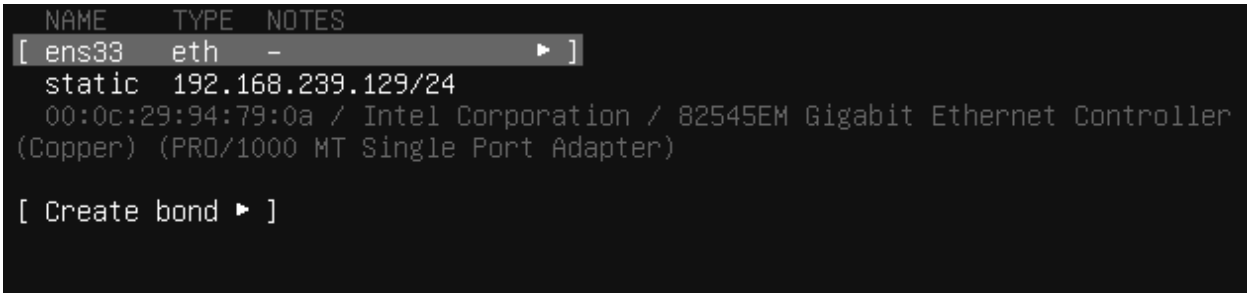
ii. 

iii. Be sure to carry forward the subnet and gateway IP from the previous step V., but adding the /24 to subnet IP. The

"Address" gets copied from what was previously assigned via DHCP.

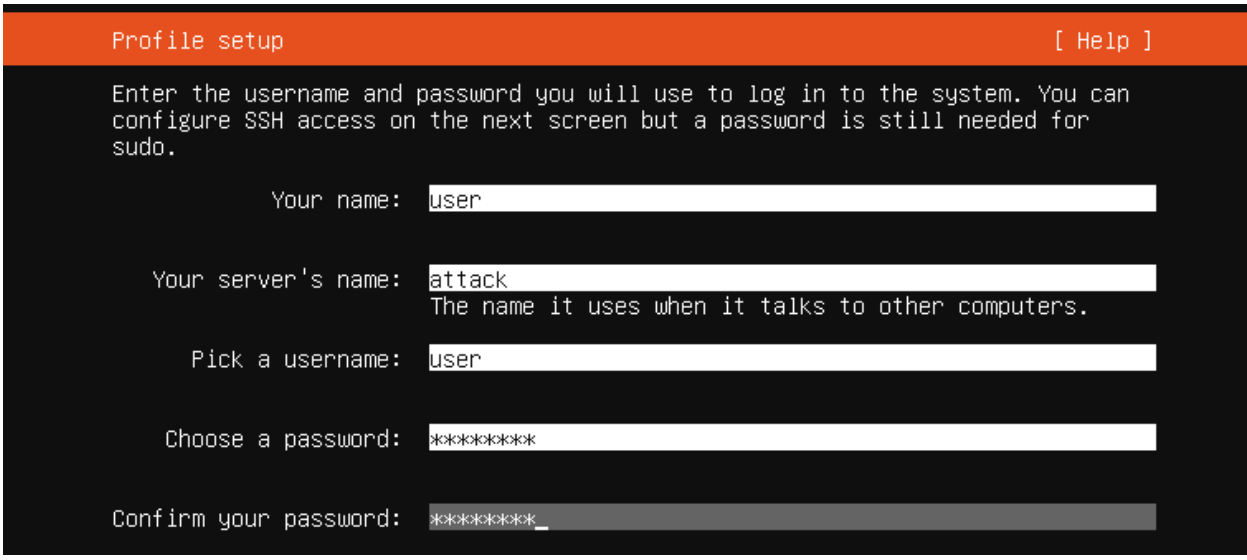
iv. 

v. When you're done, you should see this.

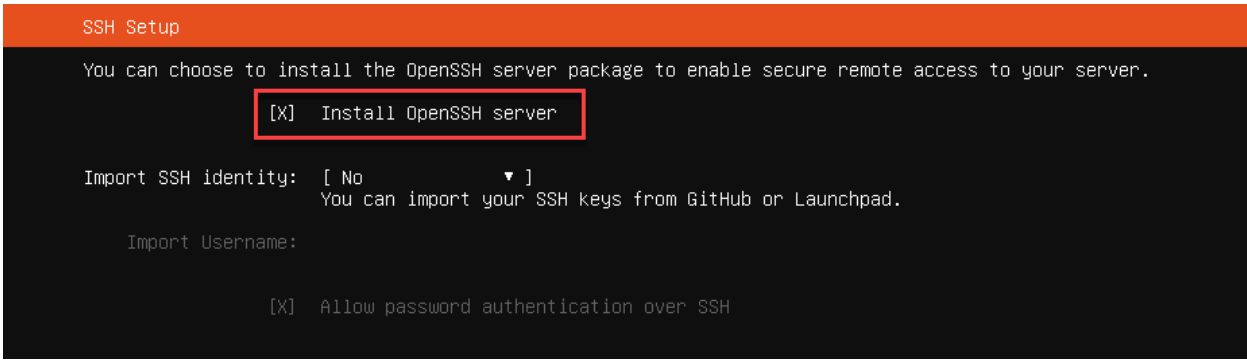
vi. 

    i. **NOTE**: Write down the Linux VM's IP address because you will need it multiple times throughout this

c. Set a memorable username/password (this is just a lab)
    i. Your name: `user`
    ii. Your server's name: `attack`
    iii. Username: `user`
    iv. Password: `password`
    v.



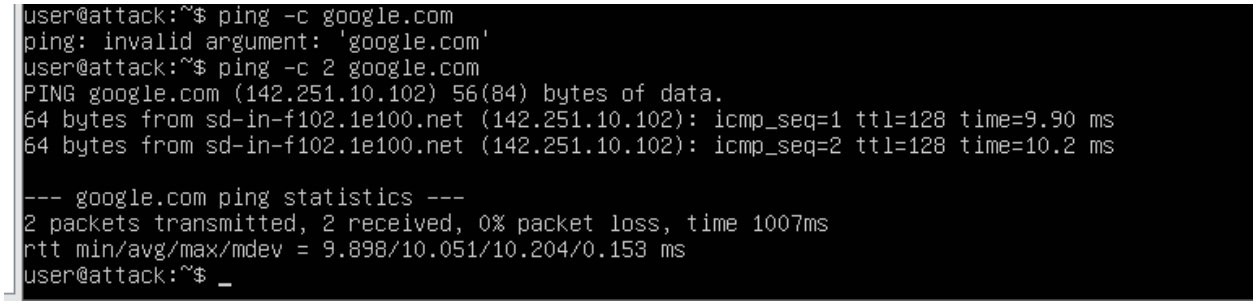
d. Install OpenSSH server?
    i. [check/yes]
    ii.



e. Continue installing OS until "Install complete!"
f. Hit Enter on [Reboot Now]
    i. If it hangs on "removing the CDROM" just press Enter
g. After the reboot, let's perform a quick connectivity check.

i. Logon with the credentials we defined during install
    i. Username: `user`
    ii. Password: `password`
ii. Make sure DNS and outbound pings are working

    Type in “ping -c 2 google.com”
iii. If your output resembles mine, you’re good to go.



Setup Attack System

We'll perform these steps from your host system, by using SSH to access the Linux VM.

Use an SSH client to access the Ubuntu VM so that you can easily copy/paste commands. MacOS/Linux/Modern Windows systems have built in SSH abilities, but there are third party tools for this as well.

1. Using the statically assigned IP address we copied down in the Linux VM installation process, let's SSH onto the VM from your host system to make future CLI activities easier thanks to copy/paste magic.
    a. I'll let you decide which SSH client to use, but from a modern Mac/Linux/Windows system, simply open a command prompt and run
    ssh user@192.168.239.129 (Use your linux IP address, this is mine.)
1. Now, from within this new SSH session, proceed with the following instructions to setup our attacker C2 server. First, let's drop into a root shell to make life easier.
    sudo su

Run the following commands to download Sliver, a Command & Control (C2) framework by BishopFox

Type in all this!

Download Sliver Linux server binary

Wget https://github.com/BishopFox/sliver/releases/download/v1.5.34/sliver-server_linux -O /usr/local/bin/sliver-server

Make it executable

chmod +x /usr/local/bin/sliver-server

install mingw-w64 for additional capabilities

apt install -y mingw-w64

Now let's create a working directory we'll use in future steps

Create our future working directory

mkdir -p /opt/sliver