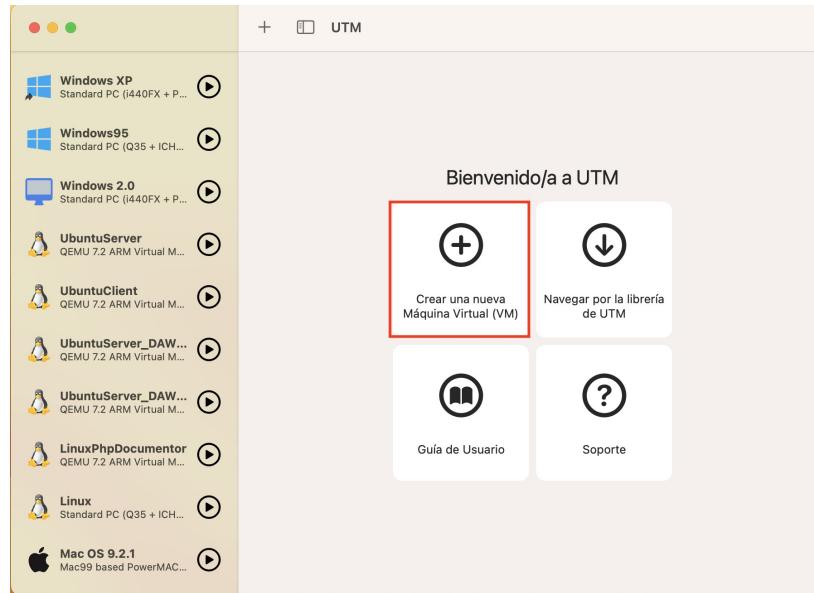


Installation of Ubuntu Server ARM on UTM Virtual Machine.

Once the .iso image file is downloaded, open UTM and select **Create a New Virtual Machine**.



Then, we choose to **virtualize** (when the architecture is ARM, we select virtualize; when the architecture is Intel or AMD, we select emulate).



Next, we select **Operating System → Linux**.



We click on **Search**.



And we search for the image we downloaded, in my case, **Ubuntu-22.10-live-server-arm64.iso**.



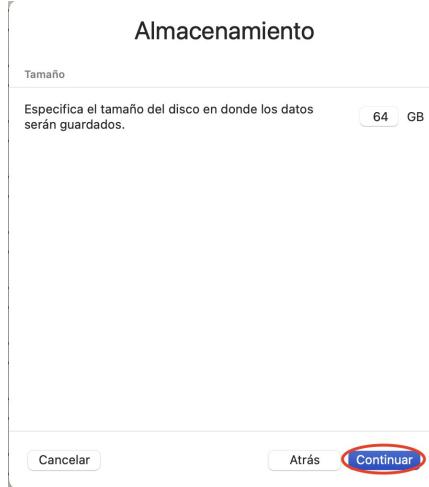
We leave the rest unselected and click **Continue**.



In my case, the MacBook has 16GB of RAM, so I leave it at 4GB. It's better to never exceed half of your computer's RAM.



Then, for the disk size, I usually leave it at the default 64GB, but you can set it to whatever amount you want. Keep in mind that part of it will be used by the operating system you are going to install. Click **Continue**.



In the **Shared Directory** window, we can set a directory where files will be shared between the virtual machine we're going to create and your computer.

A downside of UTM VM is that it doesn't allow you to drag a file from the host computer to the guest computer. So, if we want to share files between the two machines, we'll need to use a shared folder or a USB drive.

In this case, the directory can always be set later, so we leave it for now and click **Continue**.



Then, we change the name to whatever we prefer and click **Save**.

Resumen

Nombre Abrir configuración de la VM

Motor QEMU Utilizar la Virtualización

Arquitectura ARM64 (aarch64)

Sistema QEMU 7.2 ARM Virtual Machine (alias of vi)

RAM 4 GB

CPU Núcleos por defecto

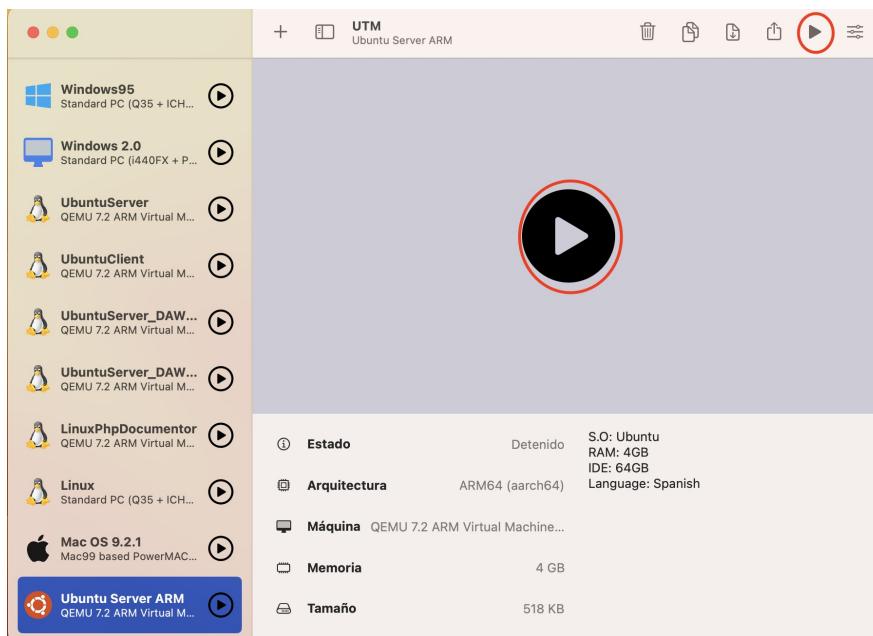
Almacenamiento 64 GB Aceleración de hardware OpenGL

Sistema operativo Linux Ignorar la imagen de arranque

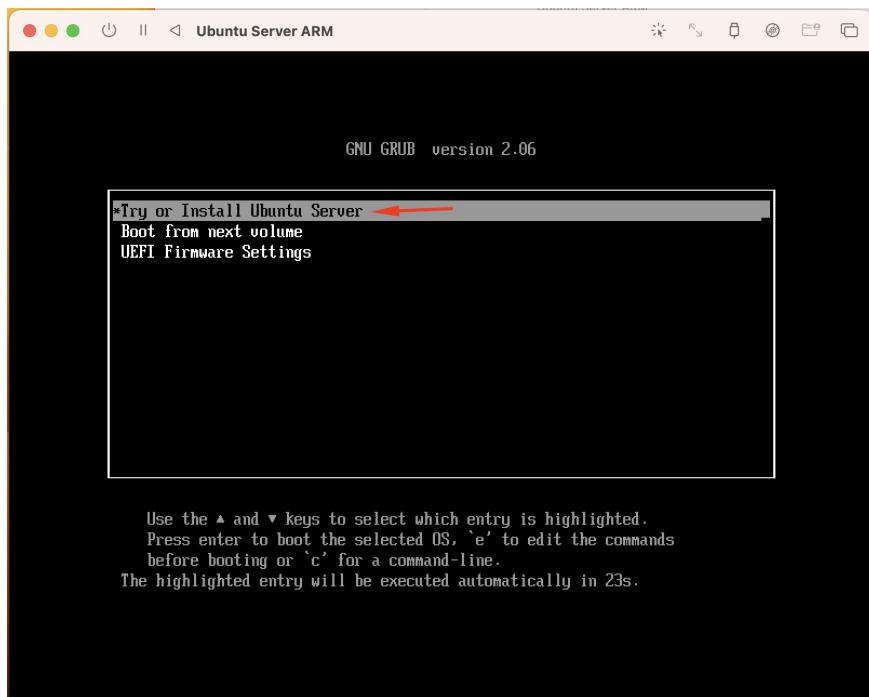
Imagen de arranque /Users/jorgecamachoochoa/Documents/qemu/Ubuntu Server ARM.qcow2

If we want to see the machine's configuration and haven't previously selected **Open VM Configuration**, we can open it later.

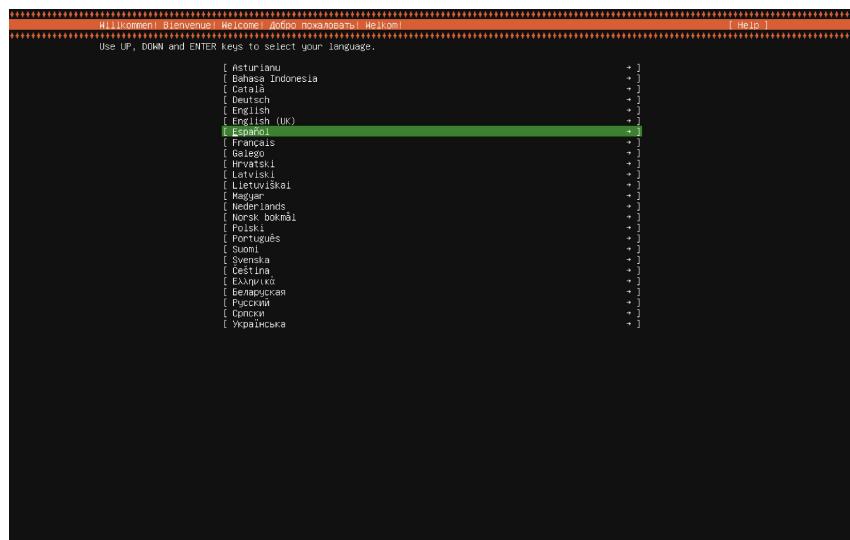
Click the **Play** button.



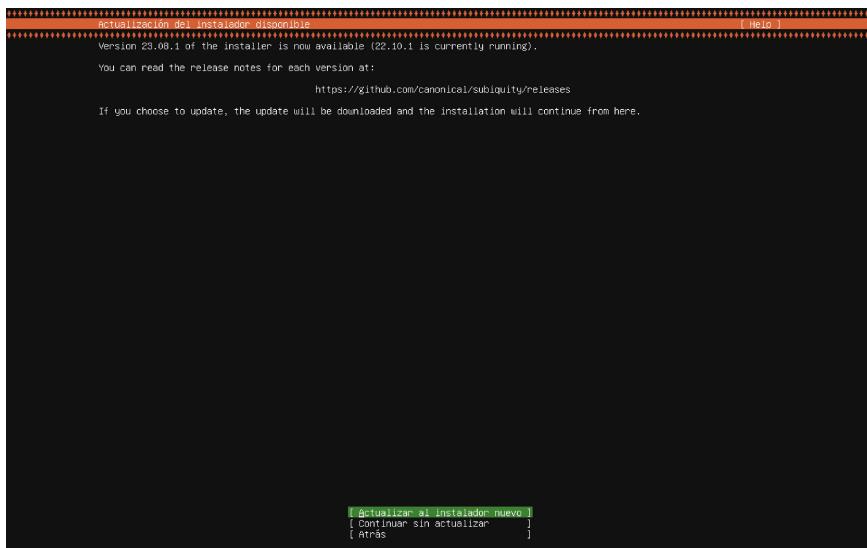
Select **Try or Install Ubuntu Server**.



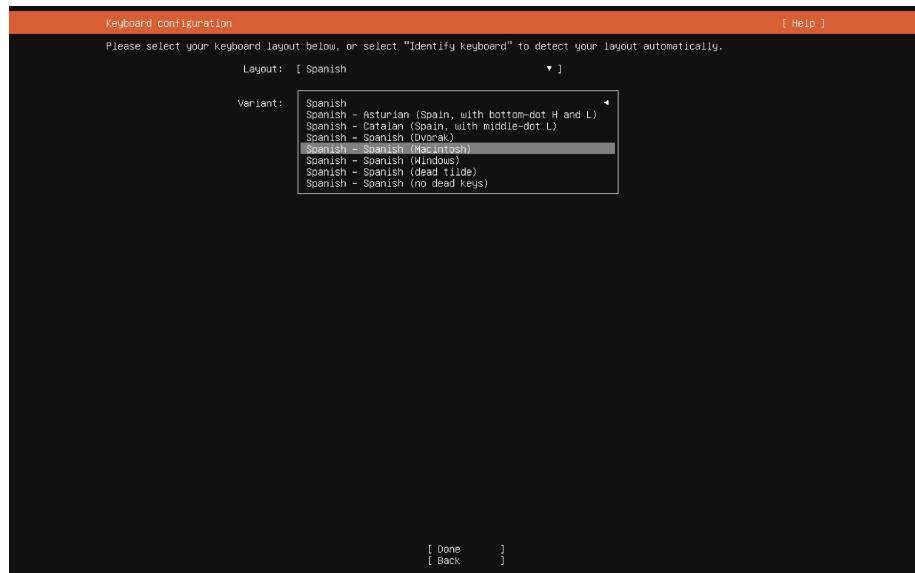
We use the cursor to select the language we want, in this case, **Spanish**, and press **Enter**. (To navigate through the terminal, we will use the **Tab** key and the arrow keys).



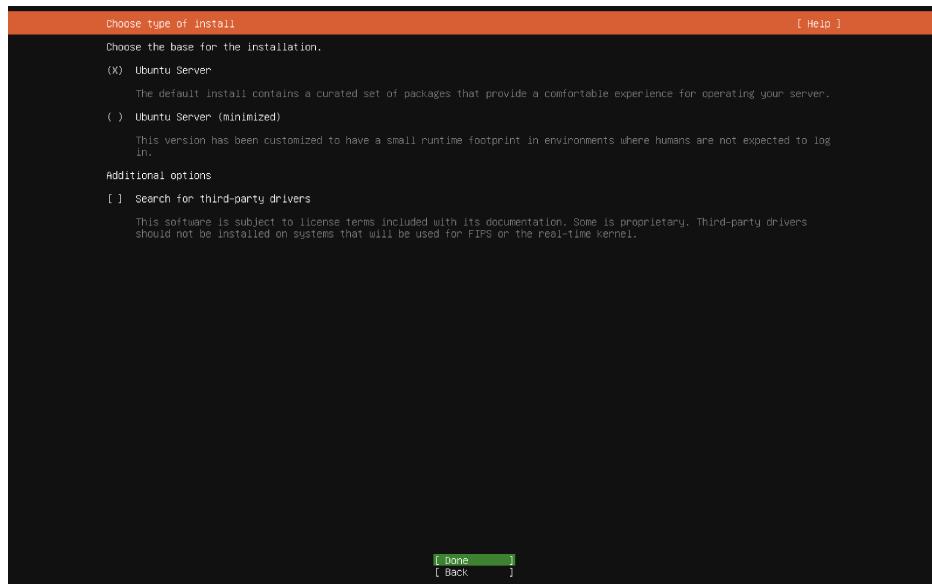
If the page appears indicating that an installer update is available, we can continue without updating or update to the new installer. Select the option you prefer and press **Enter**.



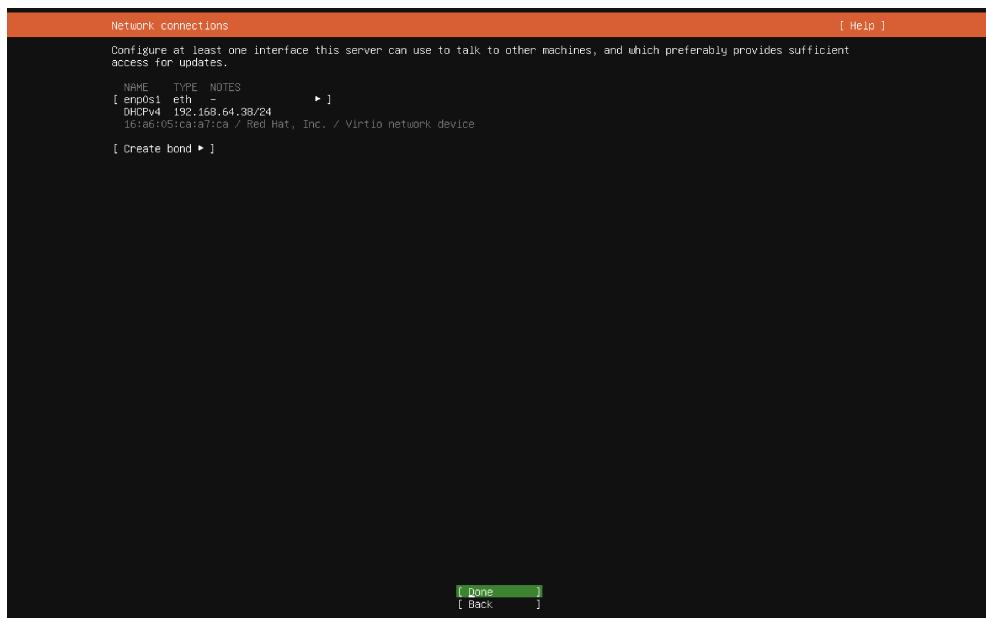
In **Keyboard**, select **Spanish – Spanish Macintosh** and click **Done**. You can also choose to have it detect the keyboard automatically, which is easier, or if both options show **Spanish**, we can leave it as is.



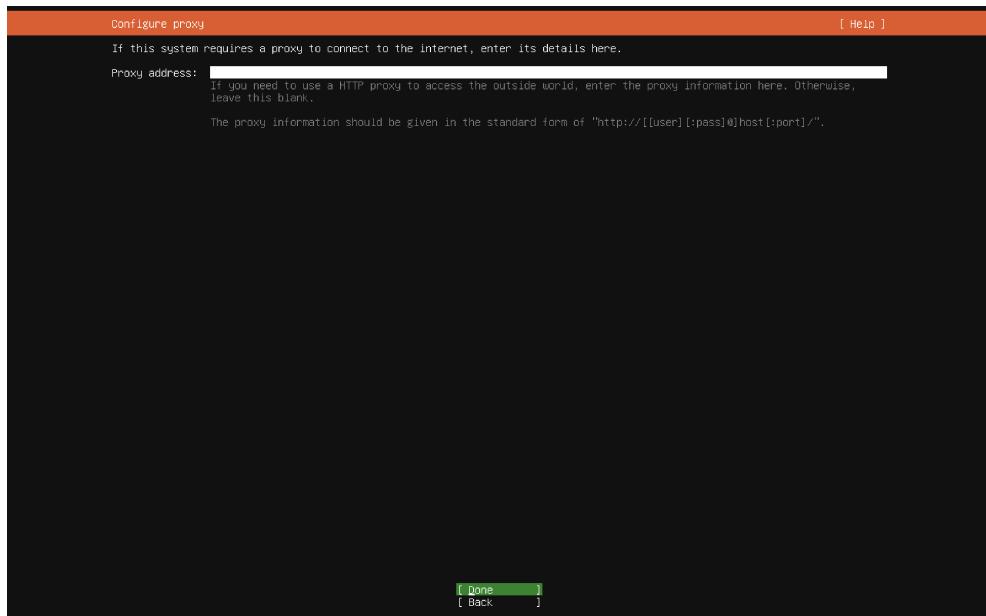
In **Select installation type**, leave it as is and select **Done**.



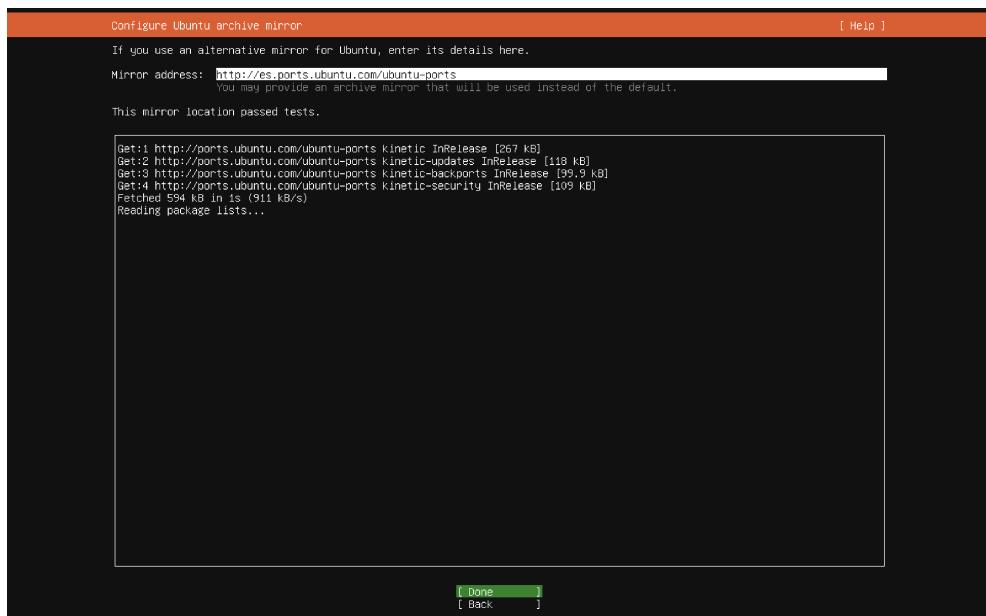
In Network connections, leave it as is and select Done.



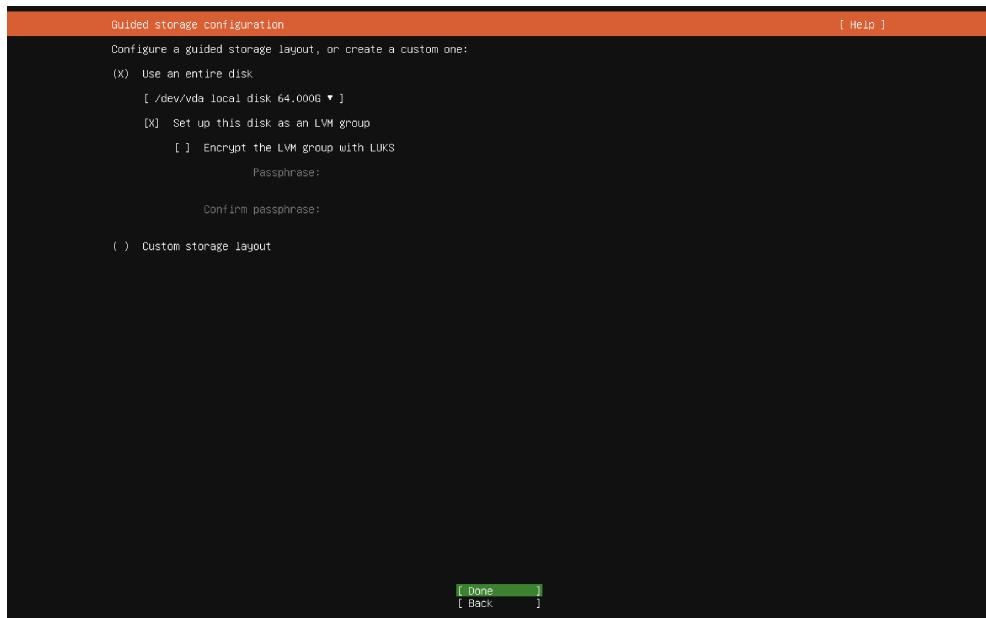
In Configure Proxy, leave it as is and select Done.



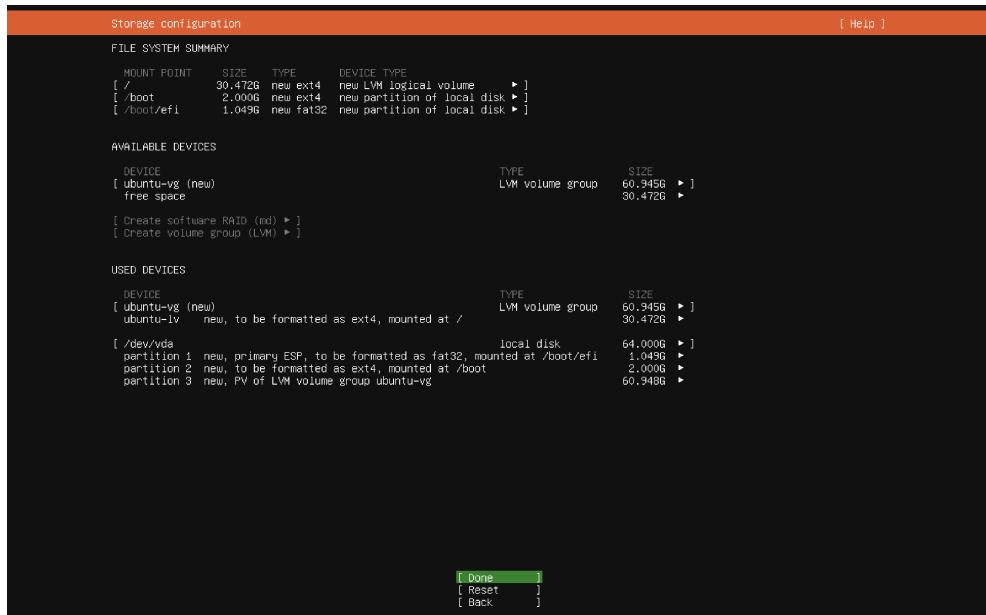
In **Configure mirror file**, leave it as is and select **Done**.



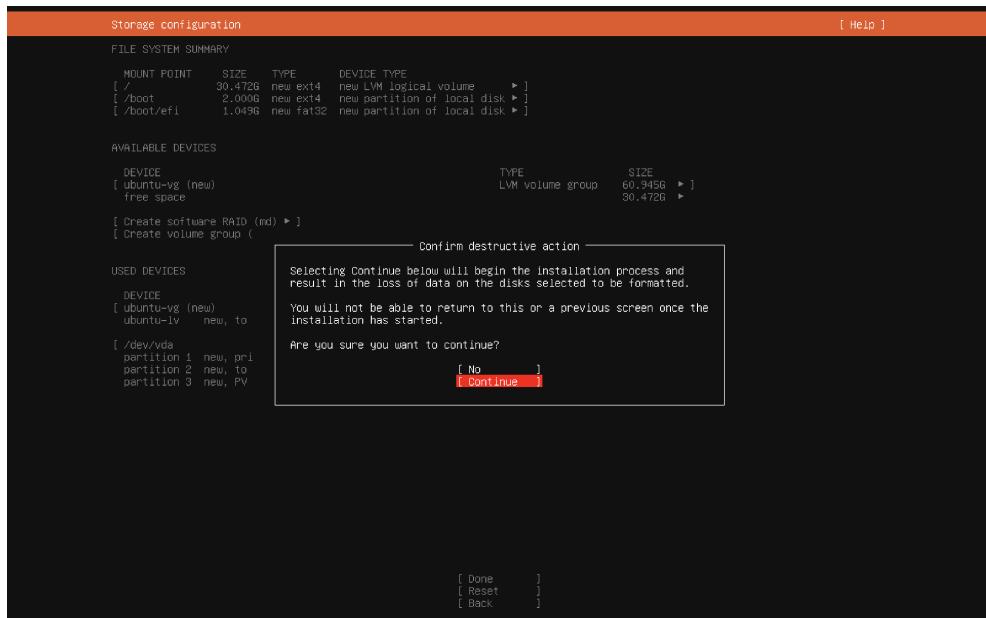
In **Guided storage configuration**, leave it as is and select **Done**.



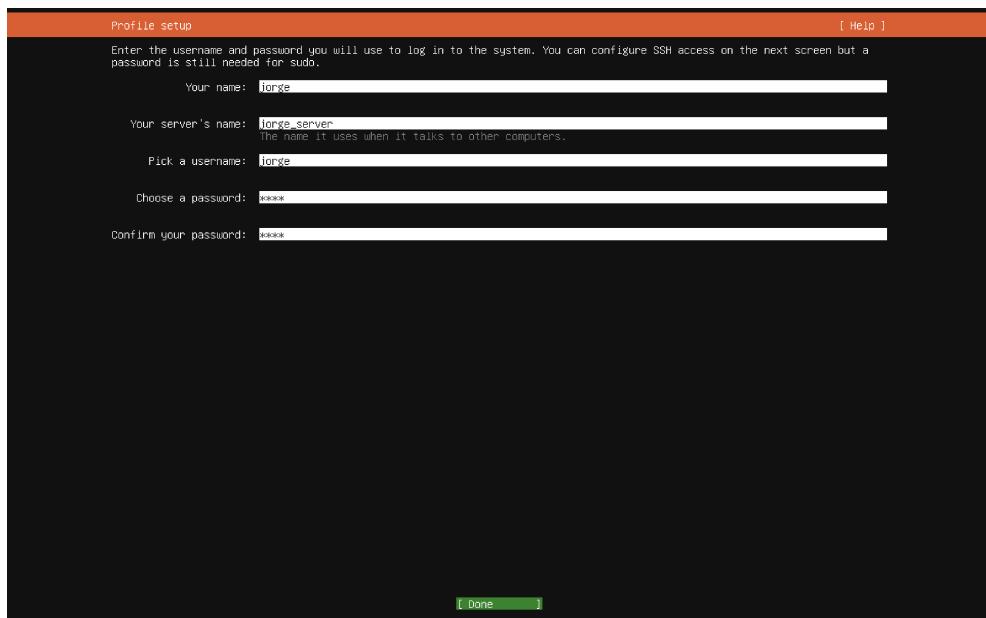
In **Storage configuration**, select **Done**.



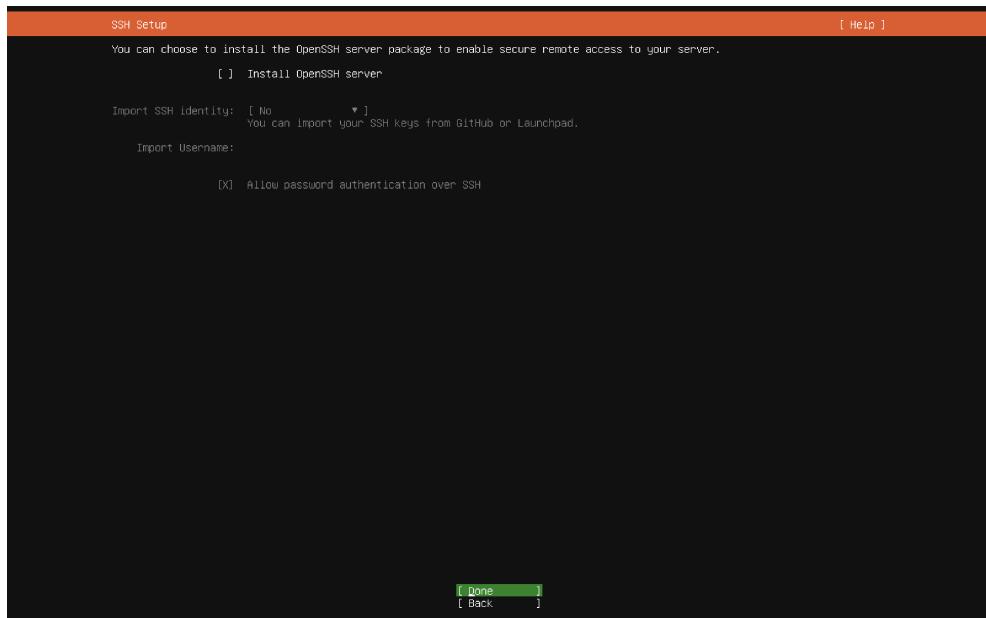
And in the pop-up window **Confirm destructive action**, select **Continue**.



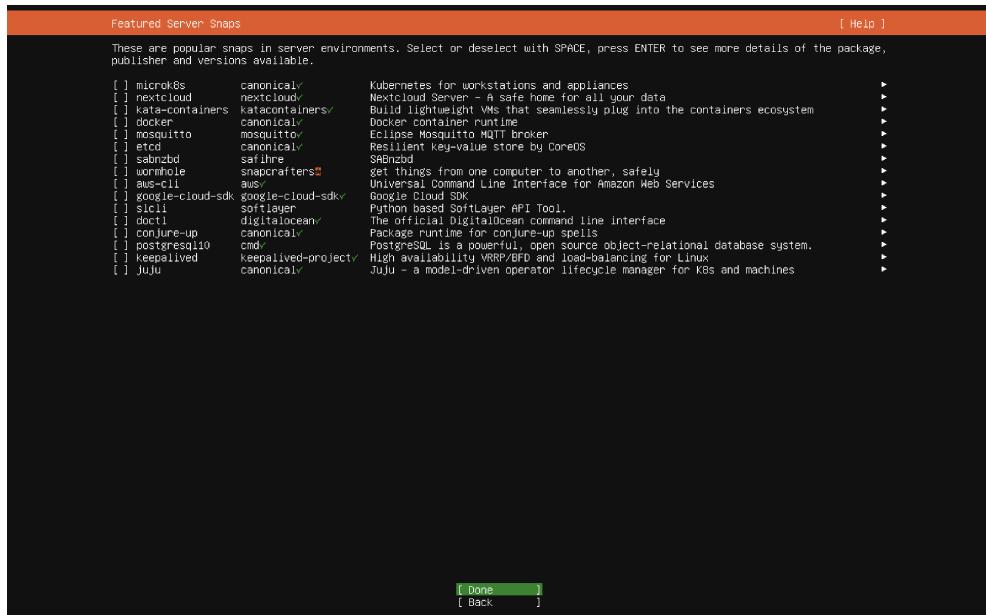
In **Profile setup**, fill in the different fields. To access later, we'll use the **Pick a username** and the **password** we set. Then select **Done**.



In **SSH Setup**, we don't select anything and click **Done**.



*In **Featured Server Snaps**, we don't select anything and click **Done**.*



Now we can see that the installation is in progress...

```
Install complete! [ Help ]  
  
configuring format: format-1  
configuring partition: partition-2  
configuring lvm_volvgroup: lvm_volvgroup-0  
configuring lvm_partition: lvm_partition-0  
configuring format: format-2  
configuring mount: mount-2  
configuring mount: mount-1  
configuring mount: mount-0  
executing curtin install extract step  
curtin command install  
writing install sources to disk  
running 'curtin extract'  
curtin command extract  
acquiring and extracting image from cp://tmp/tmpdd0zggaz/mount  
executing curtin install curthooks step  
curtin command install  
configuring installed system  
running 'curtin in-target -- setupcon --save-only'  
curtin command in-target  
running 'curtin curthooks'  
curtin command curthooks  
configuring apt  
installing missing packages  
Installing packages on target system: ['efibootmgr', 'grub-efi-arm64', 'grub-efi-arm64-signed', 'shim-signed']  
configuring iscsi service  
configuring raid (mdadm) service  
installing kernel  
setting up swap  
apply networking config  
writing etc/fstab  
configuring multipath  
updating packages on target system  
configuring pollinate user-agent on target  
updating intramfs configuration  
configuring target system bootloader  
installing grub to target devices  
final system configuration  
configuring cloud-init  
calculating extra packages to install  
downloading and installing security updates  
curtin command in-target  
restoring apt configuration  
curtin command in-target  
subiquity/Late/run  
  
[ View full log ] [ Cancel update and reboot ]
```

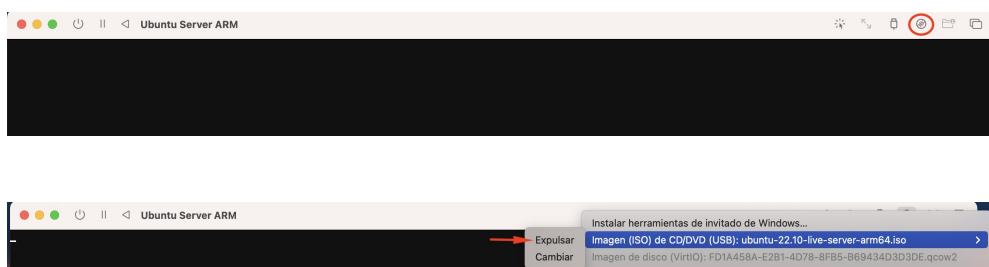
When **Reboot Now** appears at the very bottom, the installation is complete. Select it and press **Enter**.

```
Install complete! [ Help ]  
  
configuring lvm_partition: lvm_partition-0  
configuring format: format-2  
configuring mount: mount-2  
configuring mount: mount-1  
configuring mount: mount-0  
executing curtin install extract step  
curtin command install  
writing install sources to disk  
running 'curtin extract'  
curtin command extract  
acquiring and extracting image from cp://tmp/tmpdd0zggaz/mount  
executing curtin install curthooks step  
curtin command install  
configuring installed system  
running 'curtin in-target -- setupcon --save-only'  
curtin command in-target  
running 'curtin curthooks'  
curtin command curthooks  
configuring apt  
installing missing packages  
Installing packages on target system: ['efibootmgr', 'grub-efi-arm64', 'grub-efi-arm64-signed', 'shim-signed']  
configuring iscsi service  
configuring raid (mdadm) service  
installing kernel  
setting up swap  
apply networking config  
writing etc/fstab  
configuring multipath  
updating packages on target system  
configuring pollinate user-agent on target  
updating intramfs configuration  
configuring target system bootloader  
installing grub to target devices  
final system configuration  
configuring cloud-init  
calculating extra packages to install  
downloading and installing security updates  
curtin command in-target  
restoring apt configuration  
curtin command in-target  
subiquity/Late/run  
  
[ View full log ] [ Reboot Now ]
```

Now we see that the screen stays black.



Now we need to eject the disk, so we go to the disk icon in the top right corner of the virtual machine window.



And we click **Restart** on the virtual machine, located in the top left corner of the virtual machine window.



A pop-up window will appear, and we click **OK**.

After waiting a bit, the login screen appears—here we enter the **username** we previously set in **Pick a username**, and then the **password**.



And now the server terminal appears.

```
Ubuntu 22.10 jorgeserver tty1
jorgeserver login: jorge
Password:
Welcome to Ubuntu 22.10 (GNU/Linux 5.19.0-46-generic aarch64)

 * Documentation: https://help.ubuntu.com
 * Management: https://landscape.canonical.com
 * Support: https://ubuntu.com/advantage

System information as of Jue 31 ago 2023 19:29:28 UTC

System load: 0,013671875
Usage of /: 22,8% of 29,826G
Memory usage: 5%
Swap usage: 0%
Processes: 123
Users logged in: 0
IPv4 address for enp0s1: 192.168.64.38
IPv6 address for enp0s1: fd17:9f9a:2a22:c145:14a6:5ff:fea:a7ca

99 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

jorge@jorgeserver:~$
```

Now, we need to install the graphical interface to make it more familiar. To do this, we run the following command:

```
sudo apt install ubuntu-desktop
```

Then, enter the **password**.

```
Ubuntu 22.10 jorgeserver tty1
jorgeserver login: jorge
Password:
Welcome to Ubuntu 22.10 (GNU/Linux 5.19.0-46-generic aarch64)

 * Documentation: https://help.ubuntu.com
 * Management: https://landscape.canonical.com
 * Support: https://ubuntu.com/advantage

System information as of Jue 31 ago 2023 19:29:28 UTC

System load: 0,013671875
Usage of /: 22,8% of 29,826G
Memory usage: 5%
Swap usage: 0%
Processes: 123
Users logged in: 0
IPv4 address for enp0s1: 192.168.64.38
IPv6 address for enp0s1: fd17:9f9a:2a22:c145:14a6:5ff:fea:a7ca

99 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

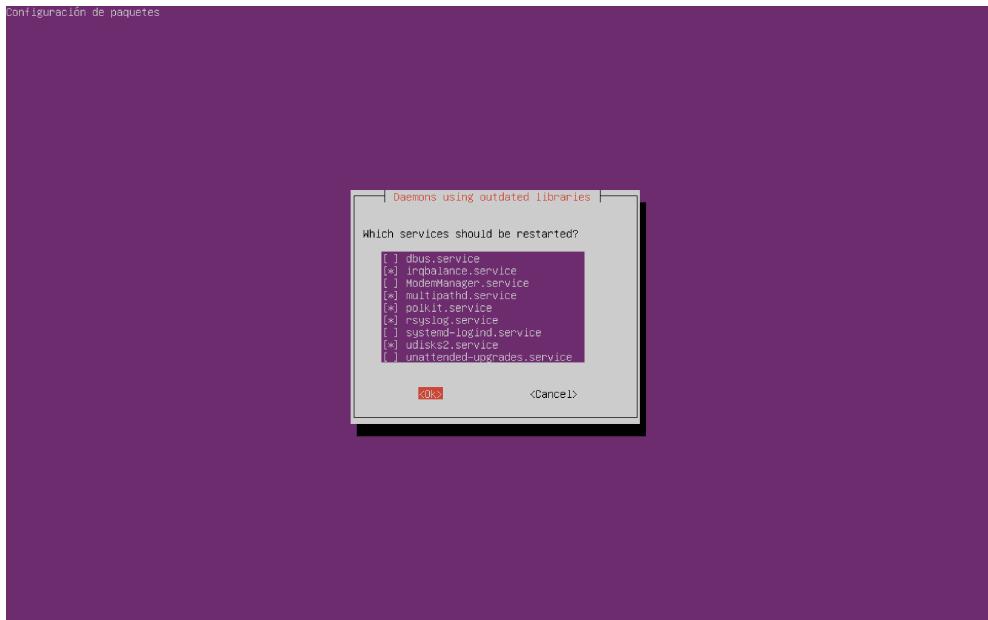
jorge@jorgeserver:~$ sudo apt install ubuntu-desktop
[sudo] password for jorge:
```

*A lot of information will appear, and it will ask: **Do you want to continue? [Y/n]** → select **Y** and press **Enter**.*

Now, we will see that the installation is running...

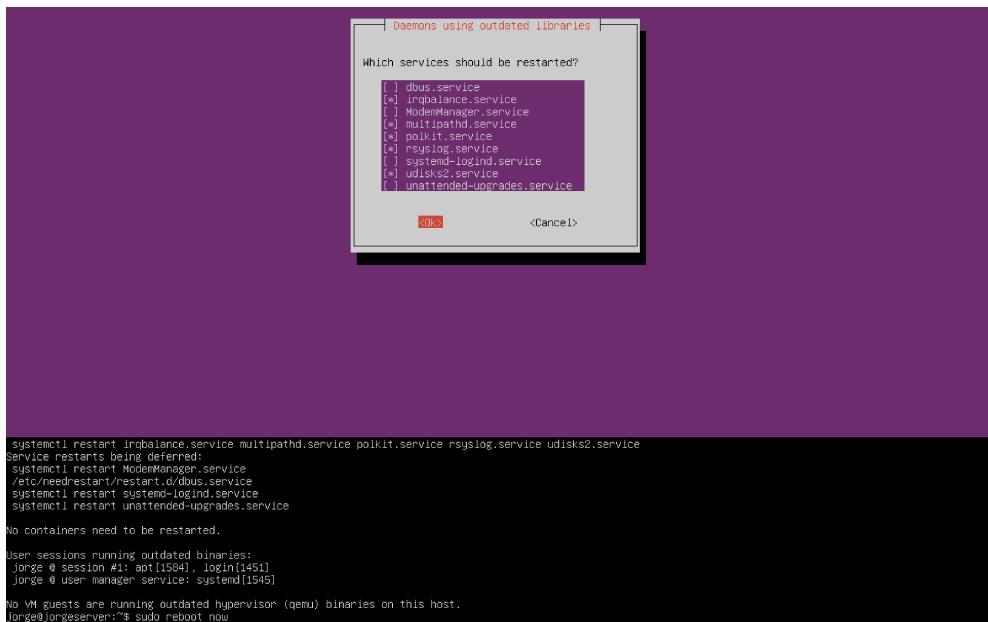
```
des200 http://ports.ubuntu.com/ubuntu-ports kinetic/main arm64 libspice-core arm64 2.46.0-1 [54.6 kB]
des201 http://ports.ubuntu.com/ubuntu-ports kinetic-updates/main arm64 avahi-autopki arm64 0.8+ubuntuu.22.10.1 [40.2 kB]
des202 http://ports.ubuntu.com/ubuntu-ports kinetic-updates/main arm64 avahi-libs arm64 0.8+ubuntuu.22.10.1 [25.4 kB]
des203 http://ports.ubuntu.com/ubuntu-ports kinetic/main arm64 libglibeprofile-1.0-0 arm64 1.10.8-1 [46.4 kB]
des204 http://ports.ubuntu.com/ubuntu-ports kinetic/main arm64 libcairo-script-interpreter2 arm64 1.16.0+ubuntuu [60.7 kB]
des205 http://ports.ubuntu.com/ubuntu-ports kinetic/main arm64 libcurl-common arm64 4.4.7-1+ubuntuu [543 kB]
des206 http://ports.ubuntu.com/ubuntu-ports kinetic/main arm64 libcurl-gnutls arm64 4.4.8-1+ubuntuu [2.197 kB]
des207 http://ports.ubuntu.com/ubuntu-ports kinetic/main arm64 libdibusui-1 arm64 1.2.0+ubuntuu2 [271 kB]
des208 http://ports.ubuntu.com/ubuntu-ports kinetic/main arm64 libbabbar arm64 43.0-1 [118 kB]
des209 http://ports.ubuntu.com/ubuntu-ports kinetic/main arm64 libbluez arm64 5.65+ubuntuu1 [1.090 kB]
des210 http://ports.ubuntu.com/ubuntu-ports kinetic-updates/main arm64 cups-common all 2.4.2-2ubuntuu.2 [254 kB]
des211 http://ports.ubuntu.com/ubuntu-ports kinetic-updates/main arm64 cups-clients arm64 2.4.2-2ubuntuu.2 [127 kB]
des212 http://ports.ubuntu.com/ubuntu-ports kinetic-updates/main arm64 cups-libs-utils arm64 2.4.2-2ubuntuu.2 [187 kB]
des213 http://ports.ubuntu.com/ubuntu-ports kinetic/main arm64 poppler-utils arm64 22.09.0+ubuntuu [187 kB]
des214 http://ports.ubuntu.com/ubuntu-ports kinetic/main arm64 libltdl0.2.3 arm64 0.12.0+ubuntuu [60.4 kB]
des215 http://ports.ubuntu.com/ubuntu-ports kinetic/main arm64 libltdl0.2.3-1 arm64 10.6.3-1 [59.8 kB]
des216 http://ports.ubuntu.com/ubuntu-ports kinetic-updates/main arm64 cups-filter-drivers arm64 2.18.16+ubuntuu.0.2 [150 kB]
des217 http://ports.ubuntu.com/ubuntu-ports kinetic-updates/main arm64 cups-core-drivers arm64 2.4.2-1ubuntuu.2 [283.6 kB]
des218 http://ports.ubuntu.com/ubuntu-ports kinetic/main arm64 libgb3d12 arm64 1.41-1 [56.0 kB]
des219 http://ports.ubuntu.com/ubuntu-ports kinetic/main arm64 liblbd12 arm64 0.35-15ubuntuu12 [16.1 kB]
des220 http://ports.ubuntu.com/ubuntu-ports kinetic/main arm64 liblbd12-common arm64 0.12+ubuntuu12 [5.8 kB]
des221 http://ports.ubuntu.com/ubuntu-ports kinetic-updates/main arm64 libgb3d12-common arm64 1.41-1+ubuntuu12 [5.8 kB]
des222 http://ports.ubuntu.com/ubuntu-ports kinetic-updates/main arm64 libgb3d12-0.2 arm64 0.35-15ubuntuu12 [4.949 kB]
des223 http://ports.ubuntu.com/ubuntu-ports kinetic-updates/main arm64 libgb3d12-0.2-1 arm64 1.41-1+ubuntuu12 [4.949 kB]
des224 http://ports.ubuntu.com/ubuntu-ports kinetic-updates/main arm64 liblfbfontend1 arm64 1.28.15+ubuntuu.0.2 [51.6 kB]
des225 http://ports.ubuntu.com/ubuntu-ports kinetic-updates/main arm64 cups-filters arm64 1.28.16+ubuntuu.0.2 [475 kB]
des226 http://ports.ubuntu.com/ubuntu-ports kinetic-updates/main arm64 cups-ppdc arm64 2.4.2-1ubuntuu.2 [93.2 kB]
des227 http://ports.ubuntu.com/ubuntu-ports kinetic-updates/main arm64 cups-server-common all 2.4.2-1ubuntuu.2 [399 kB]
des228 http://ports.ubuntu.com/ubuntu-ports kinetic-updates/main arm64 cups arm64 2.4.2-1ubuntuu.2 [265 kB]
des229 http://ports.ubuntu.com/ubuntu-ports kinetic/main arm64 libbluez-cups arm64 5.65+ubuntuu [25.7 kB]
des230 http://ports.ubuntu.com/ubuntu-ports kinetic/main arm64 libbluez-glib arm64 5.65+ubuntuu [25.7 kB]
des231 http://ports.ubuntu.com/ubuntu-ports kinetic/main arm64 libbluez-oxide arm64 5.65+ubuntuu1 [227 kB]
des232 http://ports.ubuntu.com/ubuntu-ports kinetic/main arm64 branding-ubuntu all 1.10 [303 kB]
des233 http://ports.ubuntu.com/ubuntu-ports kinetic/main arm64 bubblewrap arm64 0.6.2-1 [45.5 kB]
des234 http://ports.ubuntu.com/ubuntu-ports kinetic/main arm64 bzip arm64 1.0.8-5ubuntuu1 [34.6 kB]
des235 http://ports.ubuntu.com/ubuntu-ports kinetic/main arm64 liblbind1 arm64 1.5.0-1 [59.2 kB]
des236 http://ports.ubuntu.com/ubuntu-ports kinetic-updates/main arm64 liblbiglap-meson arm64 22.2.5+ubuntuu.1 [64.7 kB]
des237 http://ports.ubuntu.com/ubuntu-ports kinetic/main arm64 liblbiglap-0 arm64 1.15-1 [7.244 kB]
des238 http://ports.ubuntu.com/ubuntu-ports kinetic/main arm64 liblbiglap-dl12-0 arm64 1.15-1 [7.244 kB]
des239 http://ports.ubuntu.com/ubuntu-ports kinetic/main arm64 liblbiglap-dl12-1 arm64 1.15-1 [7.244 kB]
des240 http://ports.ubuntu.com/ubuntu-ports kinetic/main arm64 liblbiglap-presento arm64 1.15-1 [5.732 kB]
des241 http://ports.ubuntu.com/ubuntu-ports kinetic/main arm64 liblbigcb-synch1 arm64 1.15-1 [9.412 kB]
des242 http://ports.ubuntu.com/ubuntu-ports kinetic/main arm64 liblbigcb-xt1exec arm64 1.15-1 [10.3 kB]
des243 http://ports.ubuntu.com/ubuntu-ports kinetic/main arm64 liblxshfmcneff arm64 1.3+ubuntuu.0 [5.444 kB]
des244 http://ports.ubuntu.com/ubuntu-ports kinetic/main arm64 liblxsf16xml arm64 1:1.1.4+ubuntuu3 [10.3 kB]
des245 http://ports.ubuntu.com/ubuntu-ports kinetic/main arm64 liblbbdm-andgou2 arm64 2.4.113-2 [19.6 kB]
des246 http://ports.ubuntu.com/ubuntu-ports kinetic/main arm64 liblbbdm-nouveau2 arm64 2.4.113-2 [17.5 kB]
des247 http://ports.ubuntu.com/ubuntu-ports kinetic/main arm64 liblbbdm-redone01 arm64 2.4.113-2 [20.5 kB]
des248 http://ports.ubuntu.com/ubuntu-ports kinetic-updates/main arm64 liblxflvms arm64 1:15.0-7+ubuntuu.0.22.10.1 [24.0 MB]
```

*In the package configuration, leave everything as is and click **OK**.*

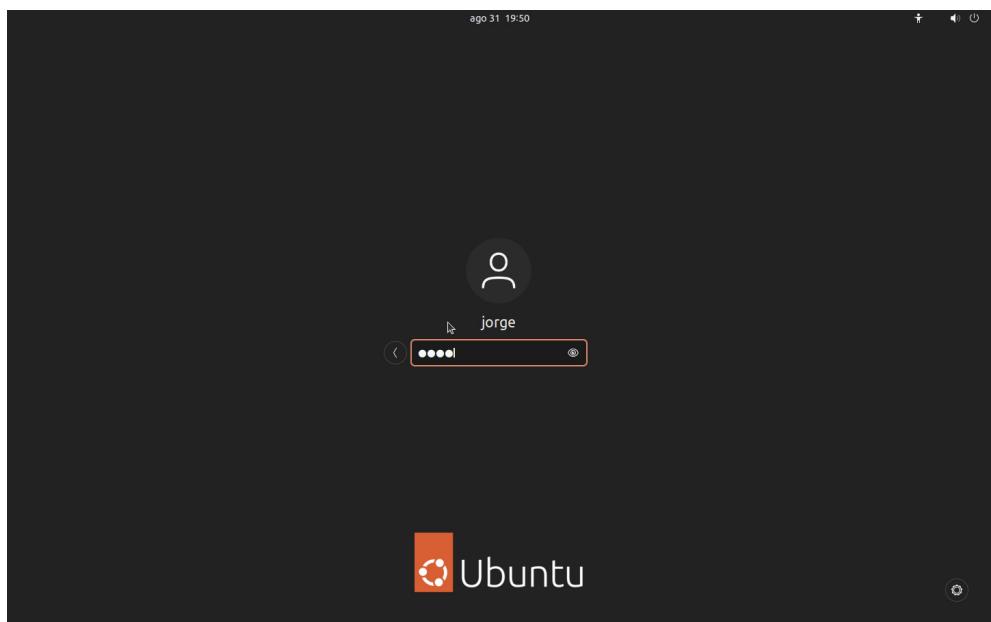
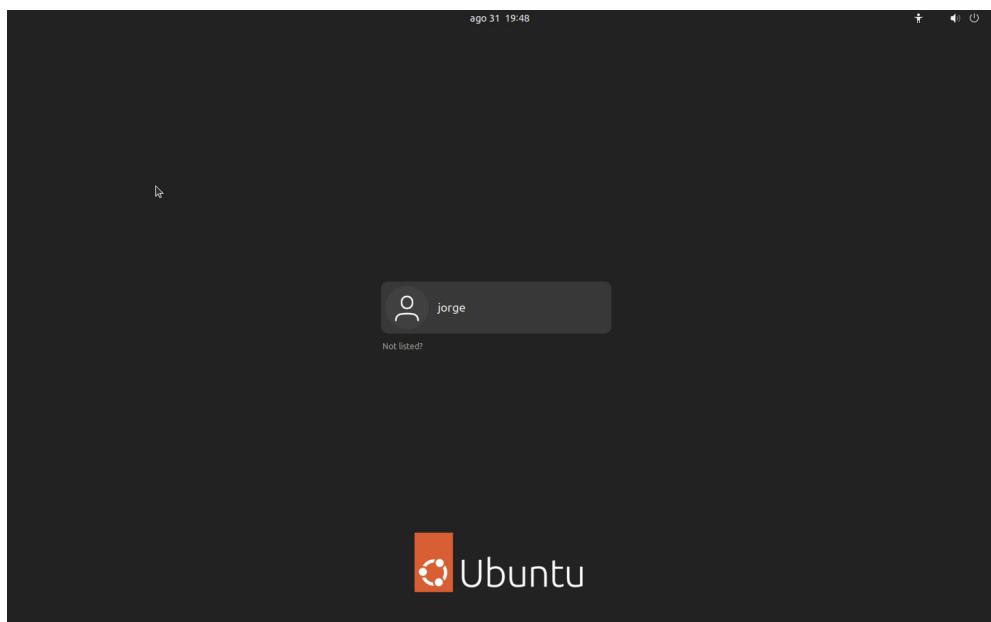


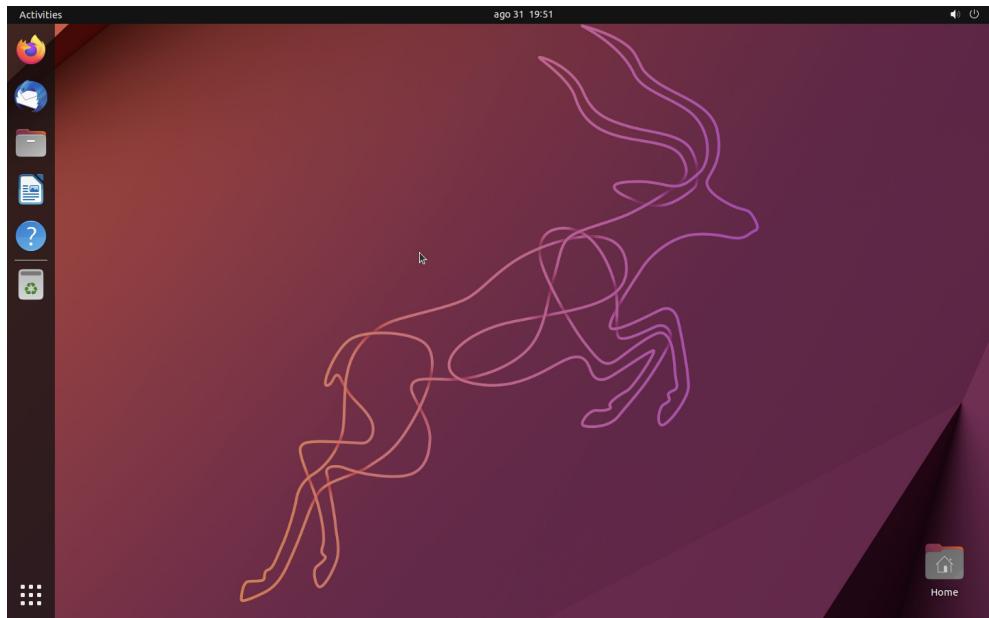
And it will return us to the terminal. Now, we will restart it from the terminal by running the following command:

```
sudo reboot now
```



Now, the graphical interface appears:

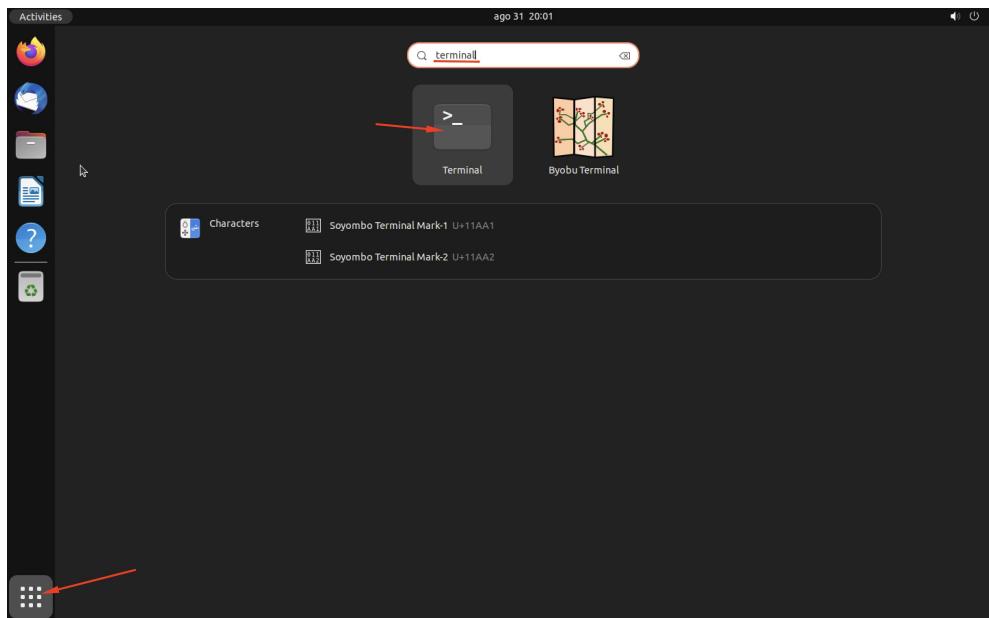




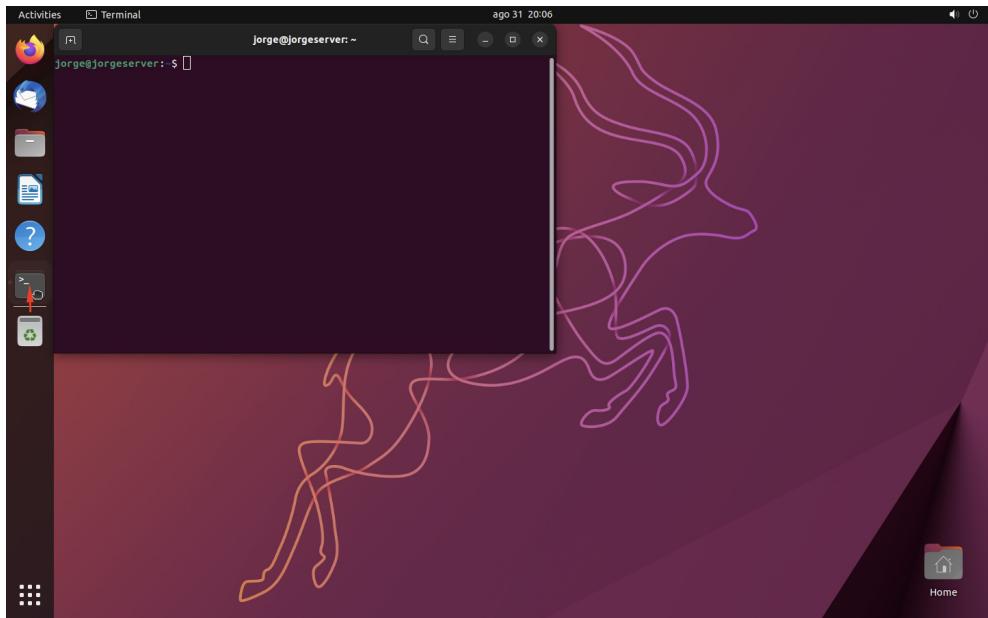
Now, there's a small issue: we can't see the **Network Manager** or the system to configure and manage networks.

To fix this, we need to open the terminal:

Select the square with dots at the bottom left, type **terminal** in the search bar, and select **Terminal**.



Now, to keep it fixed in the application bar, drag it upwards, above the separator bar.



To edit a configuration file, I recommend installing **gedit**. To do so, run the following command:

```
sudo apt install gedit
```

If it asks to restart any services at the end of the installation, don't select anything and click **OK**.

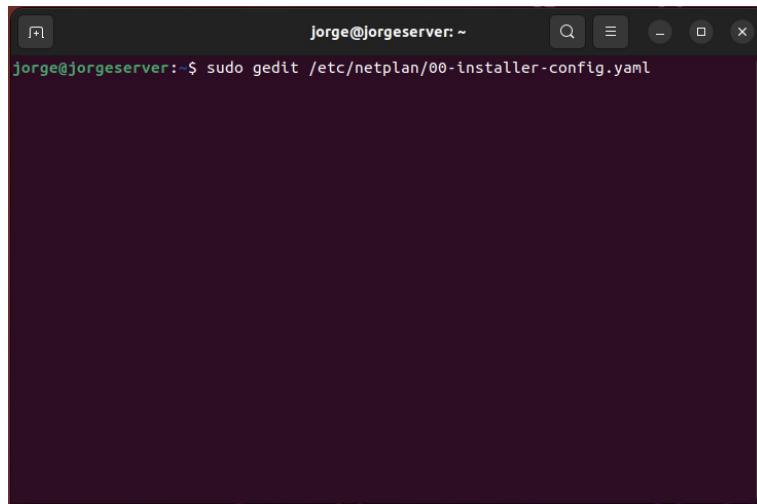
```
jorge@jorgeserver:~$ sudo apt install gedit
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Se instalarán los siguientes paquetes adicionales:
  gedit-common gir1.2-gtksource-4 libgtksourceview-4-0
  libgtksourceview-4-common python3-gi-cairo
Paquetes sugeridos:
  gedit-plugins
Se instalarán los siguientes paquetes NUEVOS:
  gedit gedit-common gir1.2-gtksource-4 libgtksourceview-4-0
  libgtksourceview-4-common python3-gi-cairo
0 actualizados, 6 nuevos se instalarán, 0 para eliminar y 0 no actualizados.
Se necesita descargar 2.853 kB de archivos.
Se utilizarán 20,7 MB de espacio de disco adicional después de esta operación.
¿Desea continuar? [S/n] s
Des:1 http://ports.ubuntu.com/ubuntu-ports kinetic/universe arm64 gedit-common all 42.2.1 [1.575 kB]
Des:2 http://ports.ubuntu.com/ubuntu-ports kinetic/universe arm64 libgtksourceview-4-common all 4.8.3-1ubuntu1 [592 kB]
Des:3 http://ports.ubuntu.com/ubuntu-ports kinetic/universe arm64 libgtksourceview-4-0 arm64 4.8.3-1ubuntu1 [228 kB]
Des:4 http://ports.ubuntu.com/ubuntu-ports kinetic/universe arm64 gir1.2-gtksource-4 arm64 4.8.3-1ubuntu1 [20,2 kB]
```

Now, we need to modify the **00-installer-config.yaml** file. To do this, run:

```
sudo gedit /etc/netplan/00-installer-config.yaml
```

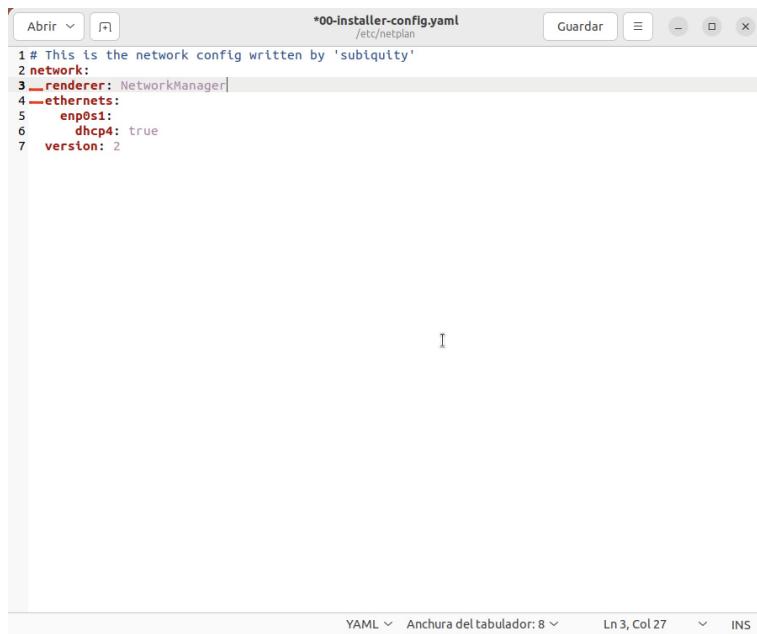
Or, if you prefer not to use **gedit**, you can use **nano** instead:

```
sudo nano /etc/netplan/00-installer-config.yaml
```



Add the line **renderer: NetworkManager** (it's very important to respect the spacing so that **renderer** is aligned with **ethernets** on the left side. In **gedit**, use two spaces with the space bar, as highlighted in red; otherwise, it will cause issues when executing it).

For more information, visit: <http://somebooks.es/establecer-una-direccion-ip-estatica-en-ubuntu-server-20-04/>



Now, click **Save**, and close the window by clicking the 'X' in the top corner.

In the terminal, run the following command to generate the configuration:

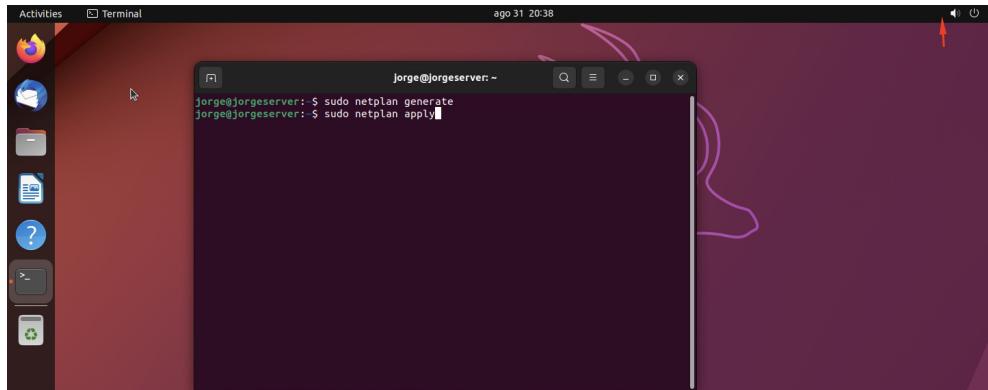
```
sudo netplan generate
```

If there's a problem, it will be indicated; otherwise, nothing will appear.

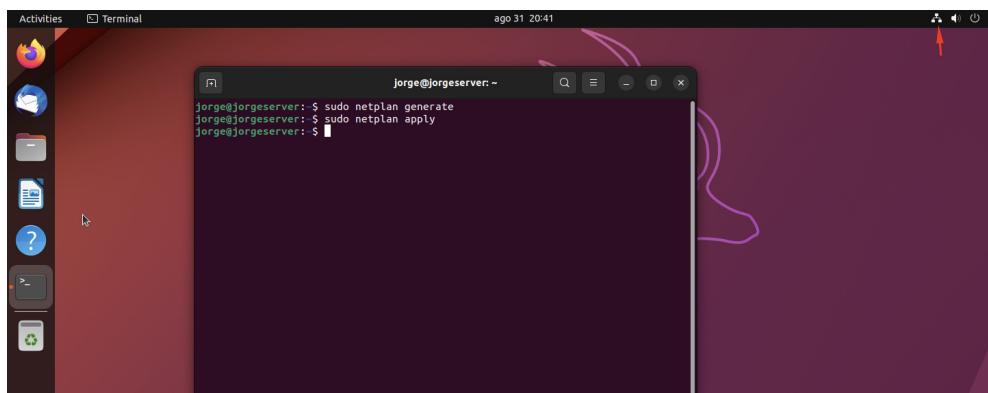
Next, run the command to apply the changes:

```
sudo netplan apply
```

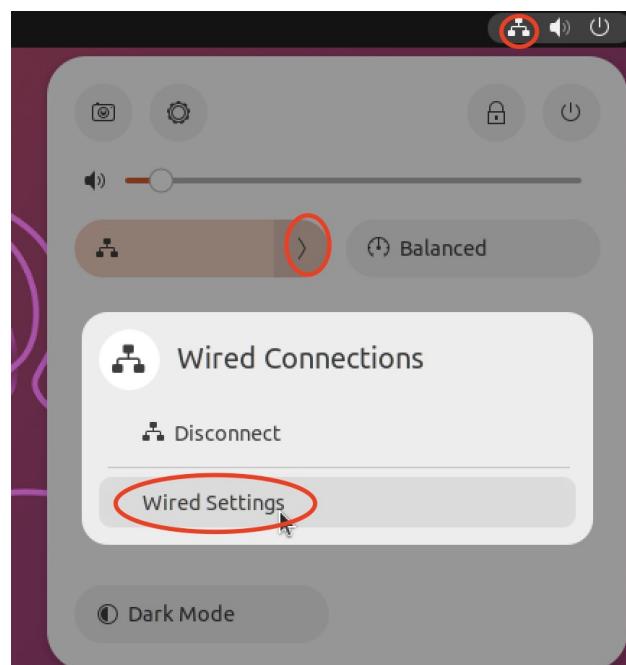
Before executing this command, you'll notice that nothing appears at the top.

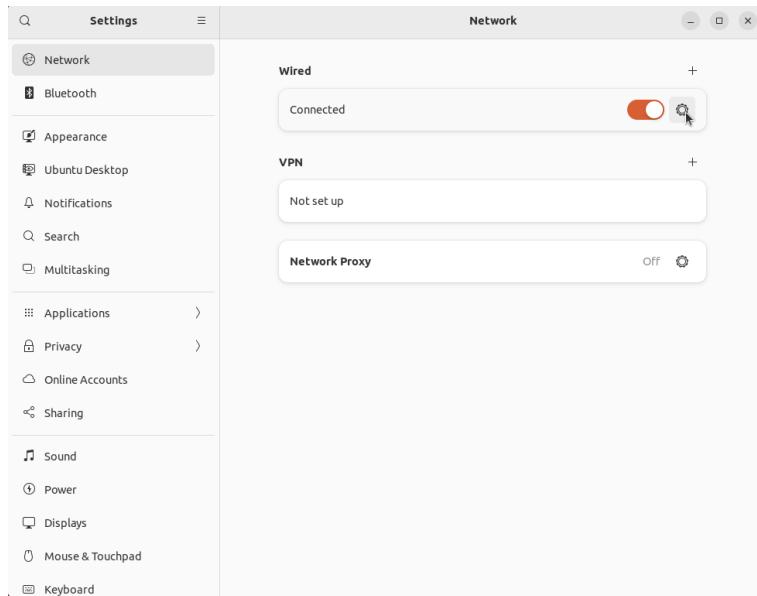


Pero después de aplicar aparece el Network Manager

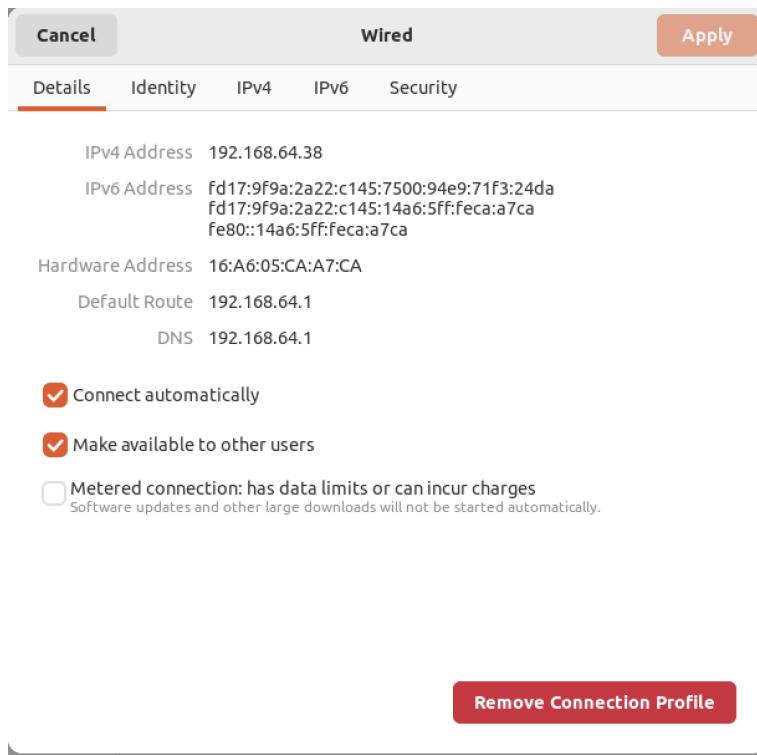


Now, we can manage the network interfaces.





And here, we can change the network card configurations.



Now, let's create a static IP from the configuration file:

1. Run the following command to edit the configuration file:

```
sudo gedit /etc/netplan/00-installer-config.yaml
```

2. Add a # before **dhcp4** to ignore it. You can copy everything else, but be careful to respect the spacing and ensure the IP address doesn't cause any connection issues (indicated by a question mark on the connection icon). The static IP should be similar to the one provided automatically. In this case, it was **192.168.64.38** → **192.168.64.x**, and for this example, I will use **192.168.64.200**.

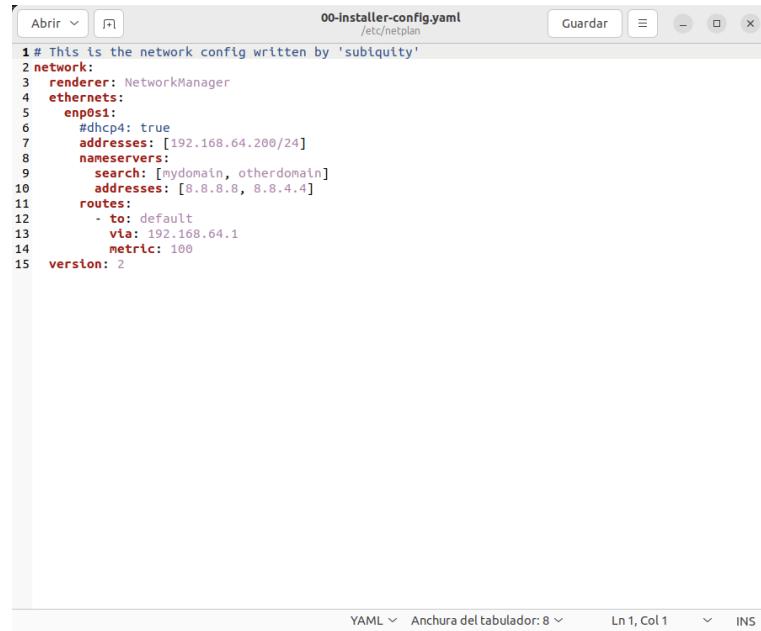
3. After editing, run:

```
sudo netplan generate
```

If there's an issue, it will notify you, otherwise, nothing will appear, and it will tell you where the error is if one occurs.

4. Finally, to apply the changes, run:

```
sudo netplan apply
```



The screenshot shows a terminal window with the file `00-installer-config.yaml` open. The content of the file is as follows:

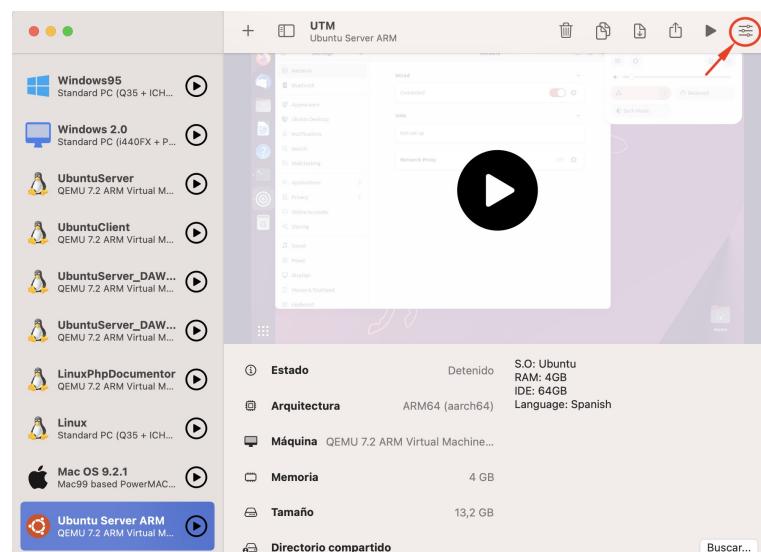
```
1 # This is the network config written by 'subiquity'
2 network:
3   renderer: NetworkManager
4   ethernets:
5     enp0s1:
6       dhcp4: true
7       addresses: [192.168.64.200/24]
8       nameservers:
9         search: [mydomain, otherdomain]
10        addresses: [8.8.8.8, 8.8.4.4]
11        routes:
12          - to: default
13            via: 192.168.64.1
14            metric: 100
15 version: 2
```

The terminal interface includes standard navigation keys like `YAML`, `Anchura del tabulador: 8`, `Ln 1, Col 1`, and `INS`.

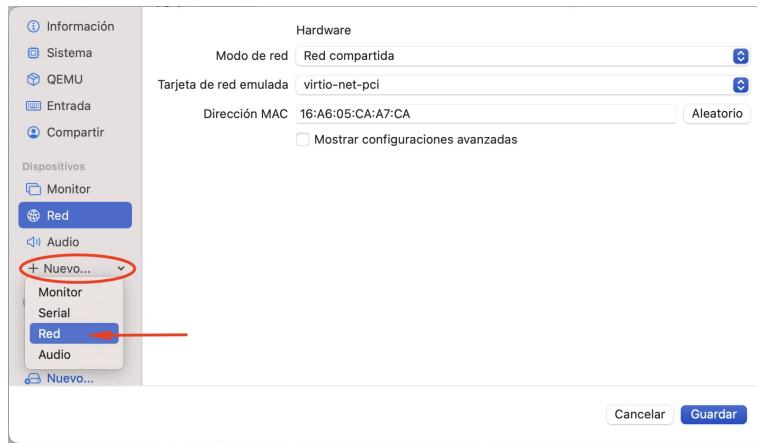
Now, let's configure 2 network interfaces:

1. Shut down Ubuntu.

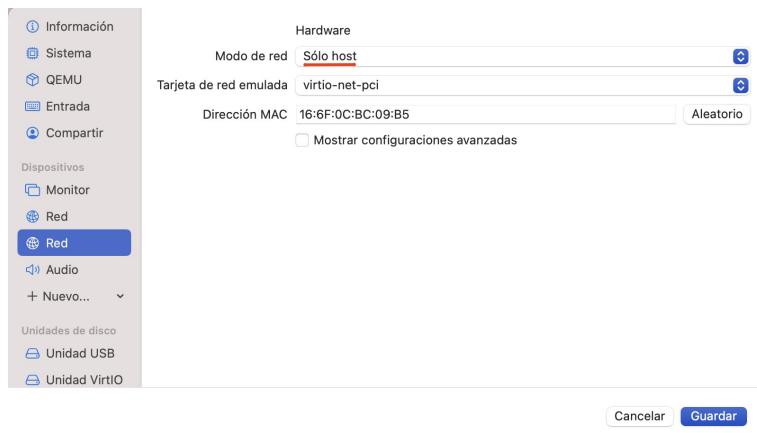
2. In UTM, go to Ubuntu Server, and then select Configuration.



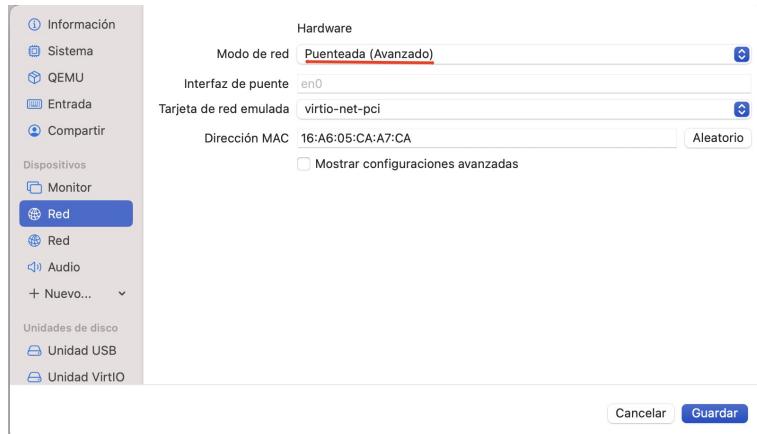
Add a new network interface:



Set the second network interface to **Host-only** (for example).

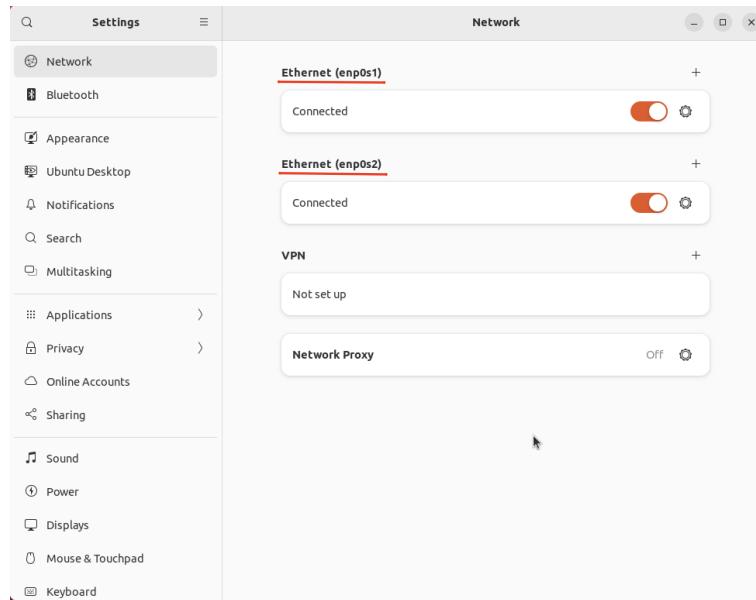


And we change the first network interface to **bridged** (for example).



Select **Save** and start the virtual machine.

We can now see that both network interfaces appear.



Let's configure them from the configuration file:

```
sudo gedit /etc/netplan/00-installer-config.yaml
```

```

1 # This is the network config written by 'subiquity'
2 network:
3   renderer: NetworkManager
4   ethernets:
5     enp0s1:
6       dhcp4: true
7       addresses: [192.168.64.200/24]
8       nameservers:
9         search: [mydomain, otherdomain]
10        addresses: [8.8.8.8, 8.8.4.4]
11        routes:
12          - to: default
13            via: 192.168.64.1
14            metric: 100
15     enp0s2:
16       addresses: [10.10.10.2/24]
17       nameservers:
18         addresses: [10.10.10.1, 1.1.1.1]
19       routes:
20         - to: default
21           via: 10.10.10.1
22           metric: 50
23 version: 2

```

```
sudo netplan generate
```

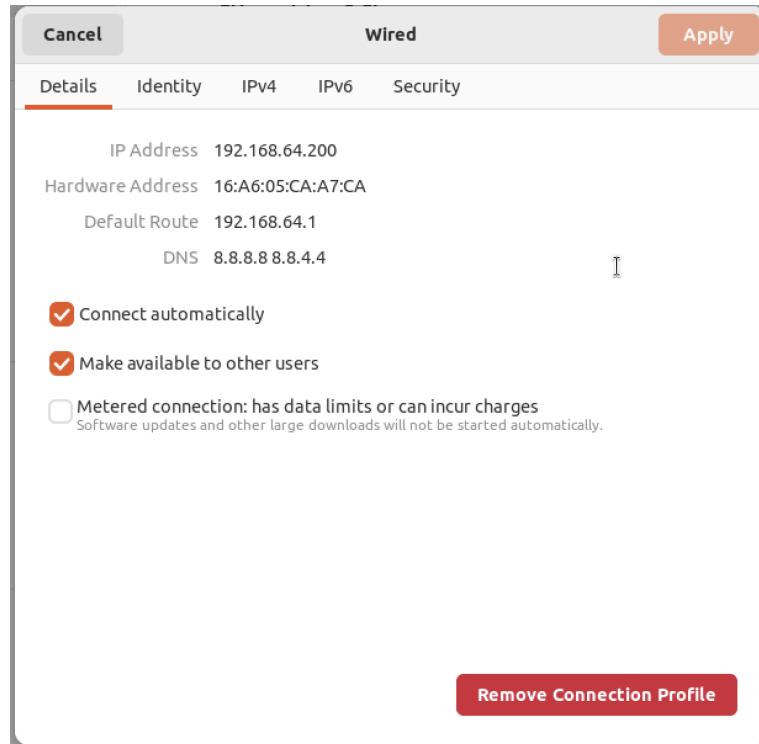
If there's an issue, it will notify you; otherwise, nothing will appear, and it will tell you where the error is if one occurs.

Then, to apply the changes, run:

```
sudo netplan apply
```

Now, we can see how the configurations have been applied to both network interfaces.

Enp0s1:



enp0s2:

