

Samba Active Directory en Ubuntu Server

ÍNDICE

- Configuración máquinas
- Servidor
 - Configuración Inicial
 - Instalación Samba AD
 - Configuración NTP
 - Configuración NFTABLES
 - Crear usuarios samba
- Cliente
 - Instalación aplicativos
 - Unión al dominio

Esquema red

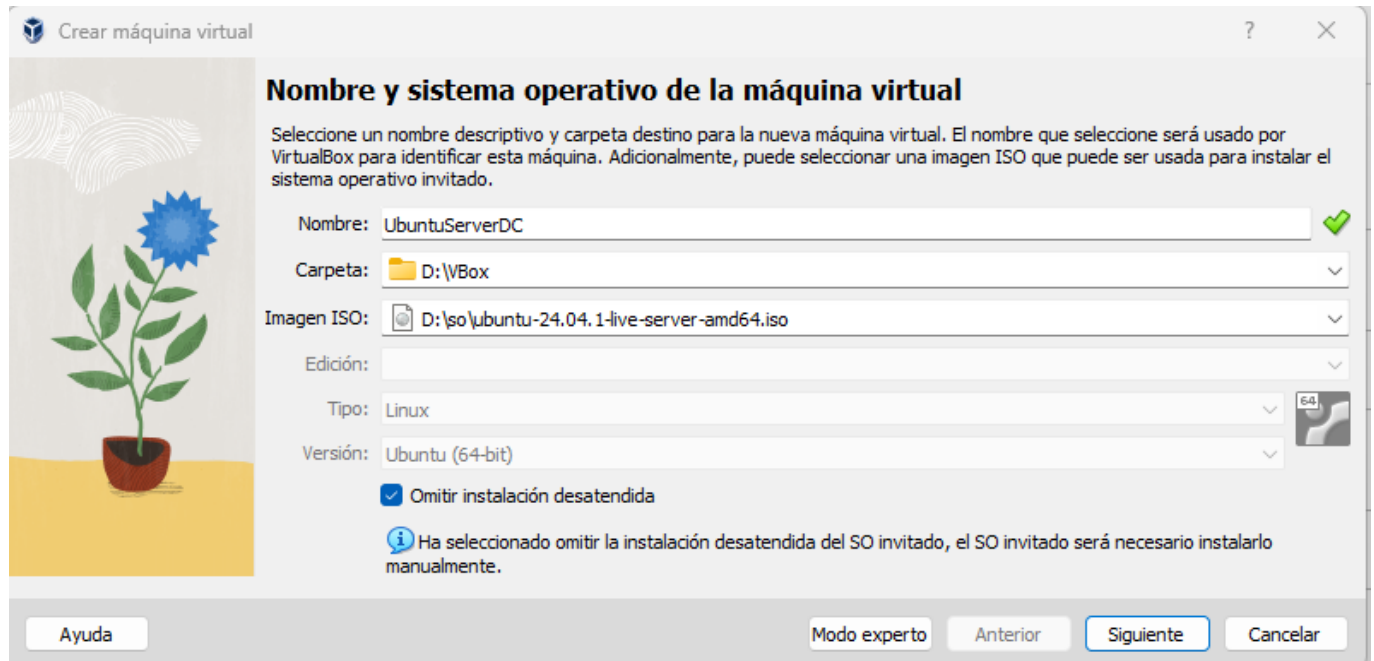
Máquina	SO	Usuario	Contraseña	IP
UbuntuServerDC	Ubuntu Server	lcm3USDC	abc123.	192.168.11.10
lcmUbuntuClient1	Ubuntu 22	lcm3UC1	abc123.	192.168.11.21

Dominio
dc=lcmASO,dc=local
FQDN: lcmASO.local

Configuración máquinas

Ubuntu Server

Creamos una VM nueva Ubuntu Server 24.01 en VirtualBox.



Crear máquina virtual

Nombre y sistema operativo de la máquina virtual

Seleccione un nombre descriptivo y carpeta destino para la nueva máquina virtual. El nombre que seleccione será usado por VirtualBox para identificar esta máquina. Adicionalmente, puede seleccionar una imagen ISO que puede ser usada para instalar el sistema operativo invitado.

Nombre: ✓

Carpeta:


Imagen ISO:

Edición:


Tipo:

Versión:

☒ Omitir instalación desatendida

 Ha seleccionado omitir la instalación desatendida del SO invitado, el SO invitado será necesario instalarlo manualmente.

[Ayuda](#) [Modo experto](#) [Anterior](#) [Siguiente](#) [Cancelar](#)



Crear máquina virtual

?

✕

Hardware

Puede modificar el hardware de la máquina virtual cambiando la cantidad de RAM y número de CPU virtuales. También es posible habilitar EFI.

Memoria base:

4 MB

16384 MB

4096 MB

Procesadores:

1 CPU

12 CPUs

2


☐ Habilitar EFI (sólo SO especiales)

Ayuda

Anterior

Siguiente

Cancelar



Crear máquina virtual

?

✕

Disco duro virtual

Si lo desea puede añadir un nuevo disco duro virtual a la nueva máquina. Puede crear un nuevo archivo de disco duro o seleccionar uno existente. De forma alternativa puede crear una máquina virtual sin un disco duro virtual.

☒ Crear un disco duro virtual ahora

Tamaño de disco:

4,00 MB

2,00 TB

25,00 GB

☐ Reservar tamaño completo

☐ Usar un archivo de disco duro virtual existente

ExamenSAD_debian.vdi (Normal, 50,00 GB)

☐ No añadir un disco duro virtual

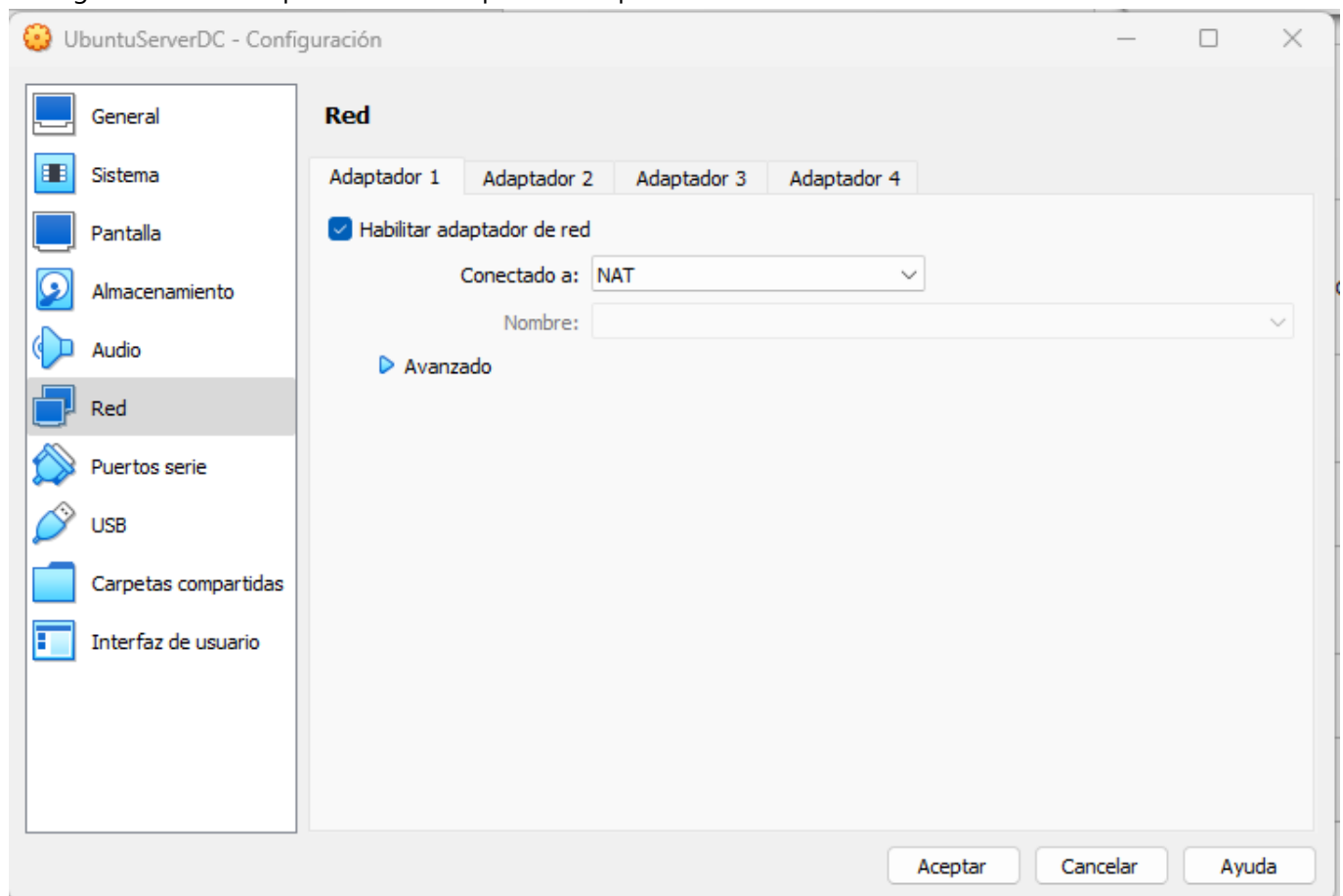
Ayuda

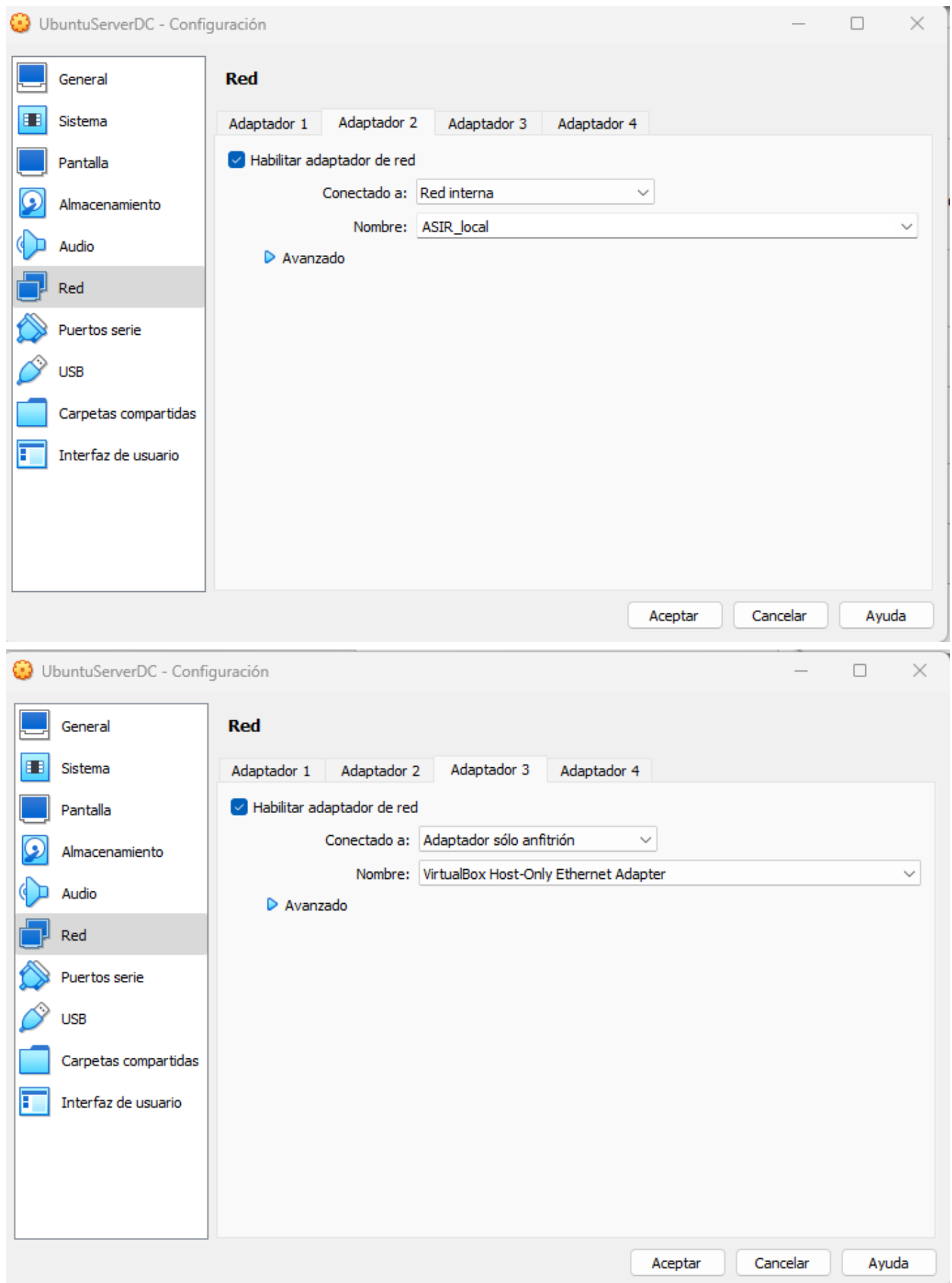
Anterior

Siguiente

Cancelar

Configuramos tres adaptadores de red para la máquina servidor.

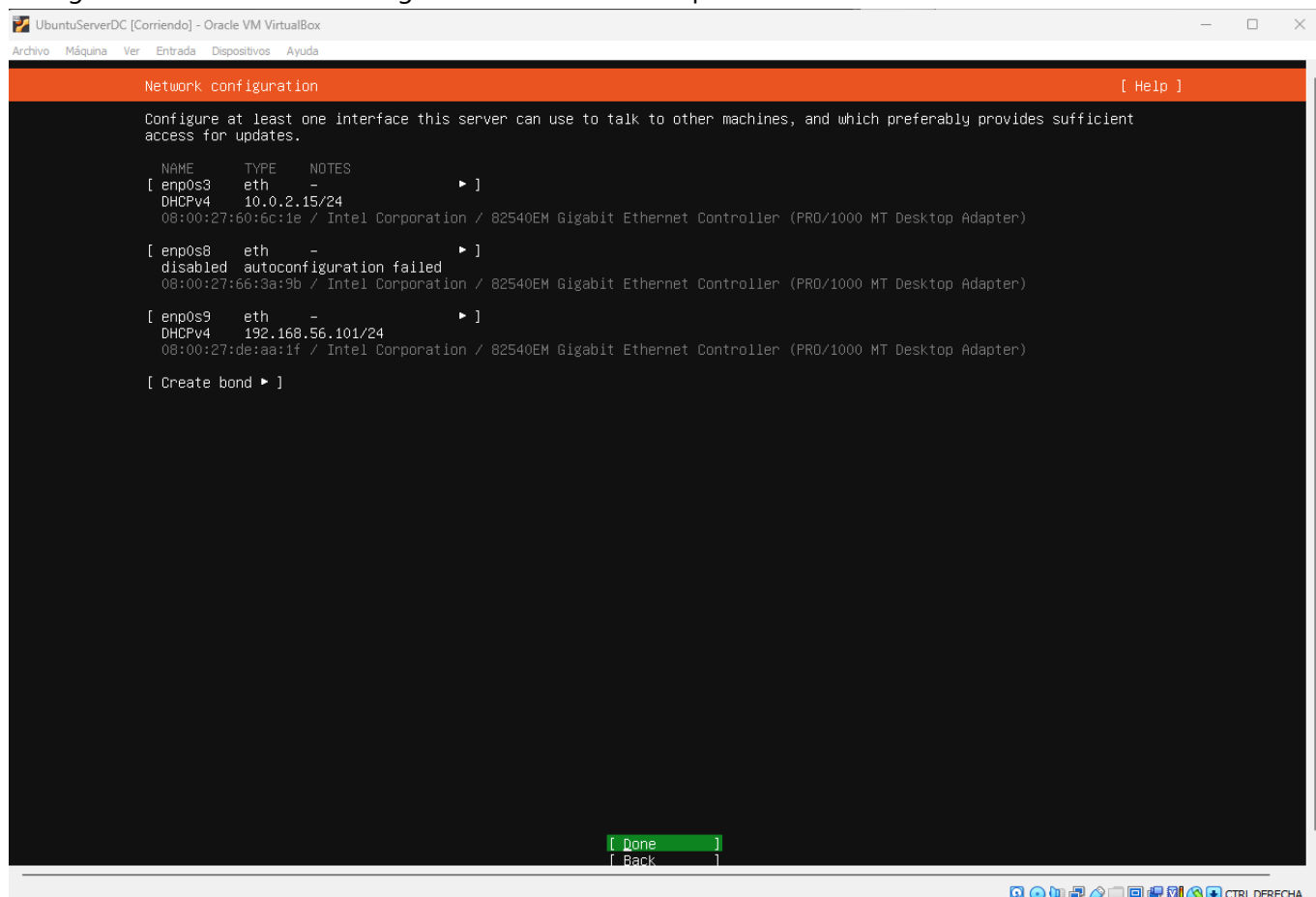




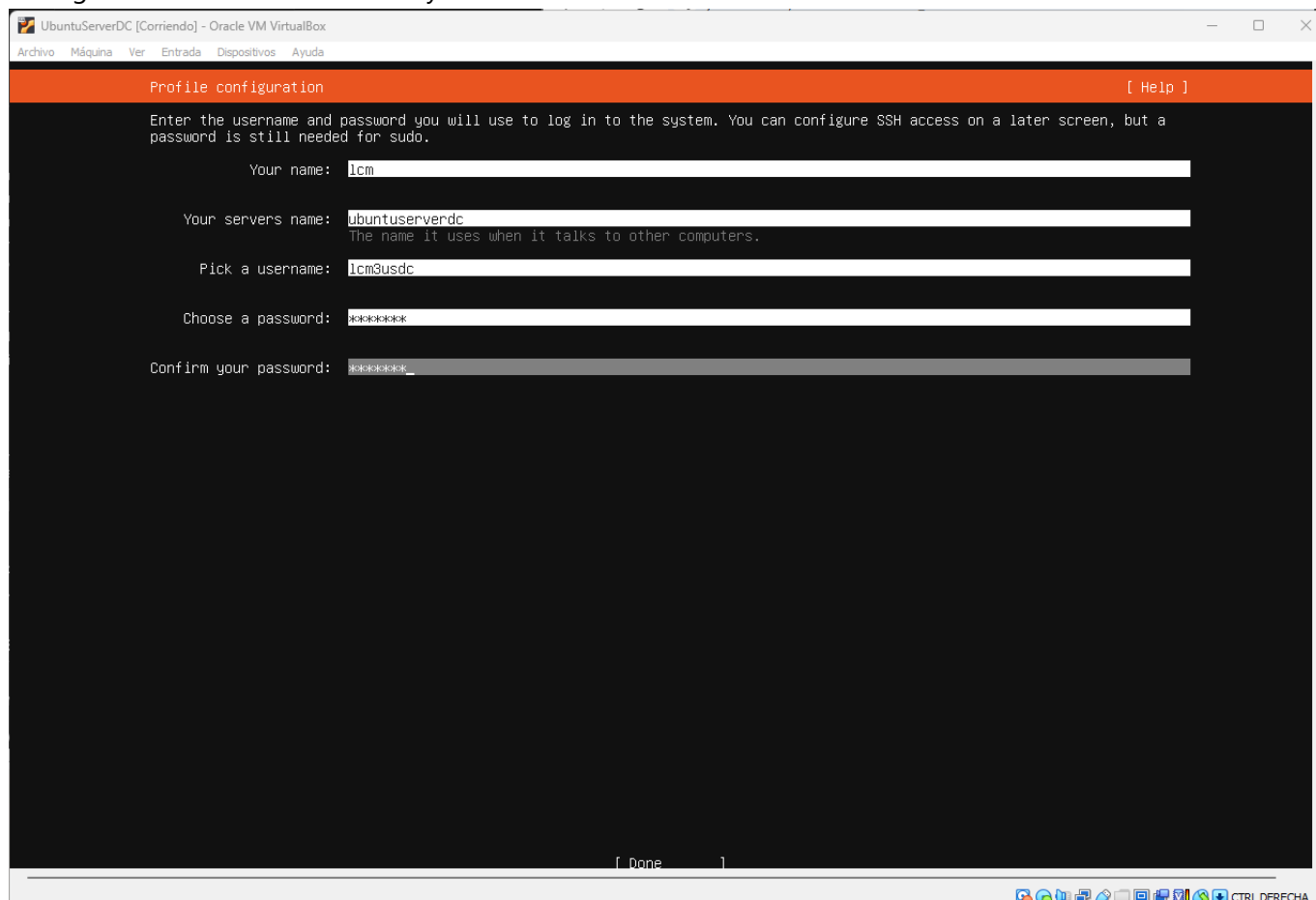
Arrancamos la máquina e instalamos el so, durante la instalación se nos va a solicitar, además del idioma de sistema y teclado, configurar la red.

Dejamos la configuración por defecto para los tres adaptadores y continuamos con la instalación ya que la

configuración de red la vamos a gestionar a través de netplan más adelante.

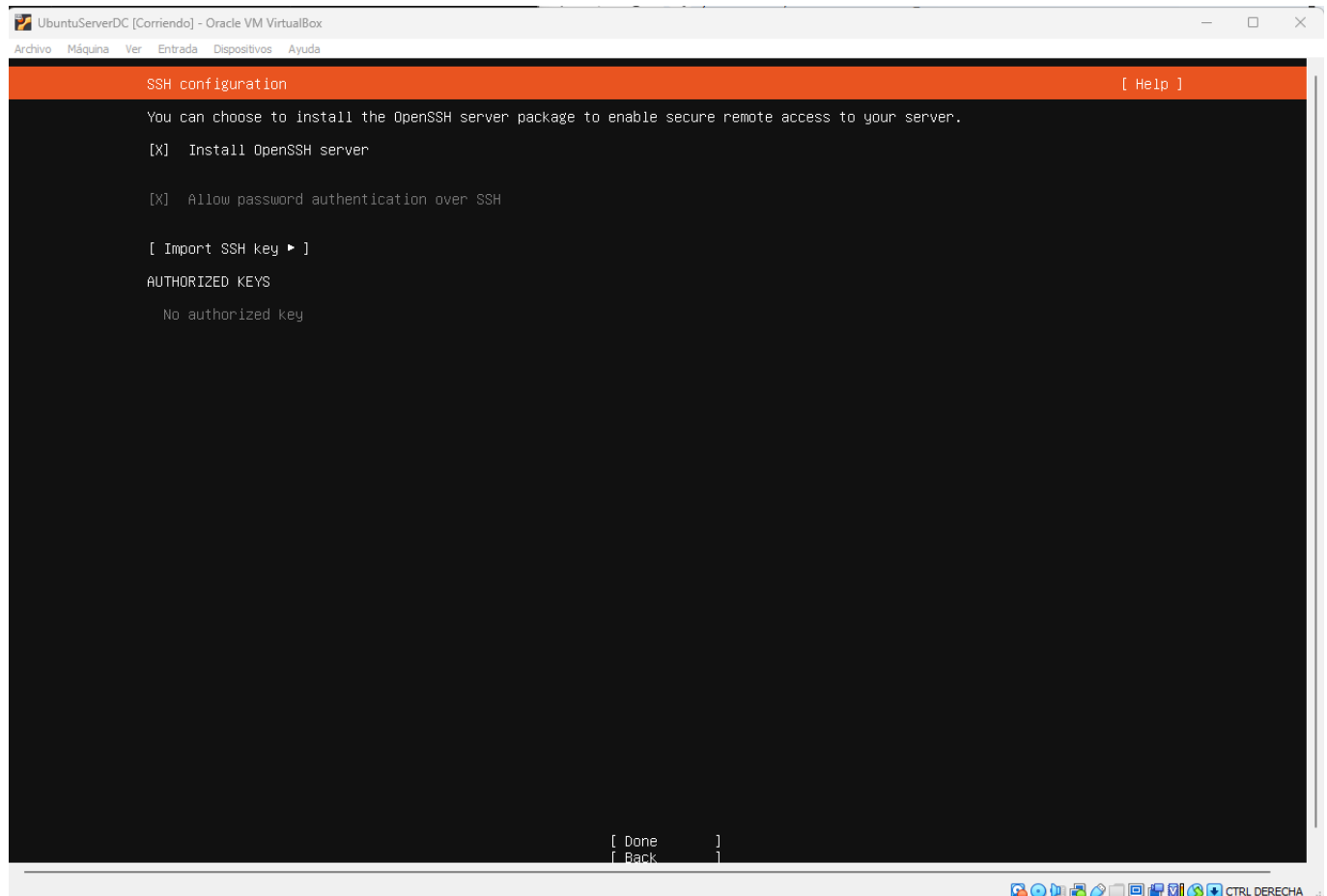


Configuramos nombre de servidor y usuario.

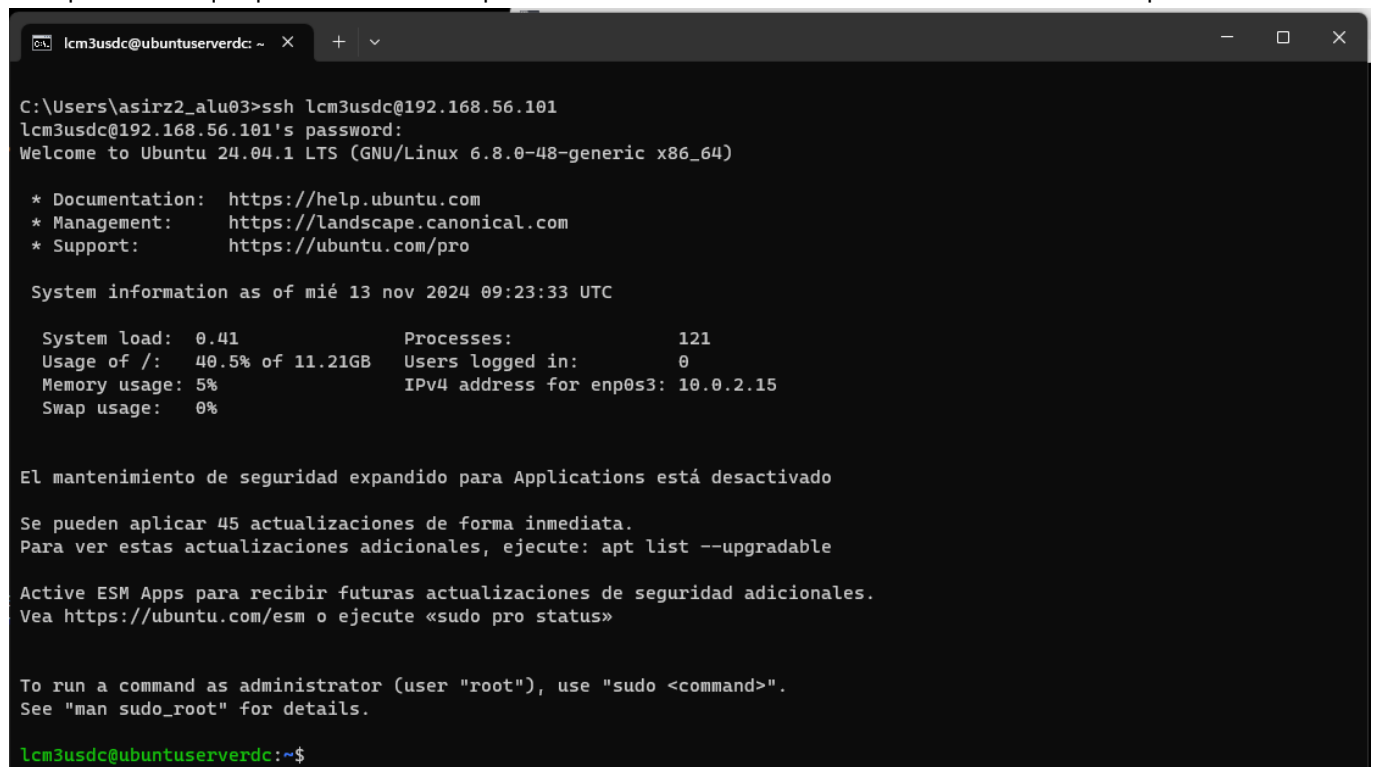


Cuando nos lo solicite instalamos OpenSSH para poder conectarnos de forma remota al server a través de

SSH.




Comprobamos que podemos acceder por SSH al servidor desde nuestro host a través de la ip 192.168.56.101.



Ubuntu Client


Creamos una VM nueva Ubuntu Cliente 22.04 en VirtualBox.



Crear máquina virtual

Nombre y sistema operativo de la máquina virtual


Seleccione un nombre descriptivo y carpeta destino para la nueva máquina virtual. El nombre que seleccione será usado por VirtualBox para identificar esta máquina. Adicionalmente, puede seleccionar una imagen ISO que puede ser usada para instalar el sistema operativo invitado.

Nombre: 

Carpeta:


Imagen ISO:


Edición:

Tipo: 

Versión:

☒ Omitir instalación desatendida

 Ha seleccionado omitir la instalación desatendida del SO invitado, el SO invitado será necesario instalarlo manualmente.



Crear máquina virtual

Hardware


Puede modificar el hardware de la máquina virtual cambiando la cantidad de RAM y número de CPU virtuales. También es posible habilitar EFI.

Memoria base: 4096 MB
4 MB 16384 MB

Procesadores: 2
1 CPU 12 CPUs

☐ Habilitar EFI (sólo SO especiales)

[Ayuda](#) [Anterior](#) [Siguiente](#) [Cancelar](#)



Crear máquina virtual

Disco duro virtual

Si lo desea puede añadir un nuevo disco duro virtual a la nueva máquina. Puede crear un nuevo archivo de disco duro o seleccionar uno existente. De forma alternativa puede crear una máquina virtual sin un disco duro virtual.

☒ Crear un disco duro virtual ahora

Tamaño de disco: 25,00 GB
4,00 MB 2,00 TB

☐ Reservar tamaño completo

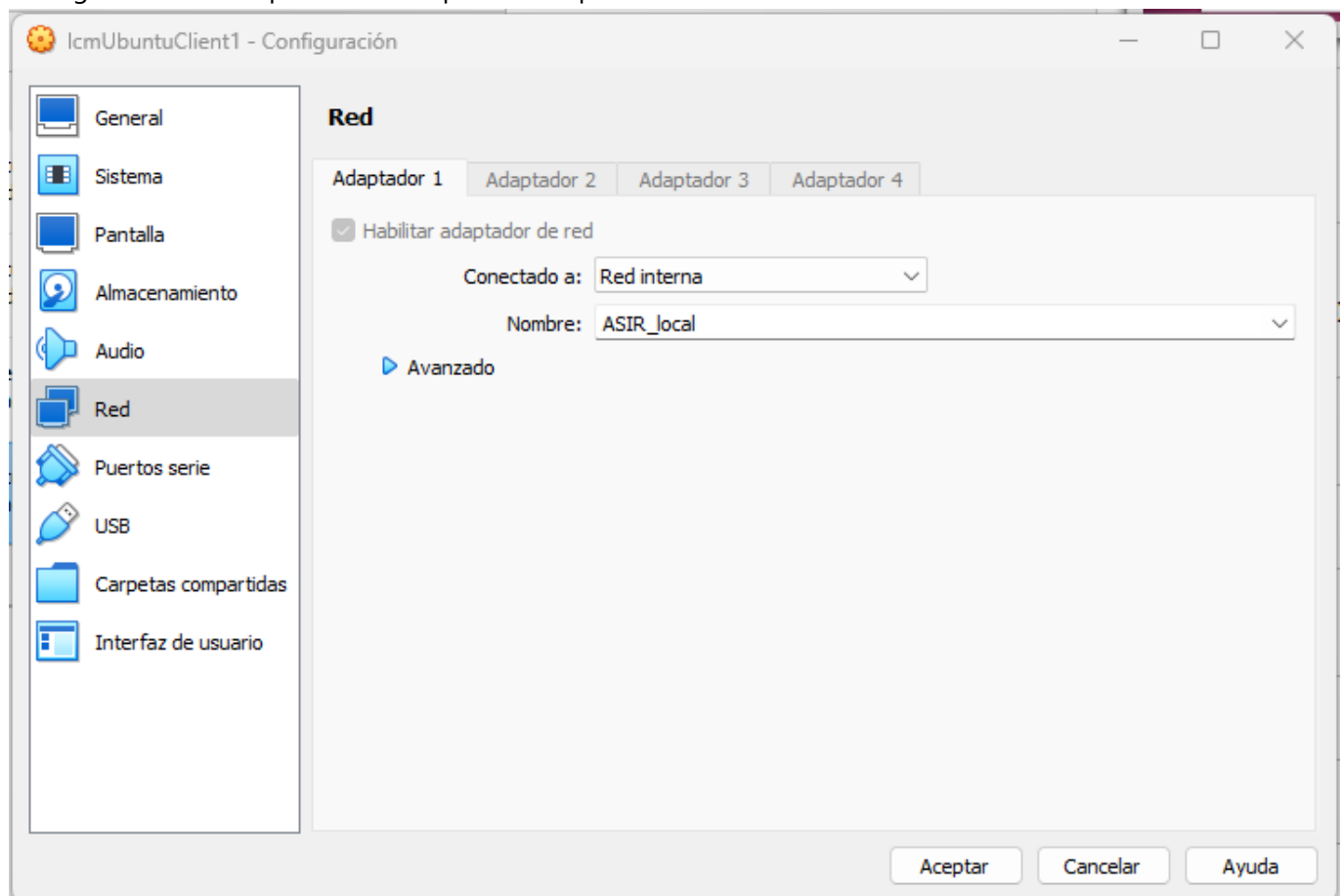
☐ Usar un archivo de disco duro virtual existente

ExamenSAD_debian.vdi (Normal, 50,00 GB)

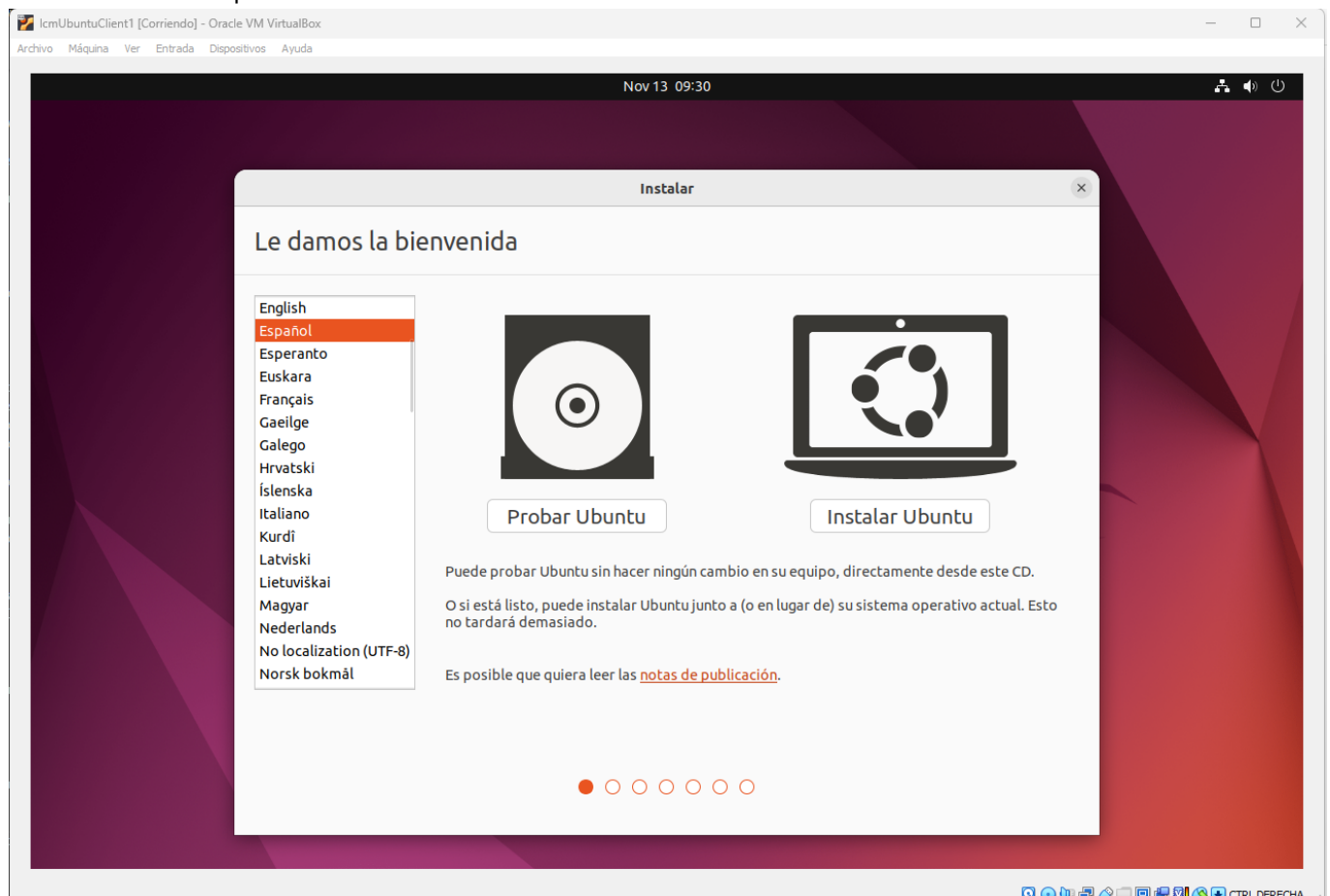
☐ No añadir un disco duro virtual

[Ayuda](#) [Anterior](#) [Siguiente](#) [Cancelar](#)

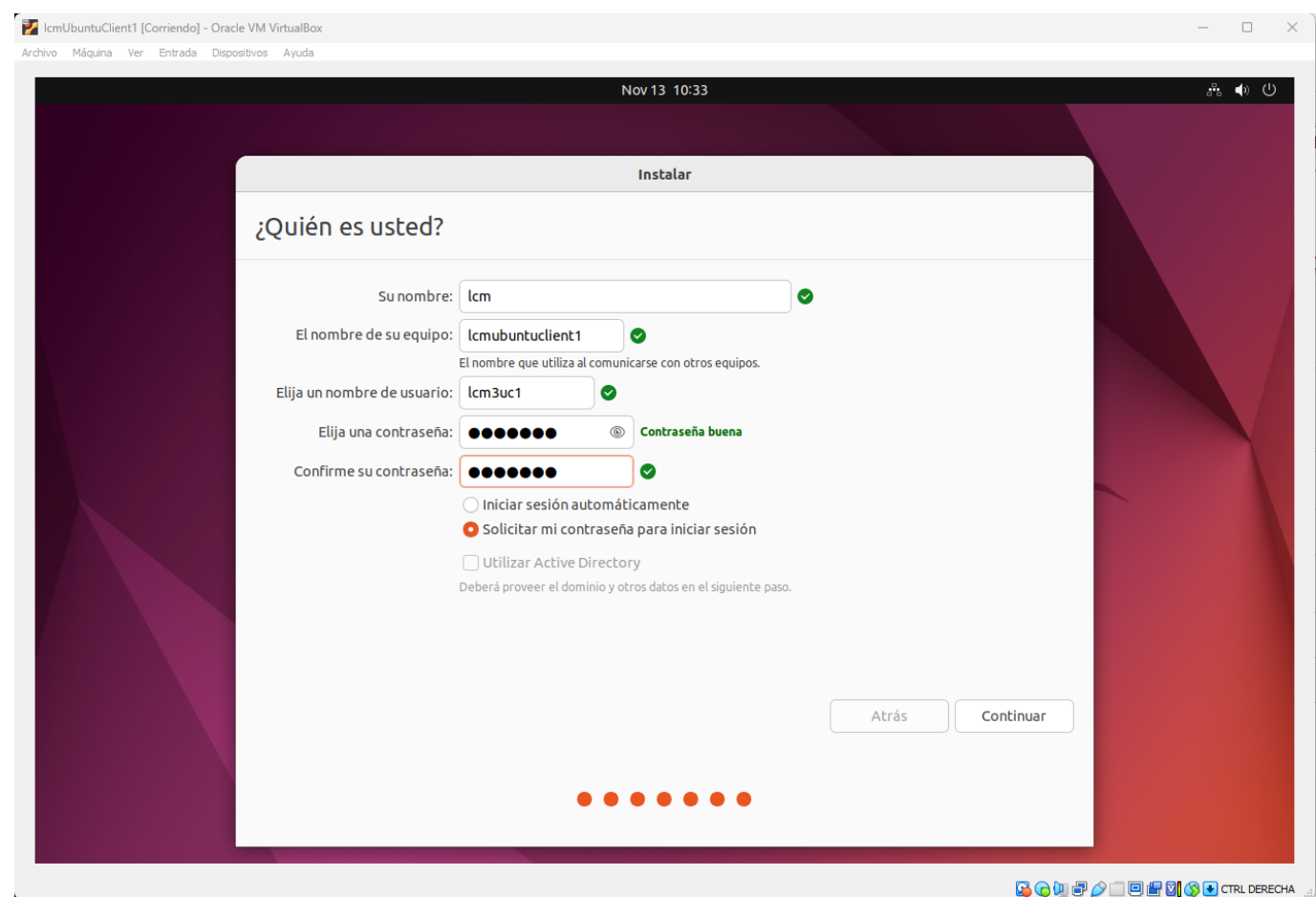
Configuramos un adaptador de red para la máquina en modo red interna.



Arrancamos la máquina e instalamos el so.



Configuramos nombre de equipo y usuario.

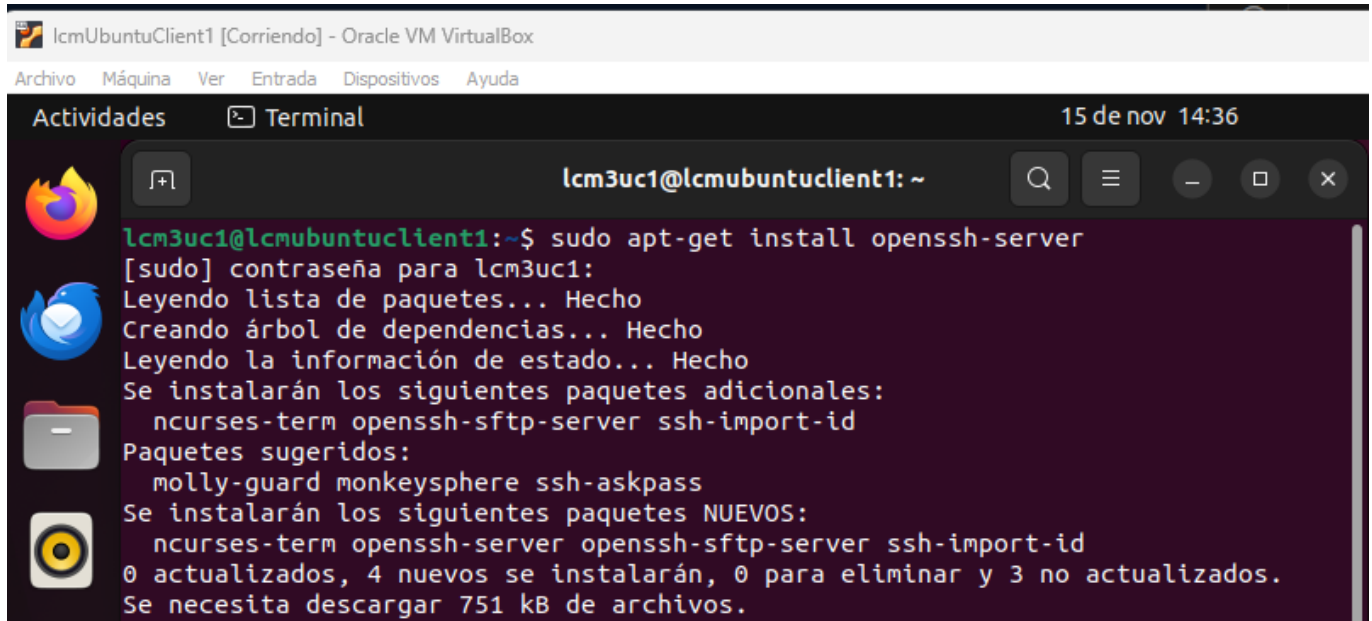


Configuramos IP fija para cliente y añadimos la IP del servidor como servidor DNS y como puerta de enlace.



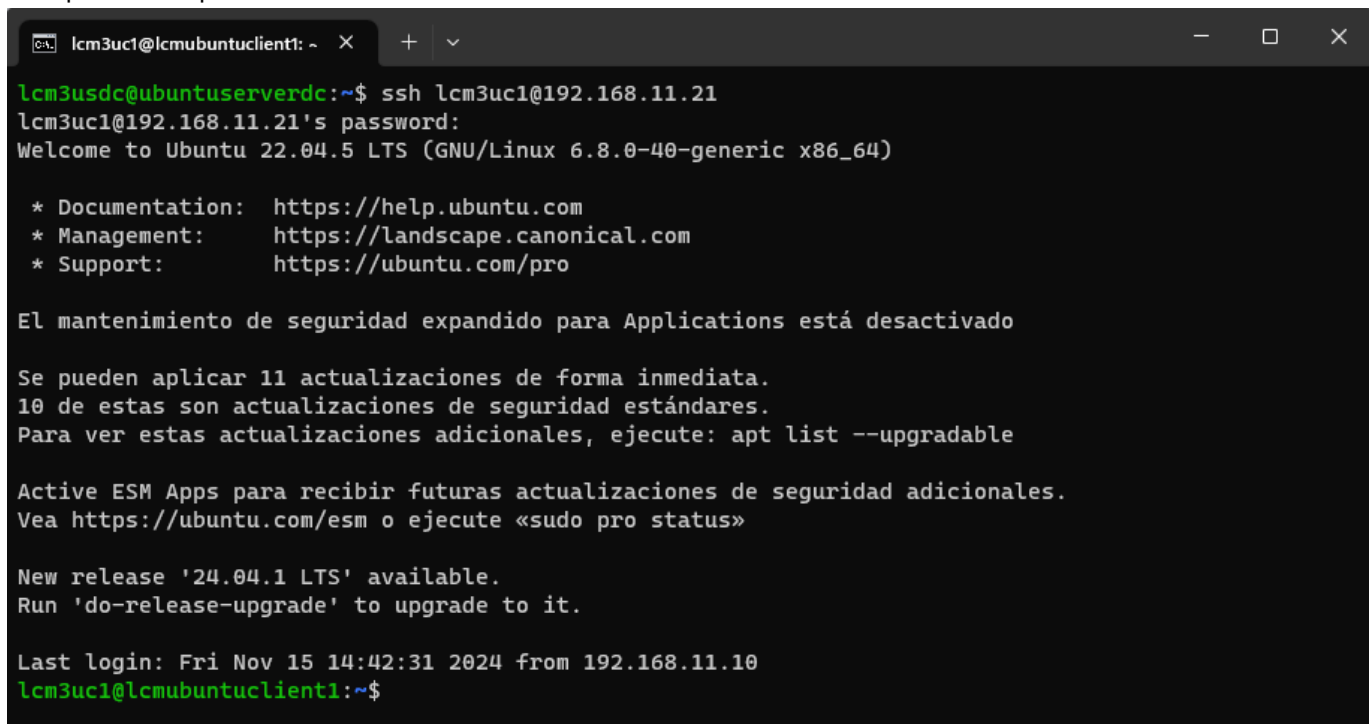
Los siguientes pasos los realizamos después de haber realizado el apartado *Configuración nftables*.

Instalamos OpenSSH para poder conectarnos en remoto por ssh.



```
lcm3uc1@lcmubuntuclient1:~$ sudo apt-get install openssh-server
[sudo] contraseña para lcm3uc1:
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes adicionales:
  ncurses-term openssh-sftp-server ssh-import-id
Paquetes sugeridos:
  molly-guard monkeysphere ssh-askpass
Se instalarán los siguientes paquetes NUEVOS:
  ncurses-term openssh-server openssh-sftp-server ssh-import-id
0 actualizados, 4 nuevos se instalarán, 0 para eliminar y 3 no actualizados.
Se necesita descargar 751 kB de archivos.
```

Comprobamos podemos conectarnos via SSH desde el servidor.



```
lcm3usdc@ubuntuserverdc:~$ ssh lcm3uc1@192.168.11.21
lcm3uc1@192.168.11.21's password:
Welcome to Ubuntu 22.04.5 LTS (GNU/Linux 6.8.0-40-generic x86_64)

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

El mantenimiento de seguridad expandido para Applications está desactivado

Se pueden aplicar 11 actualizaciones de forma inmediata.
10 de estas son actualizaciones de seguridad estándares.
Para ver estas actualizaciones adicionales, ejecute: apt list --upgradable

Active ESM Apps para recibir futuras actualizaciones de seguridad adicionales.
Vea https://ubuntu.com/esm o ejecute «sudo pro status»

New release '24.04.1 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

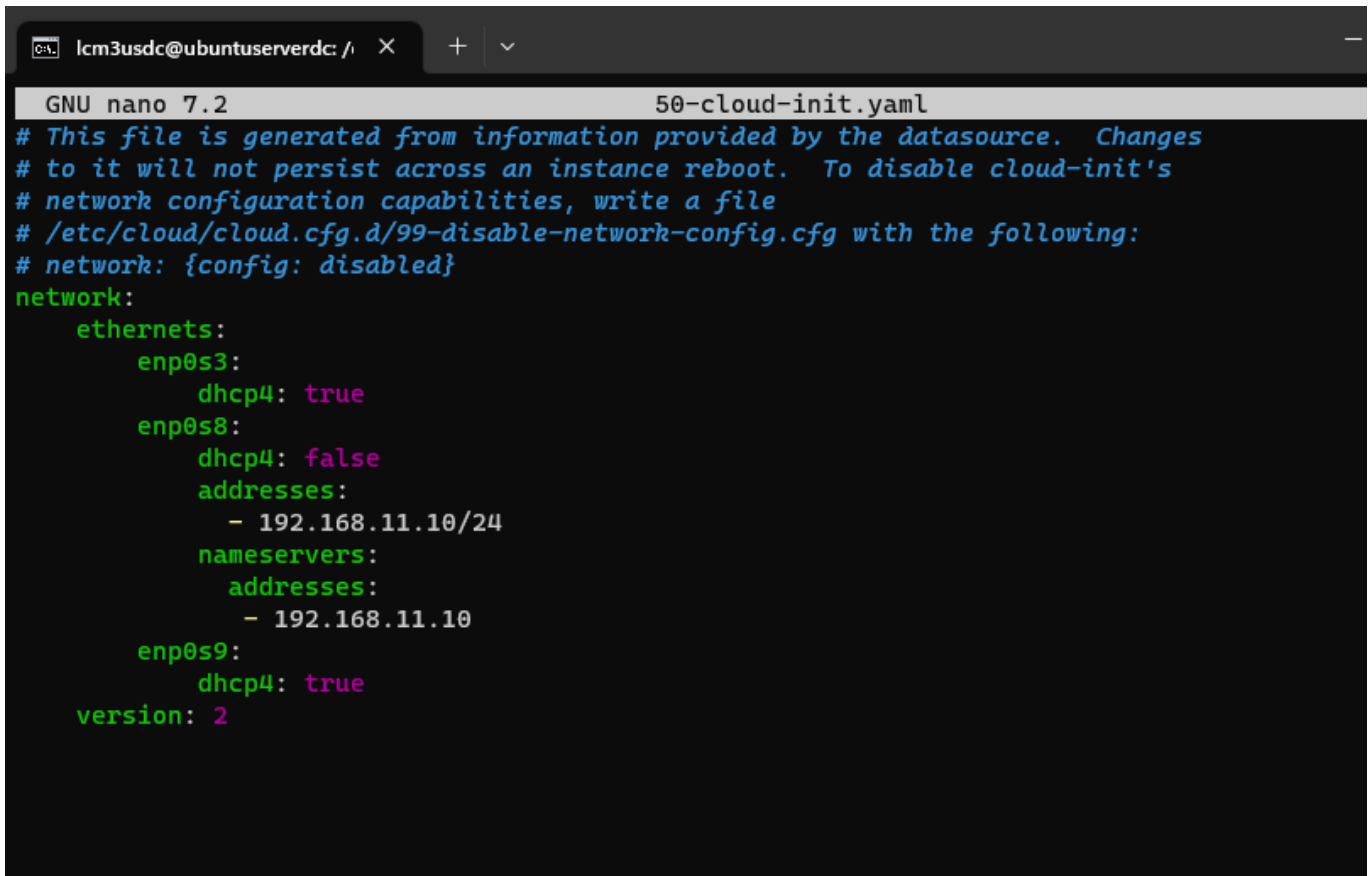
Last login: Fri Nov 15 14:42:31 2024 from 192.168.11.10
lcm3uc1@lcmubuntuclient1:~$
```

Servidor

Conectamos con el servidor desde nuestro equipo host con ssh.

Vamos a dar a nuestro servidor configuración de red para que el adaptador en red interna funcione con IP fija

192.168.11.10.



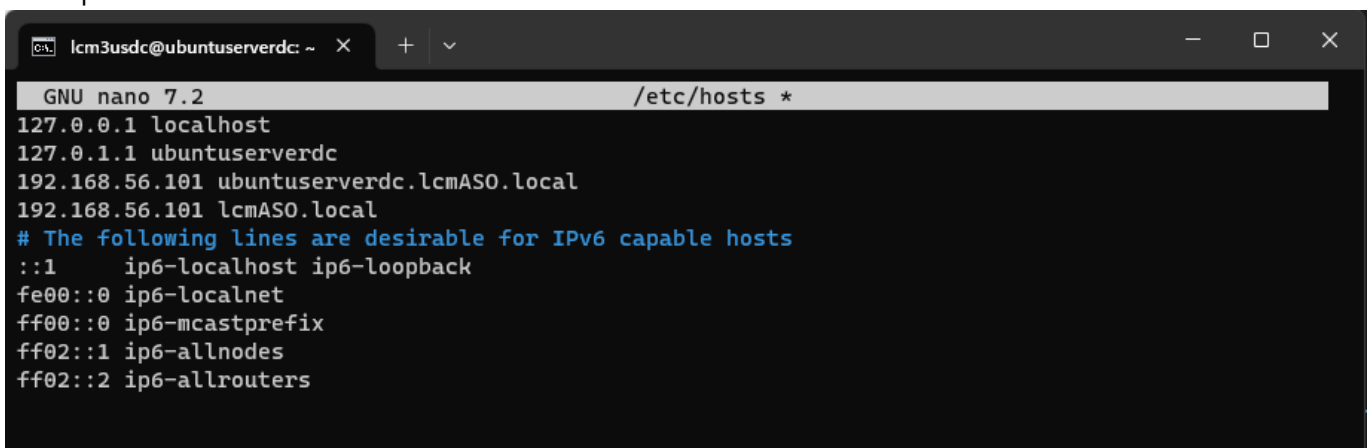
```
lcm3usdc@ubuntuserverdc: /  X + v
GNU nano 7.2                    50-cloud-init.yaml
# This file is generated from information provided by the datasource.  Changes
# to it will not persist across an instance reboot.  To disable cloud-init's
# network configuration capabilities, write a file
# /etc/cloud/cloud.cfg.d/99-disable-network-config.cfg with the following:
# network: {config: disabled}
network:
  ethernets:
    enp0s3:
      dhcp4: true
    enp0s8:
      dhcp4: false
      addresses:
        - 192.168.11.10/24
      nameservers:
        addresses:
          - 192.168.11.10
    enp0s9:
      dhcp4: true
  version: 2
```

Una vez dentro del servidor con la configuración de red correcta comprobamos el hostname.



```
lcm3usdc@ubuntuserverdc: ~ X + v
lcm3usdc@ubuntuserverdc:~$ hostname
ubuntuserverdc
lcm3usdc@ubuntuserverdc:~$
```

Ahora vamos a modificar el archivo `/etc/hosts` para indicarle al equipo tiene que atender las peticiones correspondientes al nombre de dominio **lcmubuntuserverdc.lcmASO.local**.



```
lcm3usdc@ubuntuserverdc: ~ X + v
GNU nano 7.2                    /etc/hosts *
127.0.0.1 localhost
127.0.1.1 ubuntuserverdc
192.168.56.101 ubuntuserverdc.lcmASO.local
192.168.56.101 lcmASO.local
# The following lines are desirable for IPv6 capable hosts
::1 ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
```

Comprobamos como ahora nuestra maquina resuelve el nombre **ubuntuserverdc**.

```
lcm3usdc@ubuntuserverdc: ~  
lcm3usdc@ubuntuserverdc:~$ ping ubuntuserverdc  
PING ubuntuserverdc (127.0.1.1) 56(84) bytes of data.  
64 bytes from ubuntuserverdc (127.0.1.1): icmp_seq=1 ttl=64 time=0.013 ms  
64 bytes from ubuntuserverdc (127.0.1.1): icmp_seq=2 ttl=64 time=0.045 ms  
64 bytes from ubuntuserverdc (127.0.1.1): icmp_seq=3 ttl=64 time=0.052 ms  
64 bytes from ubuntuserverdc (127.0.1.1): icmp_seq=4 ttl=64 time=0.052 ms  
^C  
--- ubuntuserverdc ping statistics ---  
4 packets transmitted, 4 received, 0% packet loss, time 3112ms  
rtt min/avg/max/mdev = 0.013/0.040/0.052/0.016 ms  
lcm3usdc@ubuntuserverdc:~$
```

Ahora vamos a deshabilitar el servicio system-resolved ya que es incompatible con samba.

```
lcm3usdc@ubuntuserverdc:~$ sudo systemctl disable --now systemd-resolved  
Removed "/etc/systemd/system/dbus-org.freedesktop.resolve1.service".  
Removed "/etc/systemd/system/sysinit.target.wants/systemd-resolved.service".  
lcm3usdc@ubuntuserverdc:~$
```

También es necesario eliminar el enlace simbólico del archivo `/etc/resolv.conf`.

```
lcm3usdc@ubuntuserverdc:~$ sudo unlink /etc/resolv.conf  
lcm3usdc@ubuntuserverdc:~$
```

Y creamos de nuevo el archivo `/etc/resolv.conf` añadiendo las siguientes lineas.

```
GNU nano 7.2 /etc/resolv.conf *  
nameserver 192.168.56.101  
nameserver 8.8.8.8  
search lcmAS0.local
```

Si queremos asegurarnos de que el archivo no pueda ser modificado, ejecutamos:

```
sudo chattr +i/etc/resolv.conf
```

Ahora vamos a proceder a instalar *Samba*, para esto primero actualizamos los repositorios.

```
sudo apt-get update
```

Instalamos samba y sus dependencias.

```
lcm3usdc@ubuntuerverdc:/$ sudo apt-get install -y acl attr samba samba-dsdb-modules samba-vfs-modules smbclient winbind libpam-winbind libnss-winbind libpam-krb5 krb5-config krb5-user dnsutils chrony net-tools
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
acl ya está en su versión más reciente (2.3.2-1build1).
attr ya está en su versión más reciente (1:2.5.2-1build1).
samba ya está en su versión más reciente (2:4.19.5+dfsg-4ubuntu9).
```

Deshabilitamos los servicios que tiene el servidor de AD que no vamos a utilizar porque samba no los requiere.

```
lcm3usdc@ubuntuerverdc:/$ sudo systemctl disable --now smbd nmbd winbind
Synchronizing state of smbd.service with SysV service script with /usr/lib/systemd/systemd-sysv-install.
Executing: /usr/lib/systemd/systemd-sysv-install disable smbd
Synchronizing state of nmbd.service with SysV service script with /usr/lib/systemd/systemd-sysv-install.
Executing: /usr/lib/systemd/systemd-sysv-install disable nmbd
Synchronizing state of winbind.service with SysV service script with /usr/lib/systemd/systemd-sysv-install.
Executing: /usr/lib/systemd/systemd-sysv-install disable winbind
Removed "/etc/systemd/system/nmb.service".
Removed "/etc/systemd/system/multi-user.target.wants/nmbd.service".
Removed "/etc/systemd/system/multi-user.target.wants/winbind.service".
Removed "/etc/systemd/system/multi-user.target.wants/smbd.service".
Removed "/etc/systemd/system/smb.service".
lcm3usdc@ubuntuerverdc:/$
```

```
lcm3usdc@ubuntuerverdc:/$ sudo systemctl unmask samba-ad-dc
lcm3usdc@ubuntuerverdc:/$ sudo systemctl enable samba-ad-dc
Synchronizing state of samba-ad-dc.service with SysV service script with /usr/lib/systemd/systemd-sysv-install.
Executing: /usr/lib/systemd/systemd-sysv-install enable samba-ad-dc
lcm3usdc@ubuntuerverdc:/$
```

Creamos una copia de seguridad del archivo `/etc/samba/smb.conf`.

```
sudo mv /etc/samba/smb.conf /etc/samba/smb.conf.cp
```

Aprovisionamos el servidor como *Samba Active Directory* e indicamos los datos que nos solicita.

```
lcm3usdc@ubuntuerverdc:~$ sudo samba-tool domain provision
Realm: lcmASO.local
Domain [lcmASO]:
Server Role (dc, member, standalone) [dc]:
DNS backend (SAMBA_INTERNAL, BIND9_FLATFILE, BIND9_DLZ, NONE) [SAMBA_INTERNAL]:
DNS forwarder IP address (write 'none' to disable forwarding) [192.168.56.102]: 8.8.8.8
Administrator password:
Retype password:
```

- Realm: FQDN
- Domain: nombre de dominio
- Server role: rol del servidor
- DNS backend: servidor DNS por defecto
- DNS forwarder IP: IP servidor DNS de reenvío, en este caso ponemos el de google ya que aquí se van a remitir las peticiones que no estén dentro de nuestro dominio.

También añadimos la contraseña para administrador samba, en nuestro caso ponemos `abc123`. en un caso real debería ser una contraseña que cumpla ciertos criterios de seguridad.

Antes de continuar vamos a crear una copia de seguridad de la configuración predeterminada de *kerberos*.

```
sudo mv /etc/krb5.conf /etc/krb5.conf.cp
```

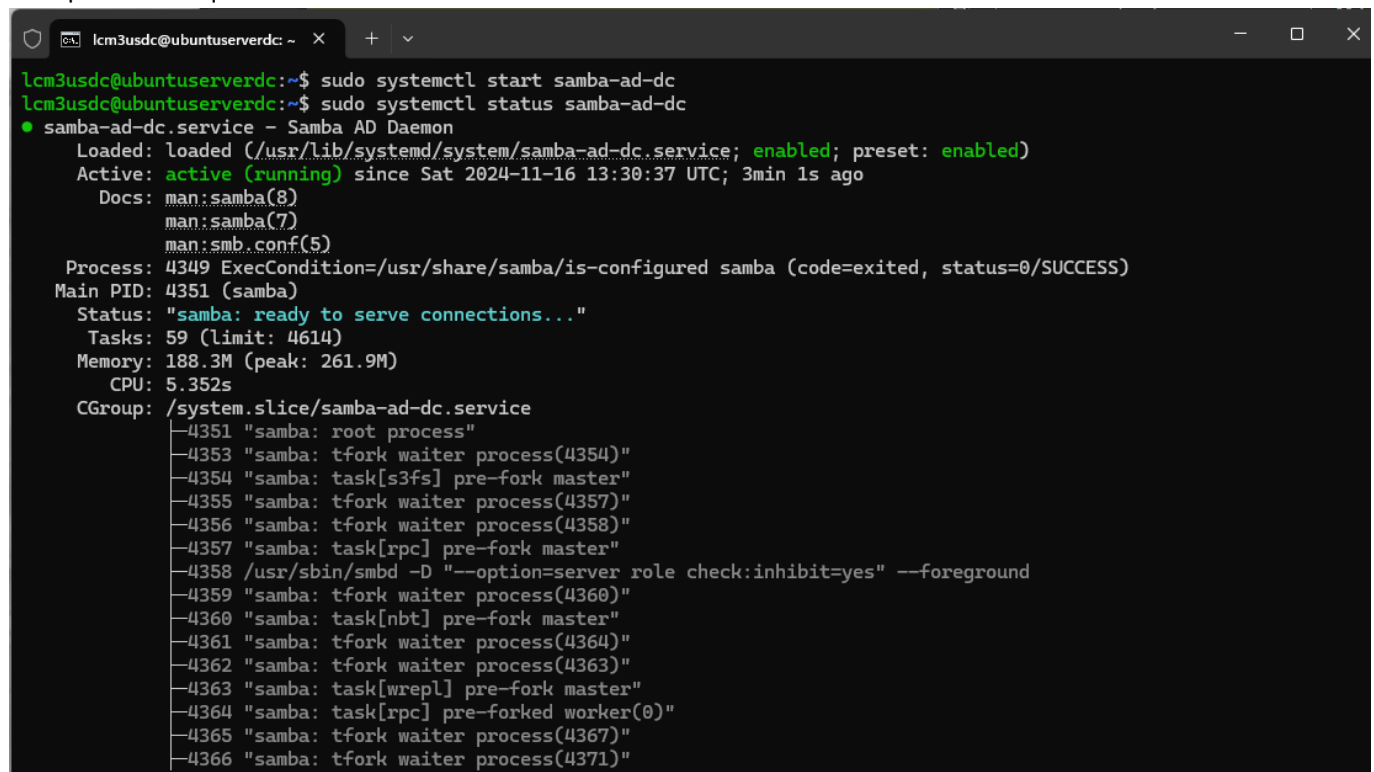
Ahora sustituimos la configuración kerberos predeterminada por la generada por Samba.

```
sudo cp /var/lib/samba/private/krb5.conf /etc/krb5.conf
```

Iniciamos el servidor samba.

```
sudo systemctl start samba-ad-dc
```

Comprobamos que esta funcionando.



```
lcm3usdc@ubuntu-serverdc: ~$ sudo systemctl start samba-ad-dc
lcm3usdc@ubuntu-serverdc: ~$ sudo systemctl status samba-ad-dc
● samba-ad-dc.service - Samba AD Daemon
   Loaded: loaded (/usr/lib/systemd/system/samba-ad-dc.service; enabled; preset: enabled)
   Active: active (running) since Sat 2024-11-16 13:30:37 UTC; 3min 1s ago
     Docs: man:samba(8)
           man:samba(7)
           man:smb.conf(5)
  Process: 4349 ExecCondition=/usr/share/samba/is-configured samba (code=exited, status=0/SUCCESS)
 Main PID: 4351 (samba)
    Status: "samba: ready to serve connections..."
   Tasks: 59 (limit: 4614)
  Memory: 188.3M (peak: 261.9M)
    CPU: 5.352s
   CGroup: /system.slice/samba-ad-dc.service
           └─4351 "samba: root process"
             └─4353 "samba: tfork waiter process(4354)"
               └─4354 "samba: task[s3fs] pre-fork master"
                 └─4355 "samba: tfork waiter process(4357)"
                   └─4356 "samba: tfork waiter process(4358)"
                     └─4357 "samba: task[rpc] pre-fork master"
                       └─4358 /usr/sbin/smbd -D "--option=server role check:inhibit=yes" --foreground
                         └─4359 "samba: tfork waiter process(4360)"
                           └─4360 "samba: task[nbt] pre-fork master"
                             └─4361 "samba: tfork waiter process(4364)"
                               └─4362 "samba: tfork waiter process(4363)"
                                 └─4363 "samba: task[wrepl] pre-fork master"
                                   └─4364 "samba: task[rpc] pre-forked worker(0)"
                                     └─4365 "samba: tfork waiter process(4367)"
                                       └─4366 "samba: tfork waiter process(4371)"
```

Configuración NTP

Samba AD es dependiente del protocolo kerberos, este necesita que los tiempos del servidor AD y la estación de trabajo estén sincronizados, para esto configuramos un servidor NTP en Samba.

Cambiamos los permisos y la propiedad del directorio *ntp_signed* para que el usuario *_chrony* tenga permitido de lectura en el directorio.


```
lcm3usdc@ubuntuserverdc: ~$ sudo chown root:_chrony /var/lib/samba/ntp_signd/
lcm3usdc@ubuntuserverdc:~$ sudo chmod 750 /var/lib/samba/ntp_signd/
lcm3usdc@ubuntuserverdc:~$
```

Modificamos el archivo de configuración `/etc/chrony/chrony.conf` para habilitar el servidor NTP y que este apunte al servidor NTP de samba `/var/lib/samba/ntp_signd`.

```
GNU nano 7.2 /etc/chrony/chrony.conf *
#log tracking measurements statistics

# Log files location.
logdir /var/log/chrony

# Stop bad estimates upsetting machine clock.
maxupdateskew 100.0

# This directive enables kernel synchronisation (every 11 minutes) of the
# real-time clock. Note that it can't be used along with the 'rtcfile' directive.
rtcsync

# Step the system clock instead of slewing it if the adjustment is larger than
# one second, but only in the first three clock updates.
makestep 1 3

# Get TAI-UTC offset and leap seconds from the system tz database.
# This directive must be commented out when using time sources serving
# leap-smeared time.
leapsectz right/UTC

# NTP Configuration
bindcmdaddress 192.168.56.102
allow 192.168.56.0/24
ntpsigndsocket /var/lib/samba/ntp_signd
```

- `bindcmdaddress 192.168.56.102` : Dirección IP en la que chronyd escucha.
- `allow 192.168.56.0/24` : Permite que los clientes de la red 192.168.56.0/24 usen este servidor para sincronización NTP.
- `ntpsigndsocket /var/lib/samba/ntp_signd` : Socket utilizado para la firma NTP requerida para integrarse con un dominio Active Directory.

Reiniciamos y verificamos el estado del servicio NTP en el servidor.

```
sudo systemctl restart chronyd
sudo systemctl status chronyd
```

```

lcm3usdc@ubuntuerverdc: ~ x Administrador: Windows Powe x + v
lcm3usdc@ubuntuerverdc:~$ sudo systemctl restart chronyd
lcm3usdc@ubuntuerverdc:~$ sudo systemctl status chronyd
● chrony.service - chrony, an NTP client/server
   Loaded: loaded (/usr/lib/systemd/system/chrony.service; enabled; preset: enabled)
   Active: active (running) since Sat 2024-11-16 13:43:43 UTC; 1s ago
     Docs: man:chronyd(8)
           man:chronyc(1)
           man:chrony.conf(5)
   Process: 4686 ExecStart=/usr/lib/systemd/scripts/chronyd-starter.sh $DAEMON_OPTS (code=exited, status=0/SUCCESS)
   Main PID: 4695 (chronyd)
     Tasks: 2 (limit: 4614)
   Memory: 1.3M (peak: 2.0M)
      CPU: 122ms
   CGroup: /system.slice/chrony.service
           └─4695 /usr/sbin/chronyd -F 1
             └─4696 /usr/sbin/chronyd -F 1

nov 16 13:43:43 ubuntuerverdc systemd[1]: Starting chrony.service - chrony, an NTP client/server...
nov 16 13:43:43 ubuntuerverdc chronyd[4695]: chronyd version 4.5 starting (+CMDMON +NTP +REFCLOCK +RTC +PRIVDROP +SCFI
nov 16 13:43:43 ubuntuerverdc chronyd[4695]: Loaded 0 symmetric keys
nov 16 13:43:43 ubuntuerverdc chronyd[4695]: Frequency 10.773 +/- 0.606 ppm read from /var/lib/chrony/chrony.drift
nov 16 13:43:43 ubuntuerverdc chronyd[4695]: Using right/UTC timezone to obtain leap second data
nov 16 13:43:43 ubuntuerverdc chronyd[4695]: MS-SNTP authentication enabled
nov 16 13:43:43 ubuntuerverdc chronyd[4695]: Loaded seccomp filter (level 1)
nov 16 13:43:43 ubuntuerverdc systemd[1]: Started chrony.service - chrony, an NTP client/server.
lines 1-23/23 (END)

```

Configuración NFTABLES

Para poder instalar los paquetes necesarios en nuestra máquina cliente necesitamos que esta tenga acceso a internet, para ello vamos a instalar nftables y crear una regla para que las peticiones de red de dicha máquina pasen por el servidor y sean redirigidas por el adaptador que el servidor tiene conectado en modo NAT y por lo tanto salda a internet.

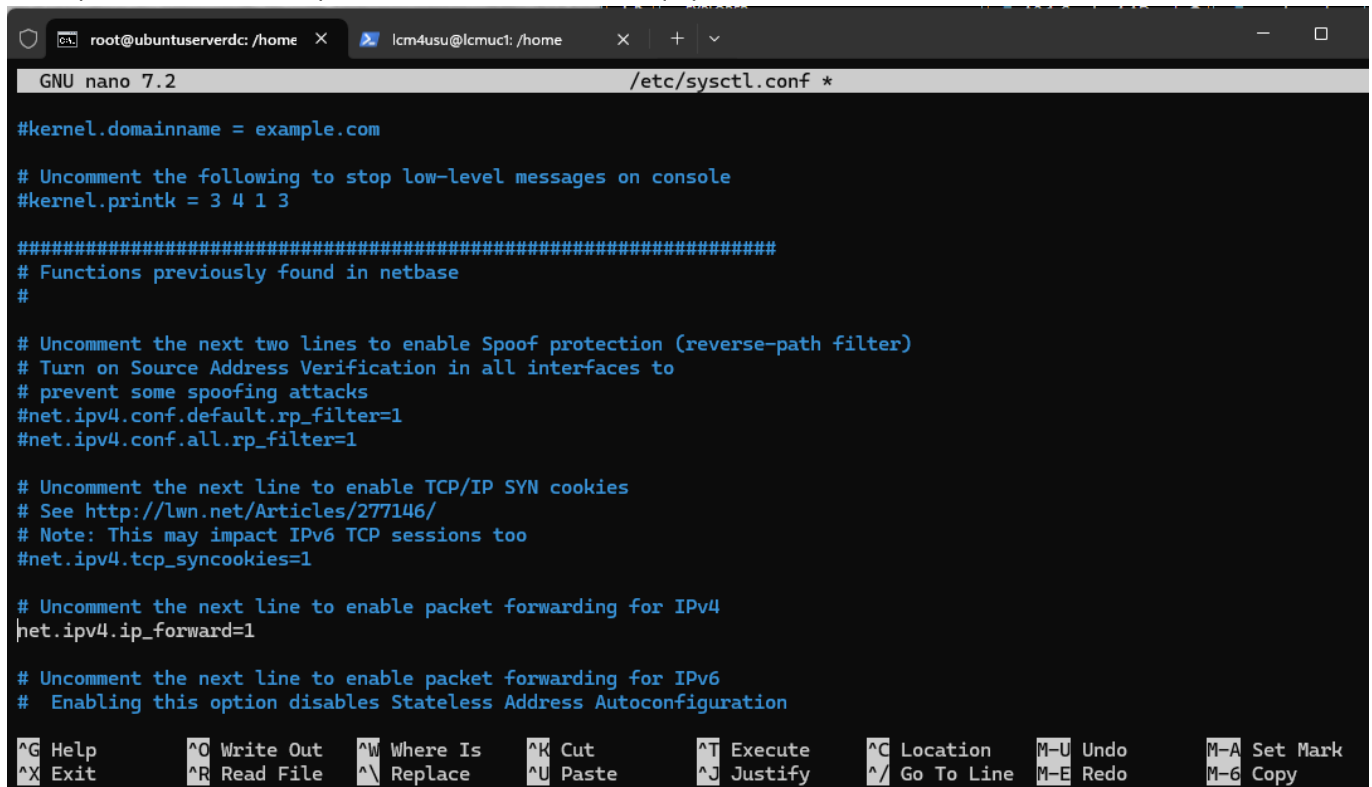
Primero debemos habilitar el reenvío de paquetes en el servidor.

```

root@ubuntuerverdc: /home x lcm4usu@lcmuc1: /home x + v
lcm3usdc@ubuntuerverdc:~$ sudo su
[sudo] password for lcm3usdc:
root@ubuntuerverdc:/home/lcm3usdc# cat /proc/sys/net/ipv4/ip_forward
0
root@ubuntuerverdc:/home/lcm3usdc# echo 1 > /proc/sys/net/ipv4/ip_forward
root@ubuntuerverdc:/home/lcm3usdc#

```

Para que la modificación permanezca al reiniciar el equipo.



```

GNU nano 7.2 /etc/sysctl.conf *

#kernel.domainname = example.com

# Uncomment the following to stop low-level messages on console
#kernel.printk = 3 4 1 3

#####
# Functions previously found in netbase
#

# Uncomment the next two lines to enable Spoof protection (reverse-path filter)
# Turn on Source Address Verification in all interfaces to
# prevent some spoofing attacks
#net.ipv4.conf.default.rp_filter=1
#net.ipv4.conf.all.rp_filter=1

# Uncomment the next line to enable TCP/IP SYN cookies
# See http://lwn.net/Articles/277146/
# Note: This may impact IPv6 TCP sessions too
#net.ipv4.tcp_syncookies=1

# Uncomment the next line to enable packet forwarding for IPv4
net.ipv4.ip_forward=1

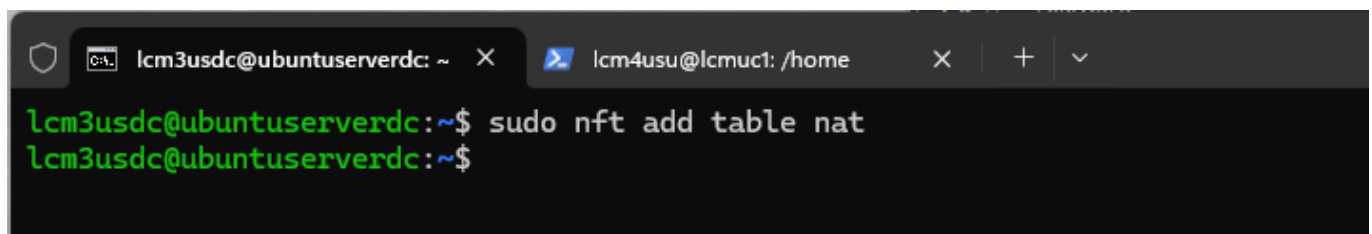
# Uncomment the next line to enable packet forwarding for IPv6
# Enabling this option disables Stateless Address Autoconfiguration

^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute    ^C Location   M-U Undo      M-A Set Mark
^X Exit      ^R Read File  ^_ Replace    ^U Paste      ^J Justify    ^_/ Go To Line  M-E Redo      M-6 Copy
  
```

Instalamos nftables.

```
sudo apt-get install nftables
```

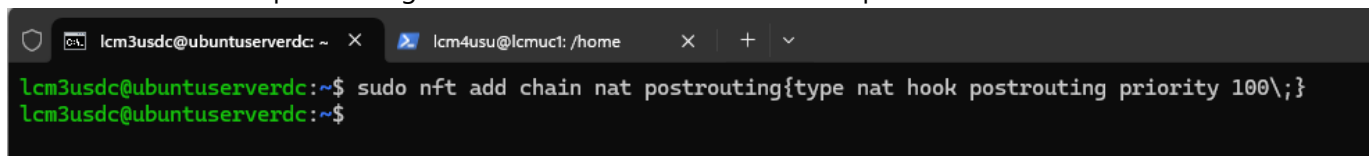
Creamos una tabla nat.



```

lcm3usdc@ubuntuserverdc: ~$ sudo nft add table nat
lcm3usdc@ubuntuserverdc: ~$
  
```

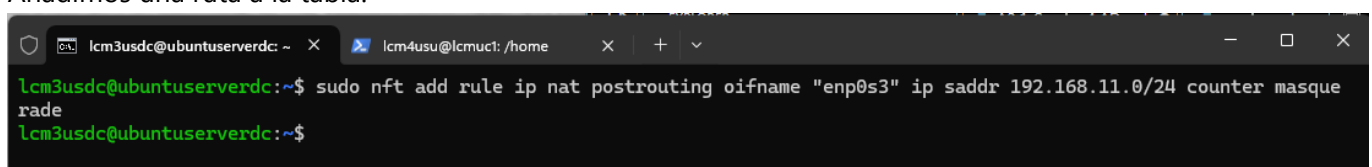
Creamos una cadena postrouting a la tabla nat añadiendo atributos aplicables a esta cadena.



```

lcm3usdc@ubuntuserverdc: ~$ sudo nft add chain nat postrouting{type nat hook postrouting priority 100\;}
lcm3usdc@ubuntuserverdc: ~$
  
```

Añadimos una ruta a la tabla.



```

lcm3usdc@ubuntuserverdc: ~$ sudo nft add rule ip nat postrouting oifname "enp0s3" ip saddr 192.168.11.0/24 counter masquerade
lcm3usdc@ubuntuserverdc: ~$
  
```

Comprobamos que la regla se ha añadido a la tabla.

```
lcm3usdc@ubuntuserverdc:~$ sudo nft list ruleset
table ip nat {
    chain postrouting {
        type nat hook postrouting priority srcnat; policy accept;
        oifname "enp0s3" ip saddr 192.168.11.0/24 counter packets 0 bytes 0 masquerade
    }
}
lcm3usdc@ubuntuserverdc:~$
```

Guardamos las reglas.

```
root@ubuntuserverdc:/home/lcm3usdc# nft list ruleset > /etc/nftables.conf
root@ubuntuserverdc:/home/lcm3usdc# nano /etc/nftables.conf
root@ubuntuserverdc:/home/lcm3usdc#
```

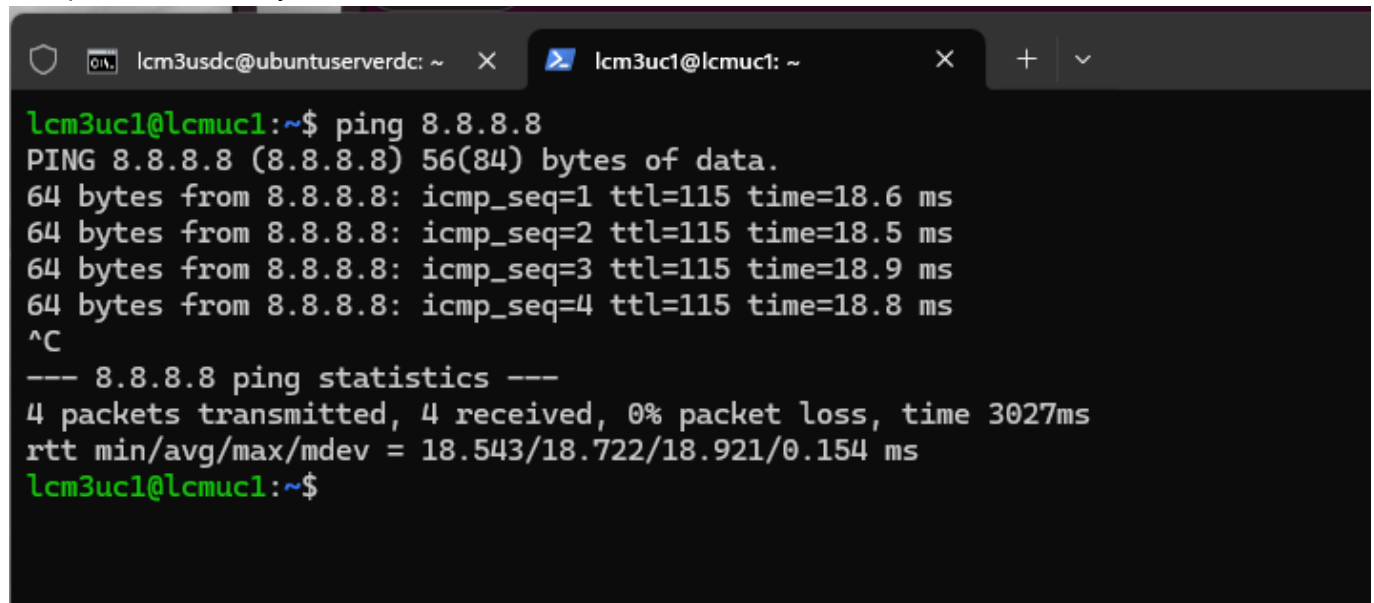
```
GNU nano 7.2 /etc/nftables.conf
table ip nat {
    chain postrouting {
        type nat hook postrouting priority srcnat; policy accept;
        oifname "enp0s3" ip saddr 192.168.11.0/24 counter packets 0 bytes 0 masquerade
        oifname "enp0s3" ip saddr 192.168.11.0/24 counter packets 0 bytes 0 masquerade
    }
}

[ Read 7 lines ]
^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute   ^C Location  M-U Undo
^X Exit      ^R Read File ^\ Replace   ^U Paste     ^J Justify   ^_ Go To Line M-E Redo
```

Habilitamos el servicio y lo reiniciamos.

```
lcm3usdc@ubuntuserverdc:~$ sudo systemctl enable nftables
lcm3usdc@ubuntuserverdc:~$ sudo systemctl restart nftables
lcm3usdc@ubuntuserverdc:~$
```

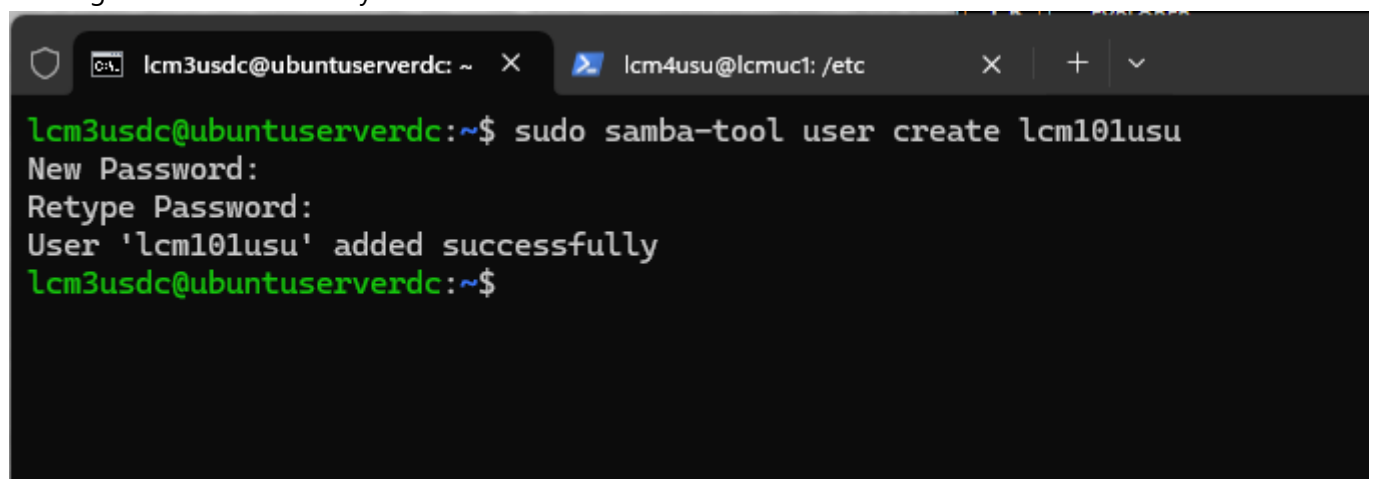
Comprobamos como ya tenemos red desde el cliente.



```
lcm3uc1@lcmuc1:~$ ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=115 time=18.6 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=115 time=18.5 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=115 time=18.9 ms
64 bytes from 8.8.8.8: icmp_seq=4 ttl=115 time=18.8 ms
^C
--- 8.8.8.8 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3027ms
rtt min/avg/max/mdev = 18.543/18.722/18.921/0.154 ms
lcm3uc1@lcmuc1:~$
```

Crear usuarios Samba

Nos logueamos en el server y creamos un usuario samba **lcm101usu**.



```
lcm3usdc@ubuntuserverdc:~$ sudo samba-tool user create lcm101usu
New Password:
Retype Password:
User 'lcm101usu' added successfully
lcm3usdc@ubuntuserverdc:~$
```

Cliente

Nos conectamos a la máquina cliente a través de ssh desde el host, para ello nos conectamos al server por ssh y una vez dentro del server nos conectamos por ssh al cliente.

Ahora vamos a unir el equipo cliente al dominio.

En primer lugar comprobamos el nombre del equipo cliente al verificar que es muy largo y el sistema nos va a devolver error procedemos a modificarlo por uno más corto, en este caso **lcmuc1**.

```
lcm3uc1@lcmubuntuclient1:~$ hostname
lcmubuntuclient1
lcm3uc1@lcmubuntuclient1:~$ sudo hostnamectl set-hostname lcmuc1
[sudo] contraseña para lcm3uc1:
lcm3uc1@lcmubuntuclient1:~$
```

Ahora modificamos el archivo `/etc/hosts` para que resuelva al servidor Ubuntu ya configurado.

```
GNU nano 6.2 /etc/hosts *
127.0.0.1    localhost
127.0.1.1    lcmubuntuclient1
192.168.56.102 ubuntuerverdc.lcmASO.local
192.168.56.102 lcmASO.local
# The following lines are desirable for IPv6 capable hosts
::1         ip6-localhost ip6-loopback
fe00::0     ip6-localnet
ff00::0     ip6-mcastprefix
ff02::1     ip6-allnodes
ff02::2     ip6-allrouters
```

Instalación de aplicativos

Instalamos `ntpd` para poder sincronizar los relojes de nuestras máquinas.

```
sudo apt-get install ntpdate
```

Utilizamos la herramienta `ntpdate` para sincronizar.

```
lcm3uc1@lcmubuntuclient1:~$ sudo ntpdate lcmASO.local
16 Nov 15:16:22 ntpdate[3466]: step time server 192.168.56.102 offset -1.919282 sec
lcm3uc1@lcmubuntuclient1:~$
```

Instalamos los paquetes necesarios.

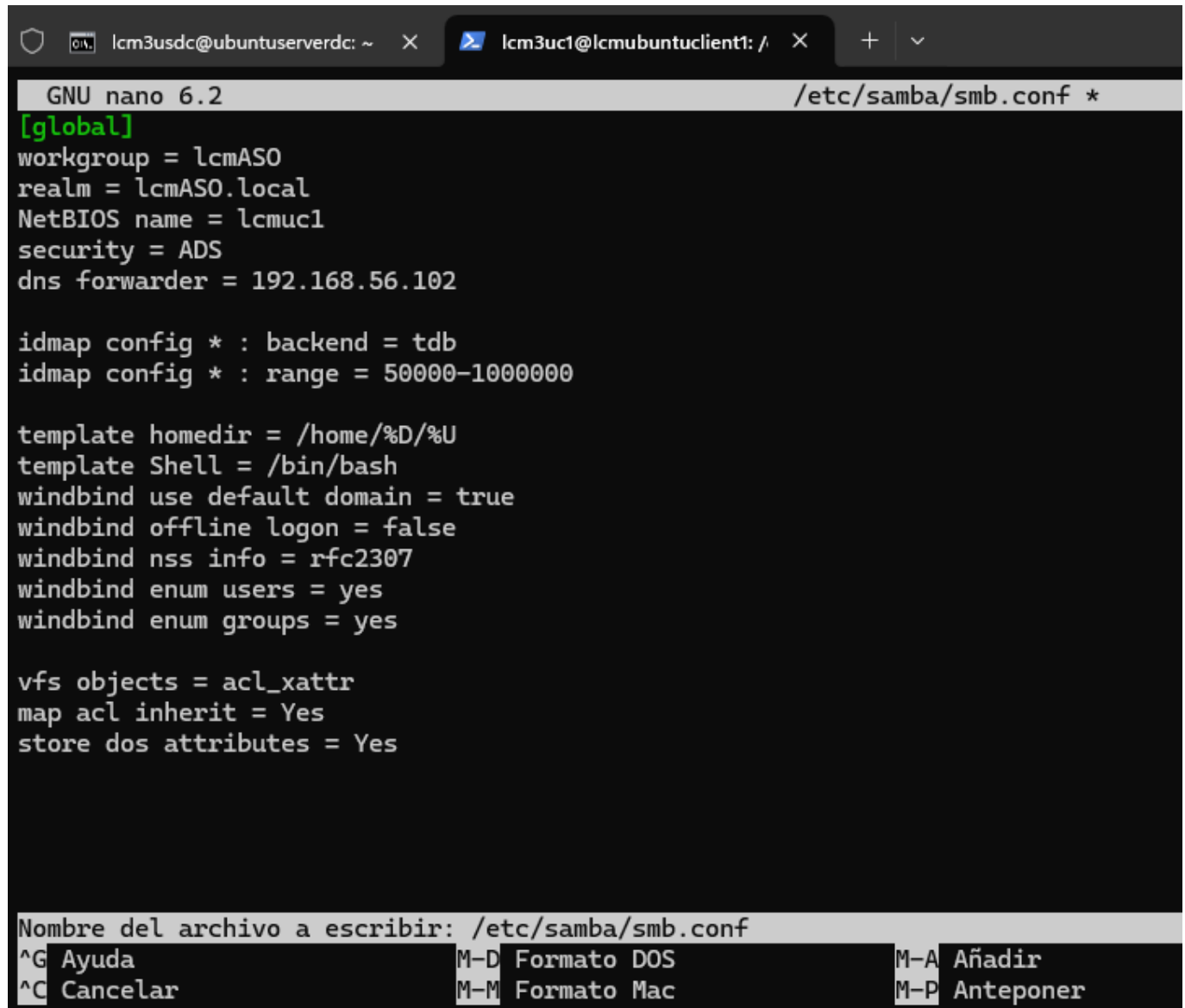
```
lcm3uc1@lcmubuntuclient1:~$ sudo apt-get install samba krb5-config krb5-user winbind libpam-winbind libnss-winbind
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
```

Unión al dominio

Realizamos una copia del archivo de configuración de samba.

```
sudo mv /etc/samba/smb.conf /etc/samba/smb.conf.cp
```

Generamos un nuevo archivo `/etc/samba/smb.conf`.



```
GNU nano 6.2 /etc/samba/smb.conf *
[global]
workgroup = lcmASO
realm = lcmASO.local
NetBIOS name = lcmuc1
security = ADS
dns forwarder = 192.168.56.102

idmap config * : backend = tdb
idmap config * : range = 50000-1000000

template homedir = /home/%D/%U
template Shell = /bin/bash
windbind use default domain = true
windbind offline logon = false
windbind nss info = rfc2307
windbind enum users = yes
windbind enum groups = yes

vfs objects = acl_xattr
map acl inherit = Yes
store dos attributes = Yes

Nombre del archivo a escribir: /etc/samba/smb.conf
^G Ayuda          M-D Formato DOS  M-A Añadir
^C Cancelar       M-M Formato Mac  M-P Anteponer
```

Paramos los servicios samba que no son necesarios para nuestro cliente.

```
sudo systemctl stop samba-ad-dc
```

Habilitamos los servicios samba.

```
lcm3uc1@lcmubuntuclient1:/etc$ sudo systemctl enable smbd nmbd
Synchronizing state of smbd.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable smbd
Synchronizing state of nmbd.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable nmbd
lcm3uc1@lcmubuntuclient1:/etc$
```

Ahora vamos a unir al cliente al dominio.

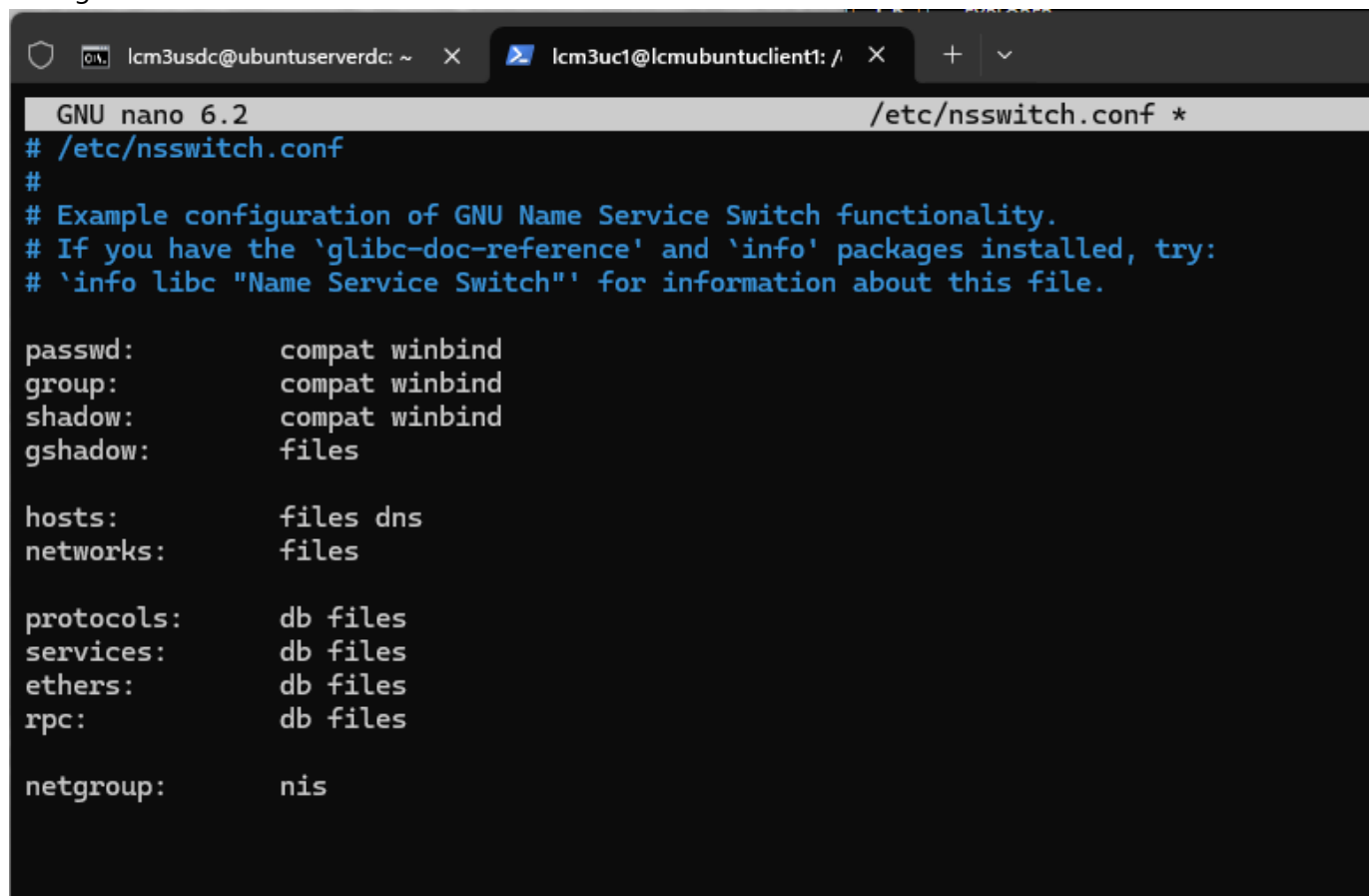
```
lcm3uc1@lcmubuntuclient1:/etc$ sudo net ads join -U administrator
Password for [LCMASO\administrator]:
Using short domain name -- LCMASO
Joined 'LCMUC1' to dns domain 'lcmaso.local'
No DNS domain configured for lcmuc1. Unable to perform DNS Update.
DNS update failed: NT_STATUS_INVALID_PARAMETER
lcm3uc1@lcmubuntuclient1:/etc$
```

NOTA:El mensaje de error respecto al DNS es correcto, una vez lo hayamos configurado se resolverá

Comprobamos como en la lista de equipos del servidor ya aparece nuestro equipo cliente **lcmuc1**.

```
lcm3usdc@ubuntuserverdc:~$ sudo samba-tool computer list
LCMUC1$
UBUNTUSERVERDC$
lcm3usdc@ubuntuserverdc:~$
```


Configuramos la autenticación de cuentas AD.



```
GNU nano 6.2 /etc/nsswitch.conf *
# /etc/nsswitch.conf
#
# Example configuration of GNU Name Service Switch functionality.
# If you have the 'glibc-doc-reference' and 'info' packages installed, try:
# 'info libc "Name Service Switch"' for information about this file.

passwd:          compat winbind
group:           compat winbind
shadow:          compat winbind
gshadow:         files

hosts:           files dns
networks:        files

protocols:       db files
services:        db files
ethers:          db files
rpc:             db files

netgroup:        nis
```

Reiniciamos el servicio winbind.

```
sudo systemctl restart winbind
```

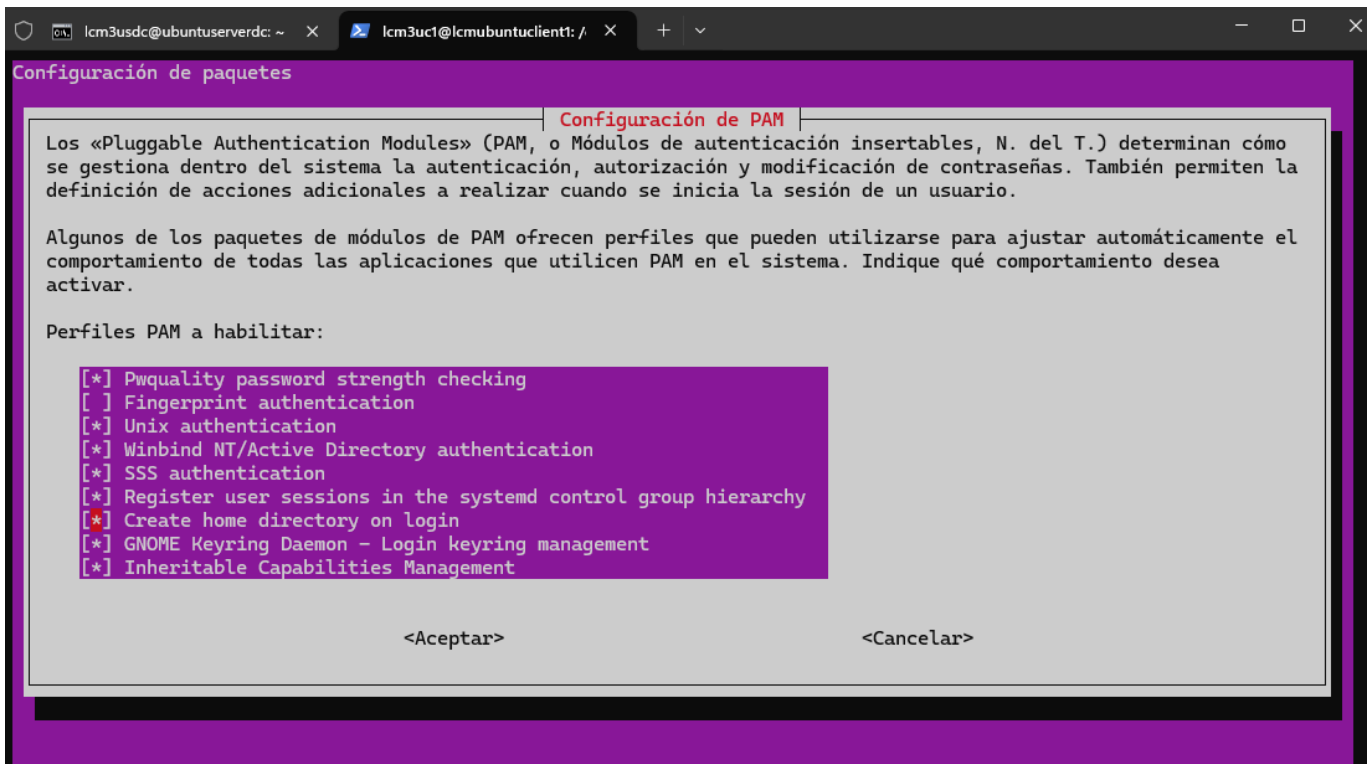
Listamos usuarios y grupos de dominio.

```
lcm3uc1@lcmubuntuclient1:/etc$ wbinfo -u
krbtgt
administrator
guest
lcm3uc1@lcmubuntuclient1:/etc$ wbinfo -g
enterprise admins
dnsupdateproxy
ras and ias servers
schema admins
domain guests
cert publishers
dnsadmins
denied rodc password replication group
domain controllers
enterprise read-only domain controllers
domain users
group policy creator owners
protected users
domain admins
read-only domain controllers
domain computers
allowed rodc password replication group
lcm3uc1@lcmubuntuclient1:/etc$
```

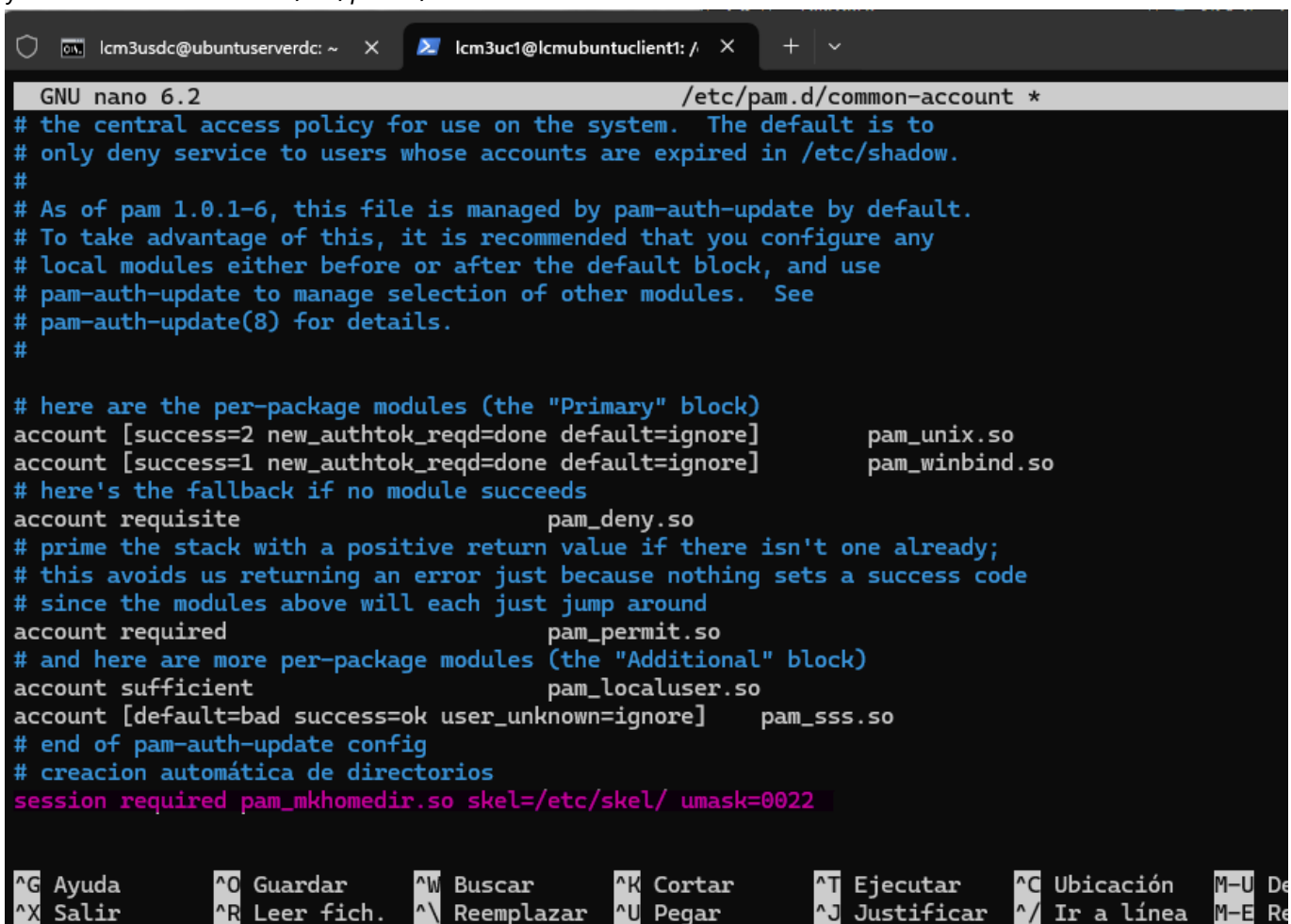
```
lcm3uc1@lcmubuntuclient1:/etc$ sudo getent passwd | grep administrator
administrator:*:50001:50000:./home/LCMASO/administrator:/bin/bash
lcm3uc1@lcmubuntuclient1:/etc$ sudo getent group | grep 'domain admins'
domain admins:x:50013:
lcm3uc1@lcmubuntuclient1:/etc$
```

Configuramos nuestro equipo para autenticarlo con cuentas de dominio y que se creen automáticamente los directorios a través del modulo *pam-auth*.

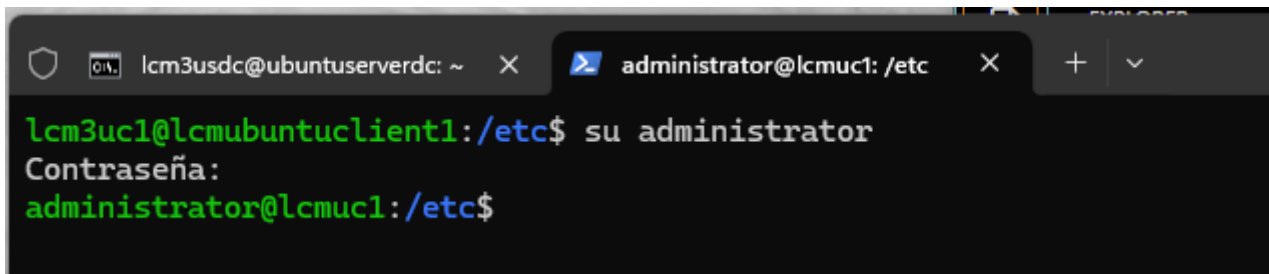
```
sudo pam-auth-update
```



y modificando el archivo `/etc/pam.d/common-account`.



Comprobamos como ahora podemos acceder desde nuestro equipo cliente a un usuario de dominio **administrator**.

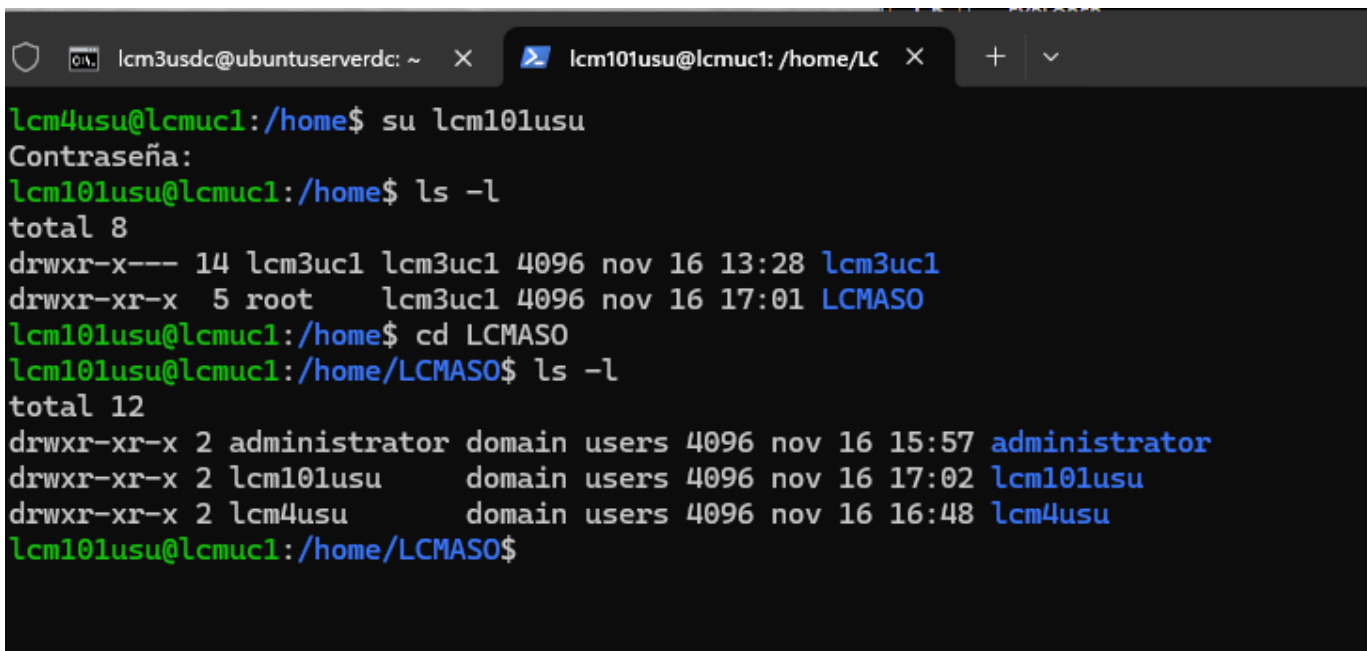


```
lcm3usdc@ubuntuserverdc: ~  
administrator@lcmuc1: /etc  
lcm3uc1@lcmubuntuclient1:/etc$ su administrator  
Contraseña:  
administrator@lcmuc1:/etc$
```

El usuario administrator es un usuario samba AD por lo que no tiene permisos de sudo en los equipos locales, para solventar esto lo añadimos al archivo *sudoers*.

```
sudo usermod -aG sudo administrator
```

Comprobamos como podemos acceder desde la máquina cliente con el usuario creado anteriormente *lcm101usu*, también verificamos que se genera el directorio correspondiente para el usuario dentro del directorio */home*



```
lcm3usdc@ubuntuserverdc: ~  
lcm101usu@lcmuc1: /home/LC  
lcm4usu@lcmuc1:/home$ su lcm101usu  
Contraseña:  
lcm101usu@lcmuc1:/home$ ls -l  
total 8  
drwxr-x--- 14 lcm3uc1 lcm3uc1 4096 nov 16 13:28 lcm3uc1  
drwxr-xr-x  5 root      lcm3uc1 4096 nov 16 17:01 LCMASO  
lcm101usu@lcmuc1:/home$ cd LCMASO  
lcm101usu@lcmuc1:/home/LCMASO$ ls -l  
total 12  
drwxr-xr-x 2 administrator domain users 4096 nov 16 15:57 administrator  
drwxr-xr-x 2 lcm101usu      domain users 4096 nov 16 17:02 lcm101usu  
drwxr-xr-x 2 lcm4usu        domain users 4096 nov 16 16:48 lcm4usu  
lcm101usu@lcmuc1:/home/LCMASO$
```