

**PONTIFICIA UNIVERSIDAD CATÓLICA DEL  
PERÚ**

**Facultad de Ciencias e Ingeniería**

**ADMINISTRACIÓN DE SISTEMAS OPERATIVOS**

**LABORATORIO 6**

**Alumno:** Saymon Nicho

**Profesor:** Mario Carpio

**Horario:** 0781

Lima, 26 de noviembre del 2024

# TABLA DE CONTENIDOS

---

<b>PREGUNTA 1</b>	<b>3</b>
Figura 1.1. Resumen de instancias creadas: GW, DC y DCSEC	3
Figura 1.2. Conexión a la instancia GW con Remmina	3
Figura 1.3. Evidencia del hostname para GW	4
Figura 1.4. Evidencia del nuevo hostname para DC	4
Figura 1.5. Evidencia del nuevo hostname para DCSEC	5
Figura 1.6. Instalación del rol de AD DS	5
Figura 1.7. Rol de AD DS instalado	6
Figura 1.8. Evidencia de creación de la zona DNS y el tipo del dominio	6
Figura 1.9. Configuración del servidor DNS	7
Figura 1.10. Vinculación exitosa tras el ingreso de credenciales	7
Figura 1.11. Full device name actualizado con el dominio	8
<b>PREGUNTA 2</b>	<b>9</b>
Figura 2.1. Acceso a DCSEC con el usuario Nicho.adso\Administrator	9
Figura 2.2. Instalación del rol de AD DS en DCSEC	9
Figura 2.3. Dominio con nombre Nicho.adso	10
Figura 2.4. Script de PowerShell equivalente	10
Figura 2.5. Evidencia de asignación de DCSEC como controlador del dominio	11
<b>PREGUNTA 3</b>	<b>12</b>
Figura 3.1. Ejecución de netdom query fsmo cuando DC los tiene	12
Figura 3.2. Paso de los roles FSMO a DCSEC desde Powershell	13
Figura 3.3. Roles FSMO adquiridos por DCSEC	13
Figura 3.4. Ejecución de netdom query fsmo cuando DCSEC los tiene	14
<b>PREGUNTA 4</b>	<b>15</b>
Figura 4.1. Full device name de SERV con el dominio	15
Figura 4.2. Permisos de Full Control para los administradores del dominio	15
Figura 4.3. Directorio Data creado en el escritorio de DC	16
Figura 4.4. Contenido del directorio Data	16
Figura 4.5. Instalación del servicio de backups	17
Figura 4.6. Creación de un custom backup	17
Figura 4.7. Se selecciona la carpeta Data para la copia a realizar	18
Figura 4.8. Configuración de la hora para los backups automáticos	18
Figura 4.9. Configuración de la hora para los backups automáticos	19
Figura 4.10. Backups automáticos configurados	19
Figura 4.11. Creación de un Backup de forma manual	20
Figura 4.12. Resultado del backup creado en la instancia SERV	20
Figura 4.13. Se borra el contenido de Data para simular un evento no deseado	21
Figura 4.14. Se selecciona la opción de recuperar directorio remoto	21
Figura 4.15. Se indica la dirección del directorio a recuperar	22
Figura 4.16. Se indica la fecha del backup realizado	22
Figura 4.17. Se seleccionan los contenidos a recuperar	23

Figura 4.18. Se indica que los contenidos deben volver a la carpeta Data	23
Figura 4.19. Resultado de la operación de recuperación	24
Figura 4.20. Archivos recuperados	24
<b>PREGUNTA 5</b>	<b>25</b>
Figura 5.1. Resumen de creación de la instancia Ubuntu	25
Figura 5.2. Conexión por ssh a la instancia Ubuntu	25
Figura 5.3. Resultado de ejecución del comando sudo apt install lxc	26
Figura 5.4. Manual de uso del comando lxc	26
Figura 5.5. Creación de un contenedor	27
Figura 5.6. Listado de contenedores creados	27
Figura 5.7. Configuración de los archivos index.html para cada contenedor	28
Figura 5.8. Conexión a cada página web con curl	28
Figura 5.9. Instalación de openssh-server en el contenedor1	29
Figura 5.10. Instalación de openssh-server en el contenedor2	29
Figura 5.11. Conexión al contenedor1 con el usuario test	30
Figura 5.12. Conexión al contenedor2 con el usuario test	30

# PREGUNTA 1

Primero, se crean las instancias según las especificaciones solicitadas. Para las instancias DC y DCSec se usa la subnet pública ubicada en la zona us-east-1b. Además, se reusa el grupo de seguridad MiDominio.

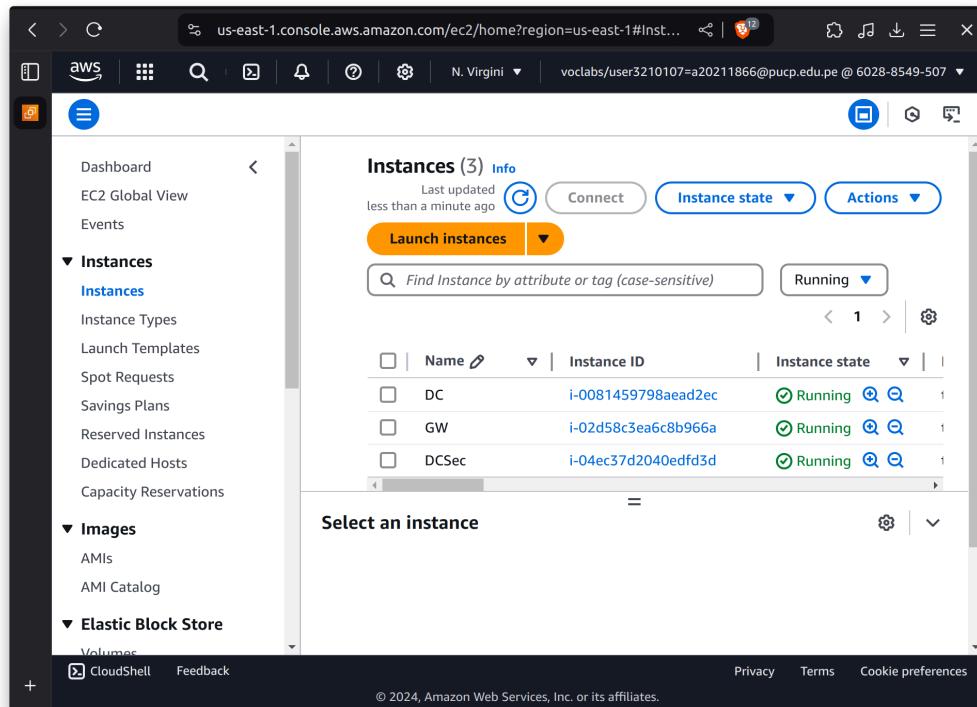


Figura 1.1. Resumen de instancias creadas: GW, DC y DCSec

Se usa el cliente de escritorio de Remmina (mi computadora personal es Linux).

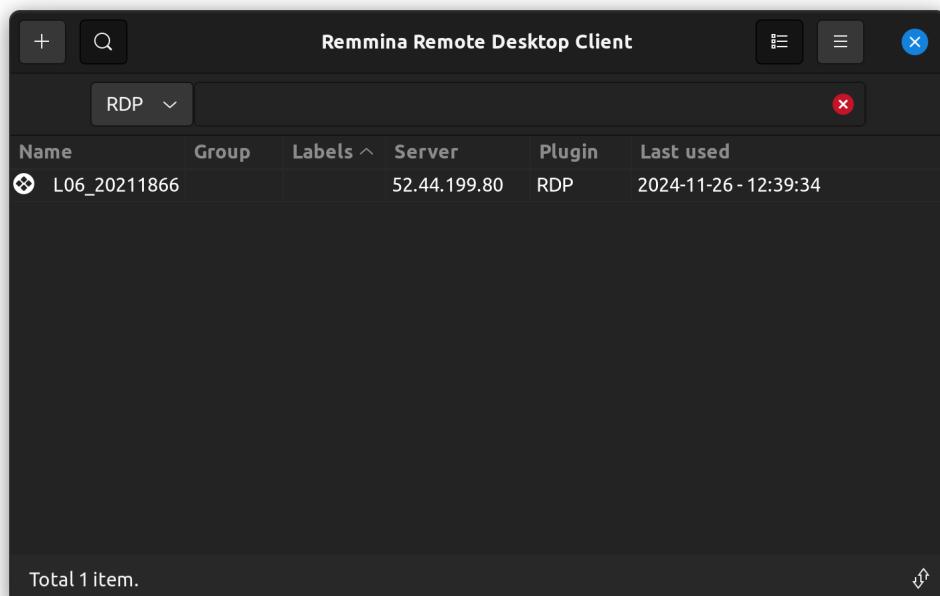


Figura 1.2. Conexión a la instancia GW con Remmina

Luego se procede a cambiar el nombre de cada instancia. Esto se muestra en las siguientes capturas.

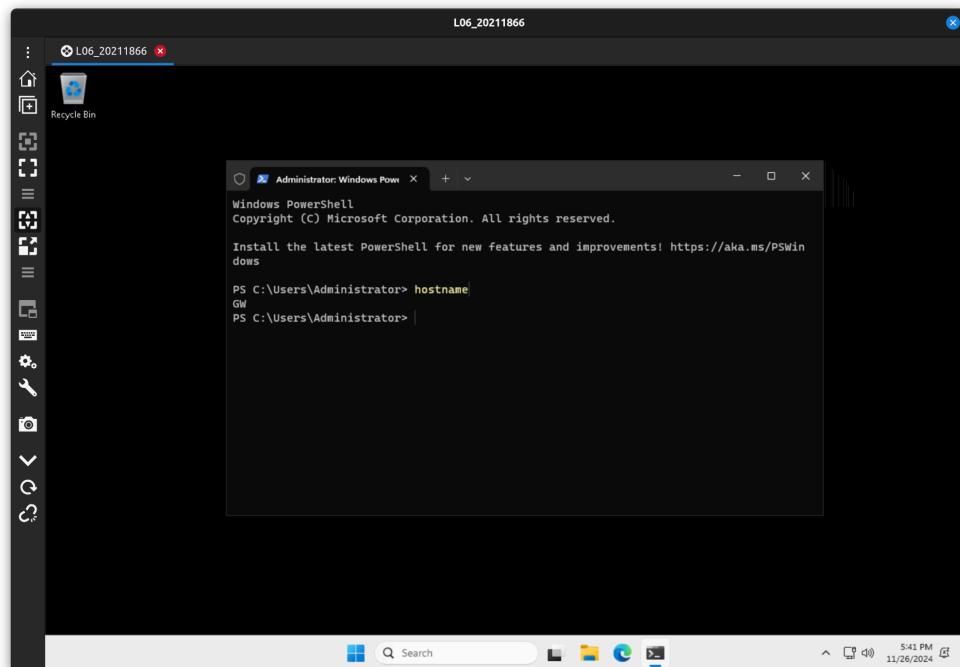


Figura 1.3. Evidencia del hostname para GW

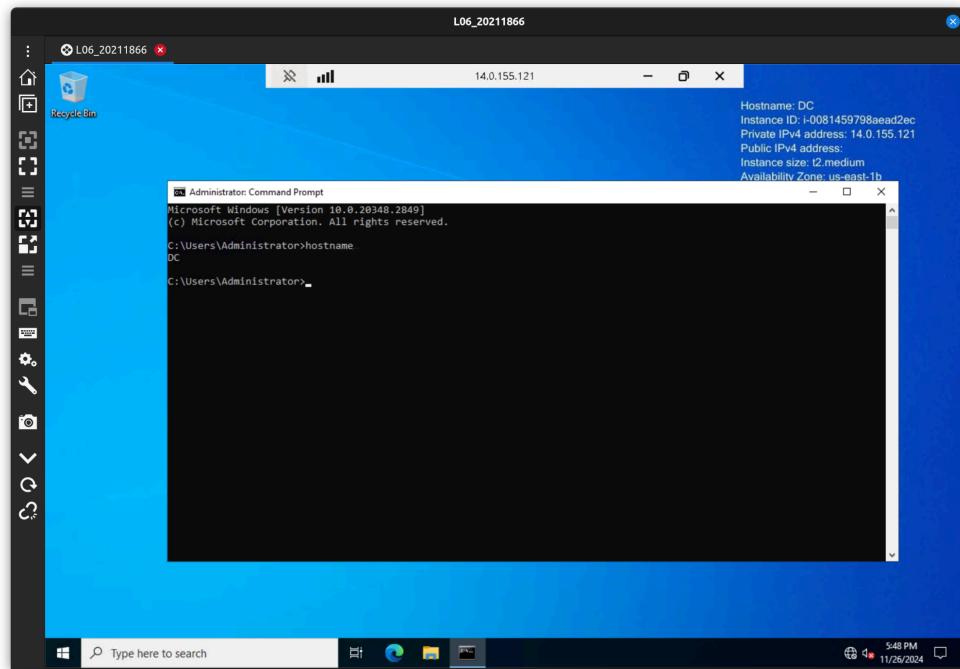


Figura 1.4. Evidencia del nuevo hostname para DC

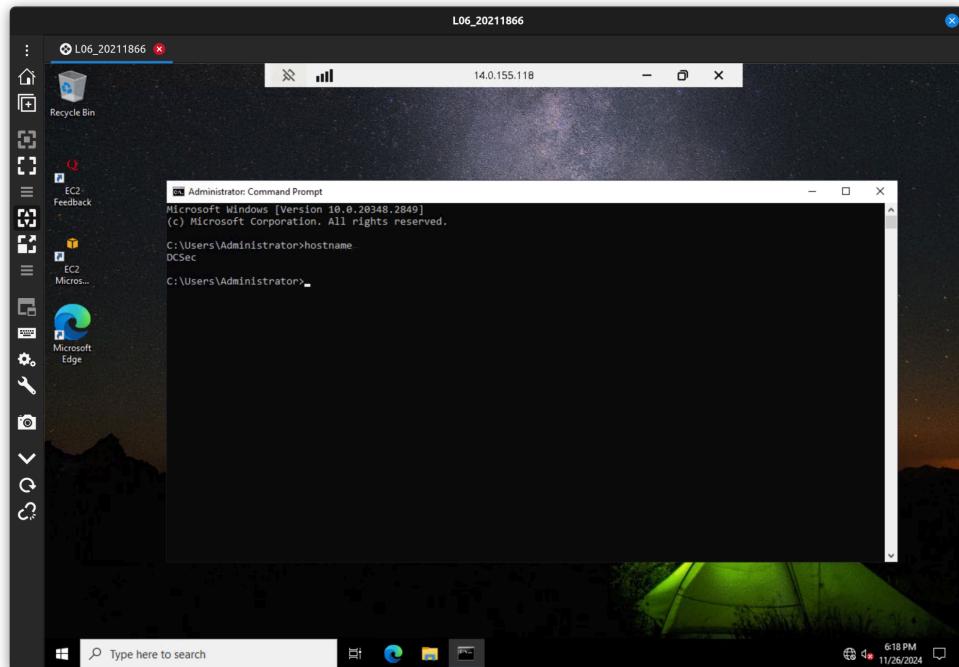


Figura 1.5. Evidencia del nuevo hostname para DCSEC

Se accede a la instancia DC y desde Server Manager se selecciona la opción para Agregar Roles y Características. Se elige la opción para instalar el rol de AD DS.

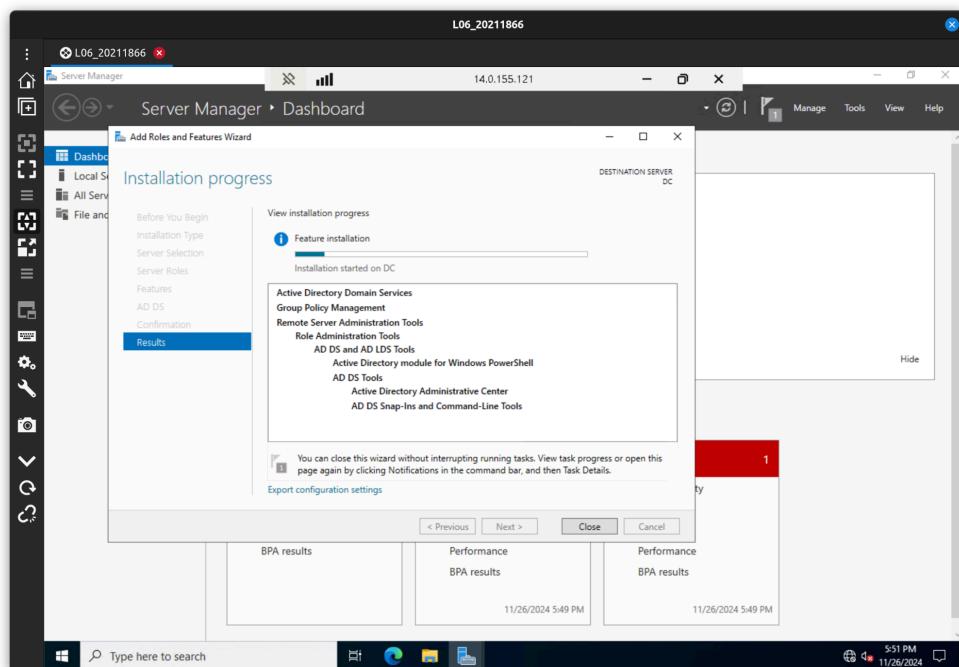


Figura 1.6. Instalación del rol de AD DS

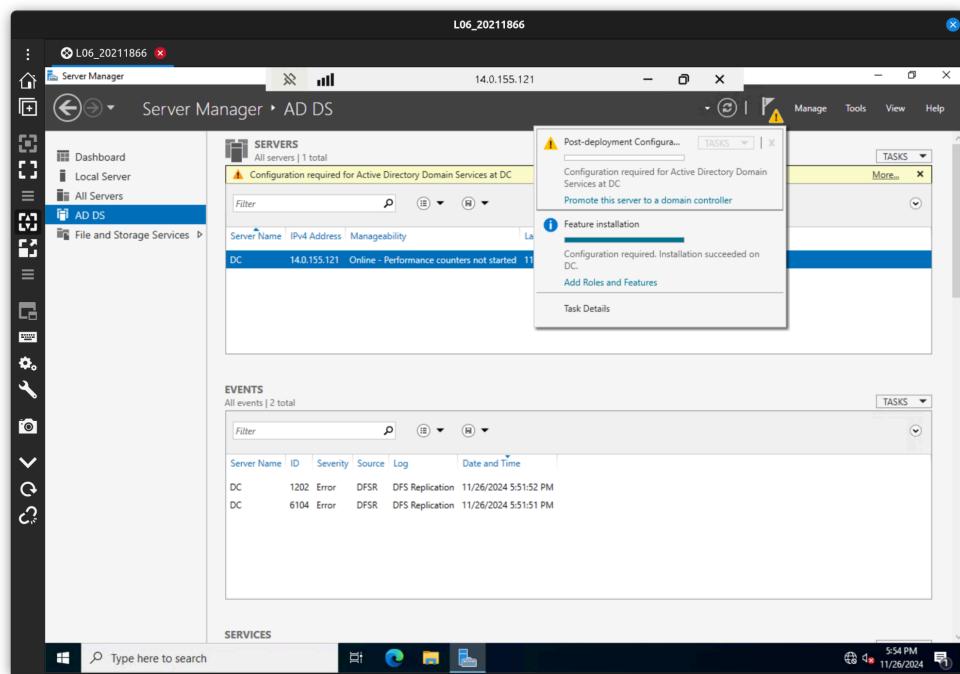


Figura 1.7. Rol de AD DS instalado

Luego, se realiza la promoción del servidor a Controlador de Dominio para Nicho.adso, igual a como se realizó en el laboratorio pasado. Para ello, se selecciona el nivel más alto disponible que es 2016, se configura una contraseña para la restauración de servicios del directorio, y se configura el nombre NETBios del dominio el cual es NICHO.

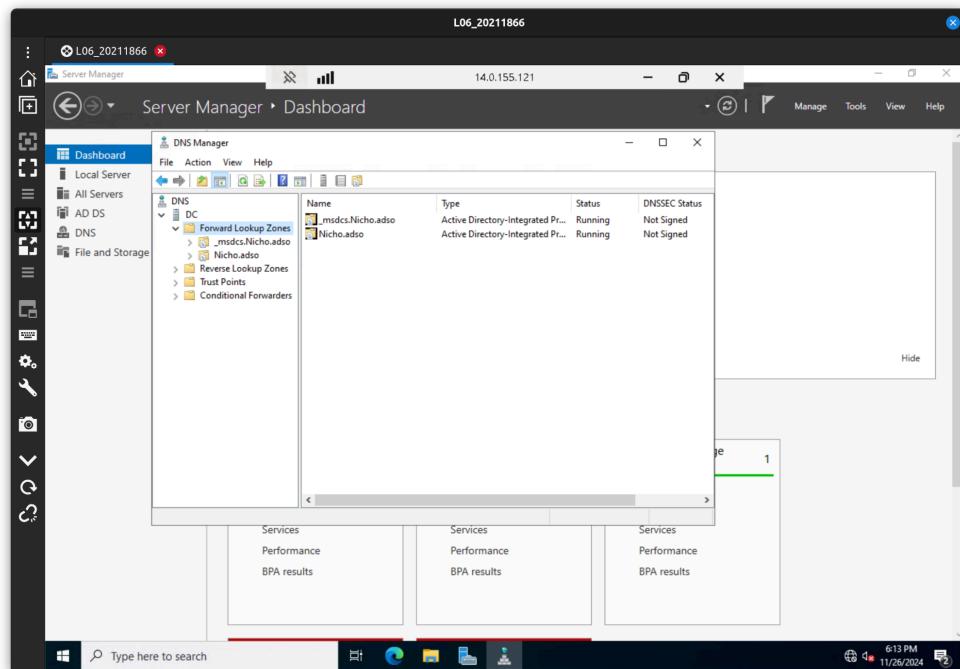


Figura 1.8. Evidencia de creación de la zona DNS y el tipo del dominio

Para agregar DCSEC como servidor miembro se configura su DNS Server Address desde las propiedades del adaptador de red.

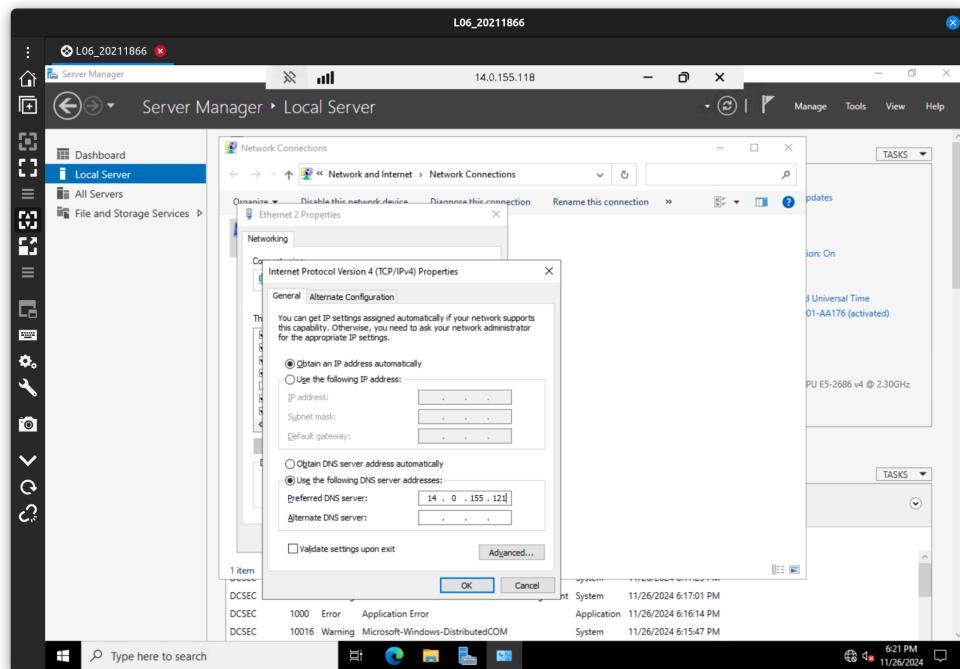


Figura 1.9. Configuración del servidor DNS

Luego, se vincula el servidor DCSEC con Server Manager.

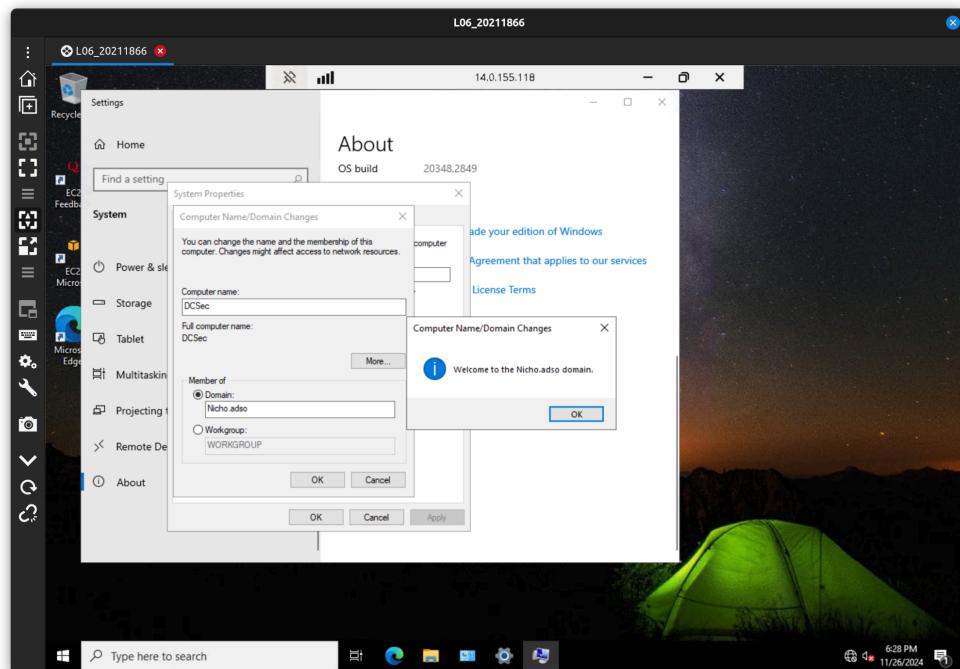


Figura 1.10. Vinculación exitosa tras el ingreso de credenciales

Finalmente, se prueba la conexión a DCSec con el usuario Administrator de DC. La conexión es exitosa y también se puede comprobar que el servidor se ha unido al dominio al observar el nombre completo del equipo desde la información del sistema.

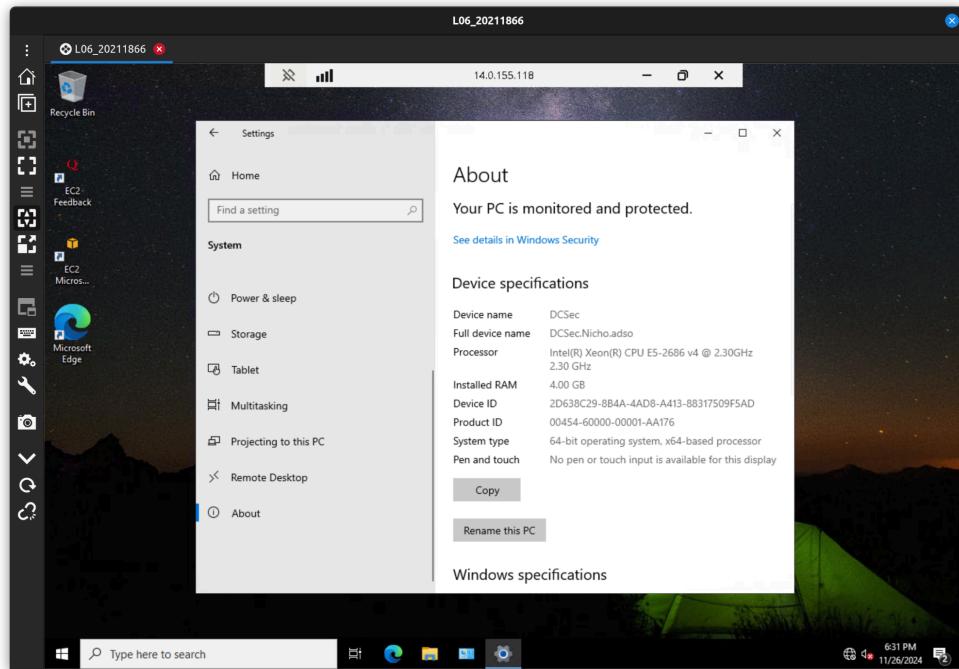


Figura 1.11. Full device name actualizado con el dominio

# PREGUNTA 2

Primero, se realiza la conexión a la instancia DCSec con la cuenta del administrador que posee privilegios en el dominio.

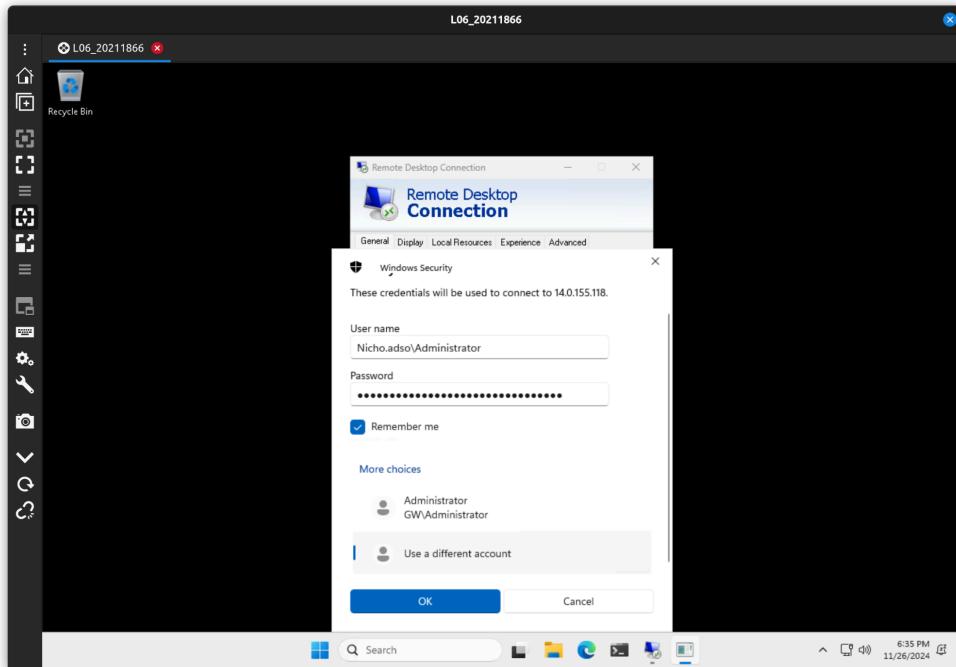


Figura 2.1. Acceso a DCSec con el usuario Nicho.adso\Administrator

Luego, desde Server Manager se elige la opción de Agregar Roles y Características para añadir el servicio de AD DS.

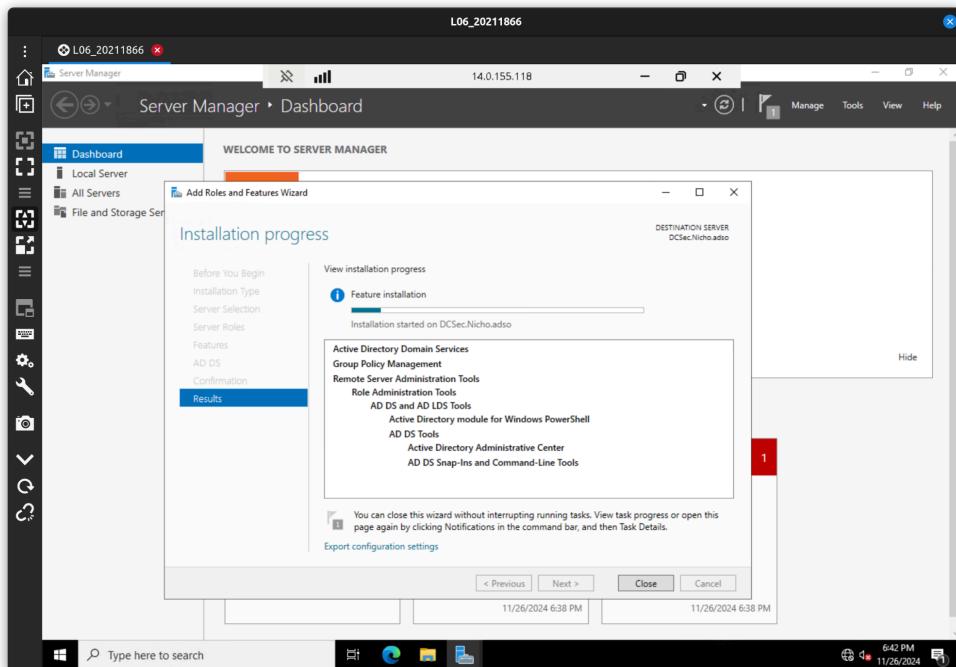


Figura 2.2. Instalación del rol de AD DS en DCSec

Luego, se realiza la promoción del servidor a Controlador de Dominio para Nicho.adso. En este caso, el dominio existe por lo que no se crea un nuevo bosque, sino que se agrega el controlador de dominio a un dominio ya existente.

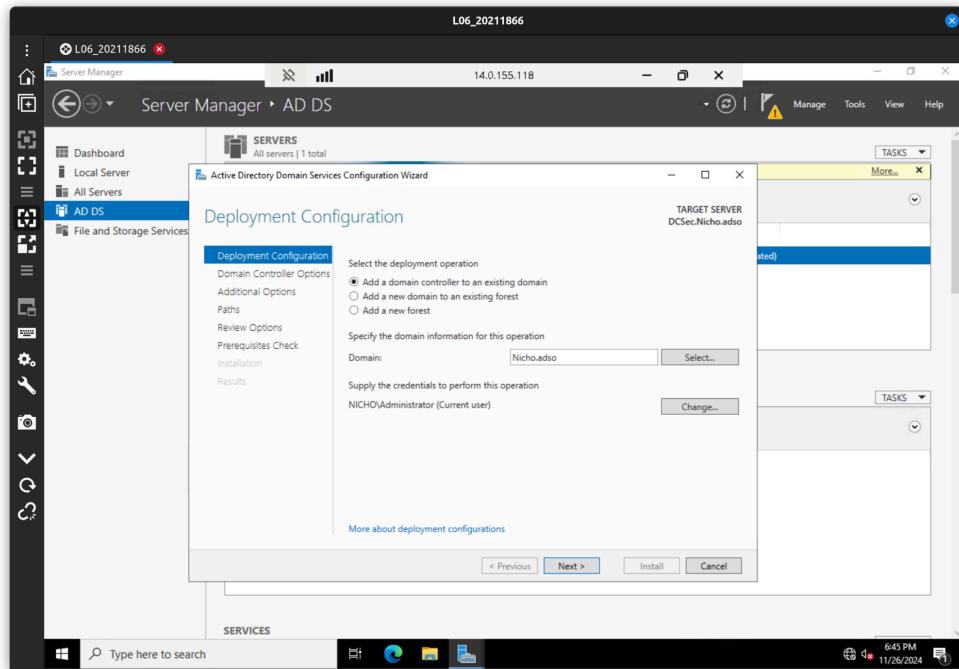


Figura 2.3. Dominio con nombre Nicho.adso

El script de PowerShell es exportado a un documento de texto.

A screenshot of a Windows Notepad window titled 'tmp8707.tmp - Notepad'. The content of the script is as follows:

```
# Windows PowerShell script for AD DS Deployment
# Import-Module ADOSSDeployment
# Install-ADDSDomainController -NoGlobalCatalog:$false -CreateDnsDelegation:$false -CriticalReplicationOnly:$false -DatabasePath "C:\Windows\NTDS" -DomainName "Nicho.adso" -InstallDns:$true -LogPath "C:\Windows\NTDS" -NoRebootOnCompletion:$false -SiteName "Default-First-Site-Name" -SysvolPath "C:\Windows\SYSVOL" -Force:$true
```

The Notepad window has a standard Windows title bar and menu bar. The code is displayed in a monospaced font. At the bottom of the window, there are buttons for 'File', 'Edit', 'Format', 'View', and 'Help'. Below the code, there are 'Previous', 'Next', 'Install', and 'Cancel' buttons.

Figura 2.4. Script de PowerShell equivalente

Finalmente, tras reiniciar se pueden observar la zona de DNS creada y el bosque.

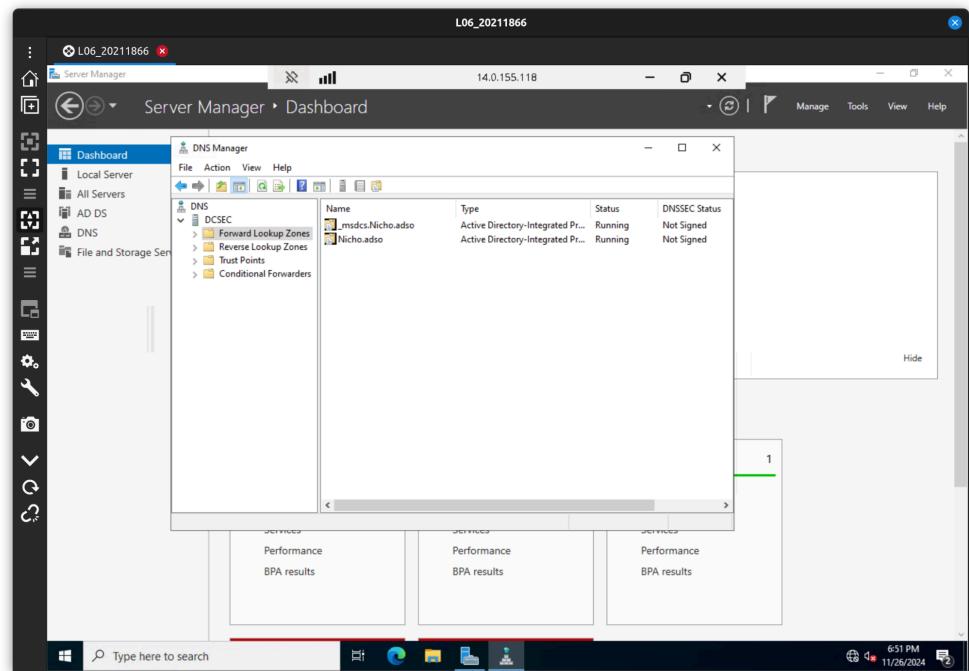


Figura 2.5. Evidencia de asignación de DCSEC como controlador del dominio

# PREGUNTA 3

En DC se ejecuta el comando indicado.

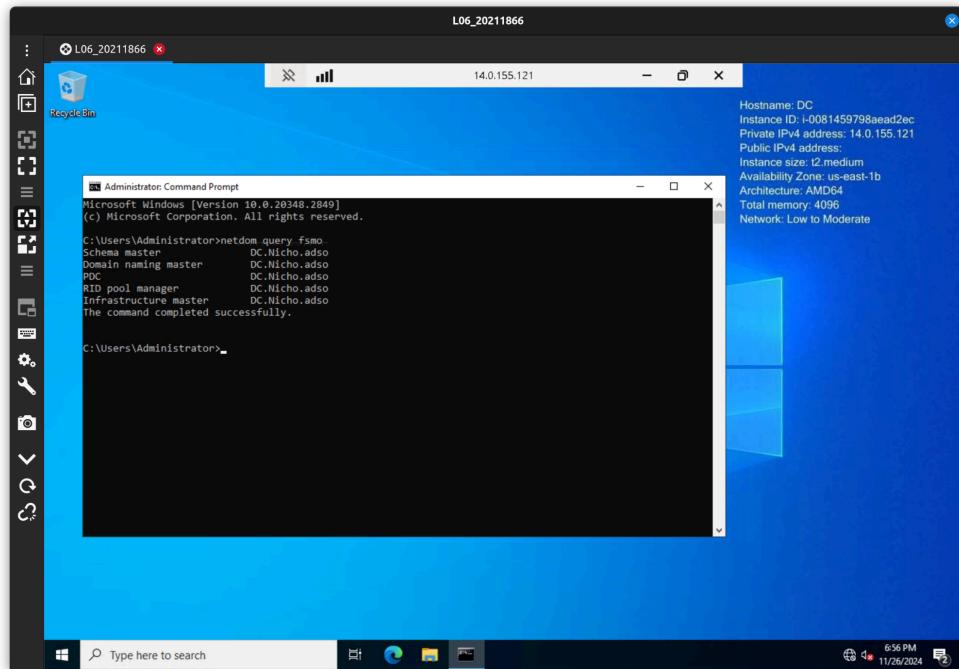


Figura 3.1. Ejecución de `netdom query fsmo` cuando DC los tiene

La figura anterior muestra los cinco roles FSMO actualmente existentes, así como el servidor que los posee, que es DC. Estos 5 roles son: Schema Master, Domain Naming Master, PDC Emulator, RID Emulator e Infrastructure Master.

En DCSec, desde Powershell se transfieren los roles. Lo que cambia en cada comando es solo el rol que se está pasando de DC a DCSec.

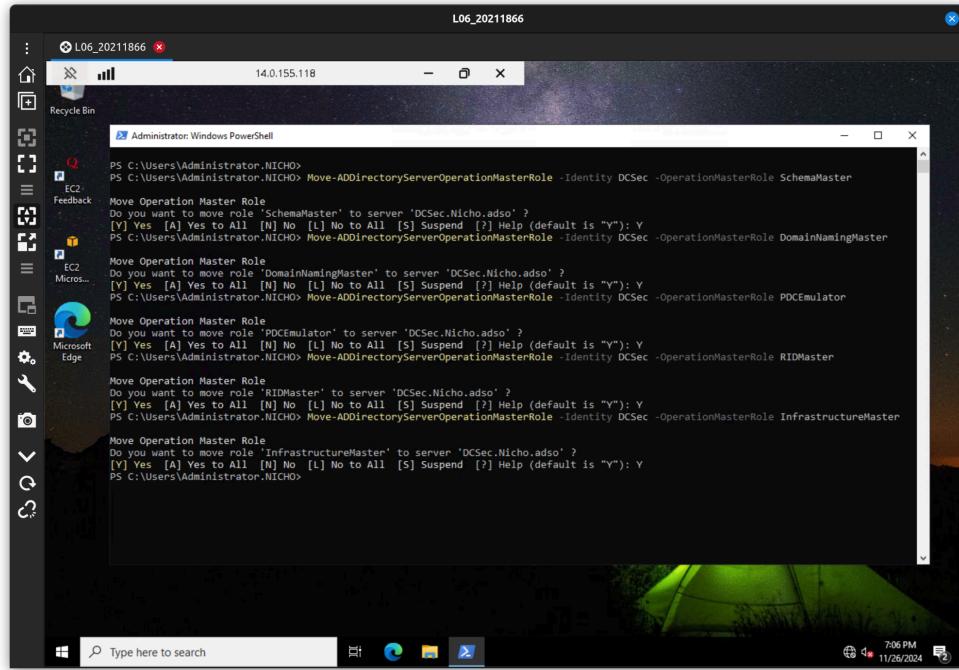


Figura 3.2. Paso de los roles FSMO a DCSec desde Powershell

También se listan los roles adquiridos con el siguiente comando.

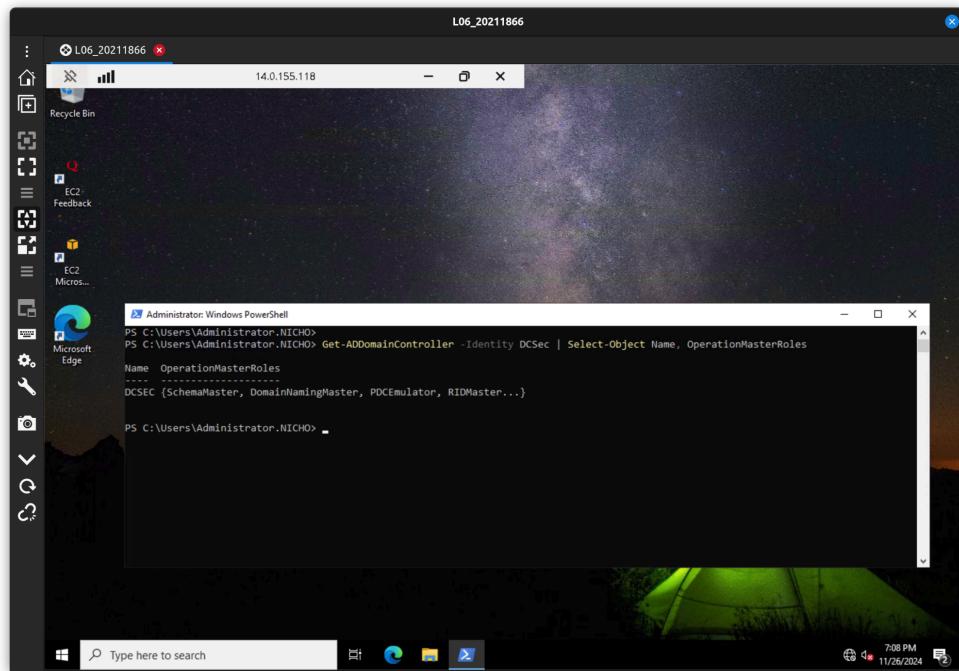


Figura 3.3. Roles FSMO adquiridos por DCSec

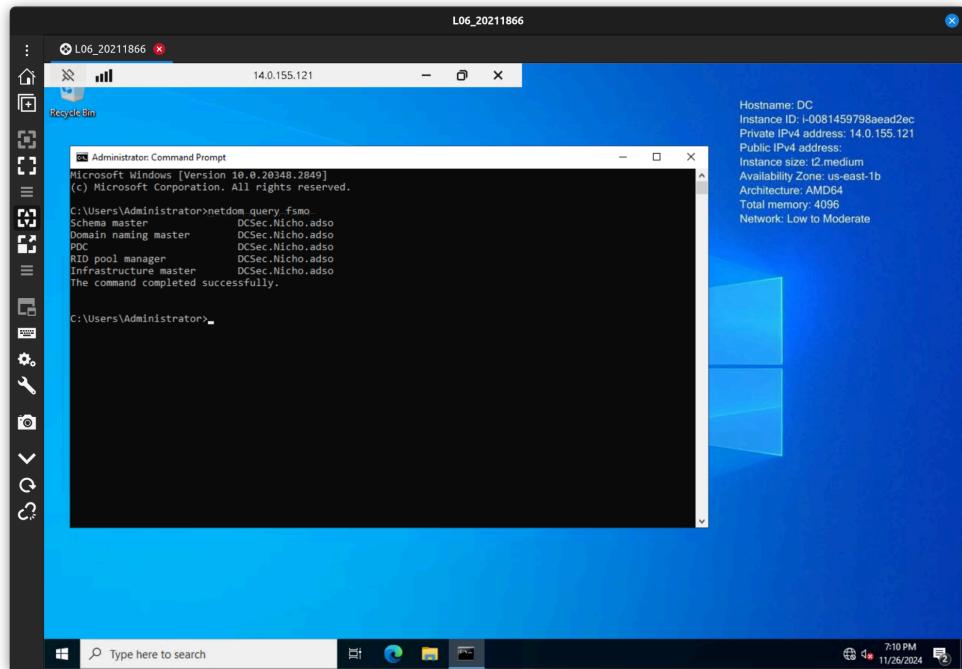


Figura 3.4. Ejecución de `netdom query fsmo` cuando DCSec los tiene

Se muestra que ahora DCSec es el propietario de los roles FSMO. Por ello, en lugar del servidor DC que aparece en la primera captura, ahora aparece DCSec.

# PREGUNTA 4

En primer lugar, se verifica la pertenencia de SERV al dominio Nicho.adso.

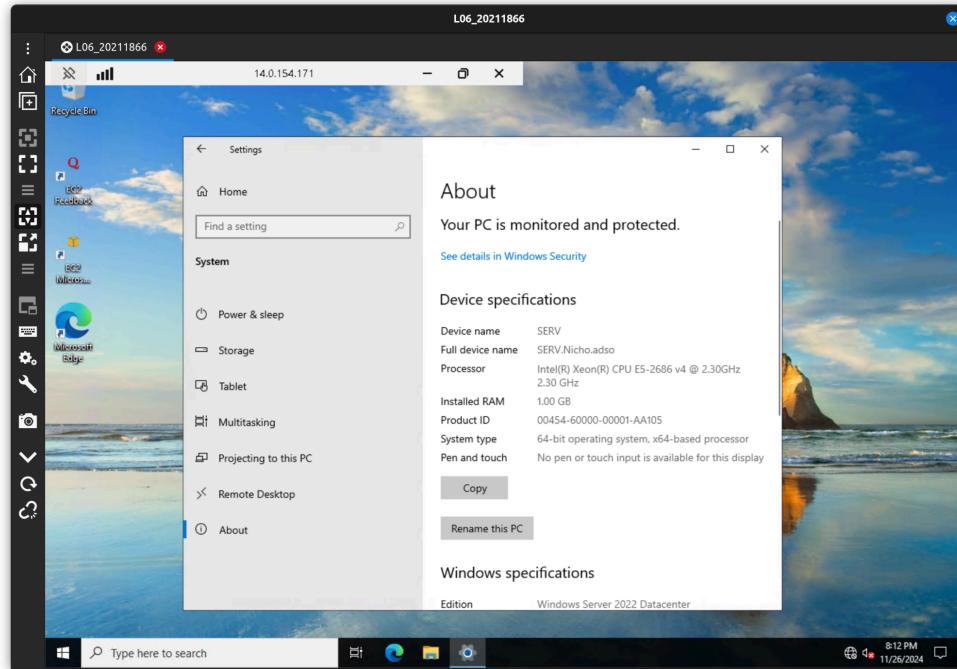


Figura 4.1. Full device name de SERV con el dominio

Luego, se otorgaron los permisos de compartición con los Domain Admins.

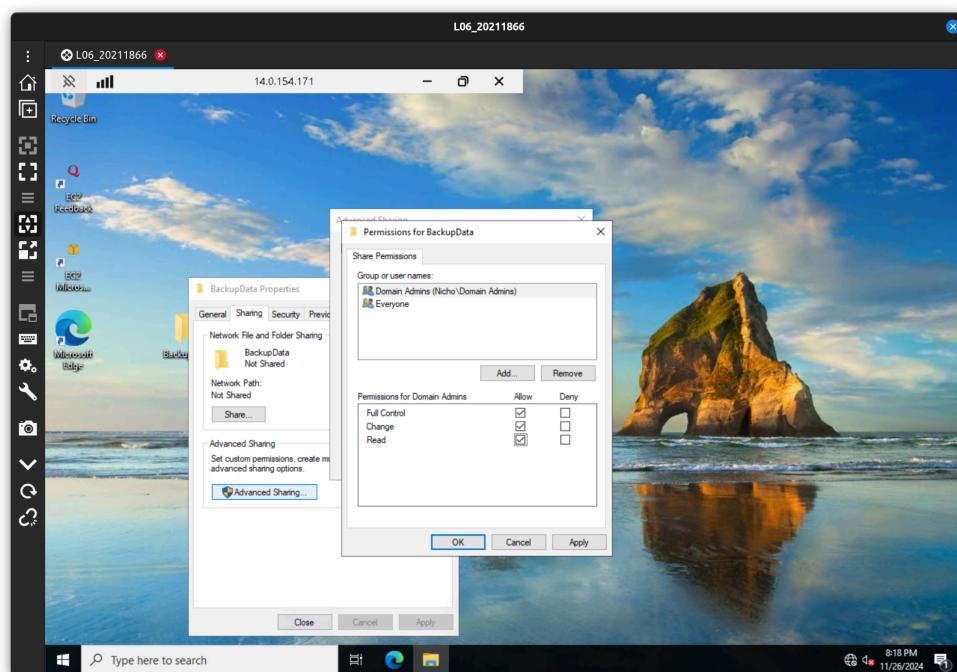


Figura 4.2. Permisos de Full Control para los administradores del dominio

En la instancia DC se crea la carpeta Data con 2 imágenes y 2 archivos de texto.

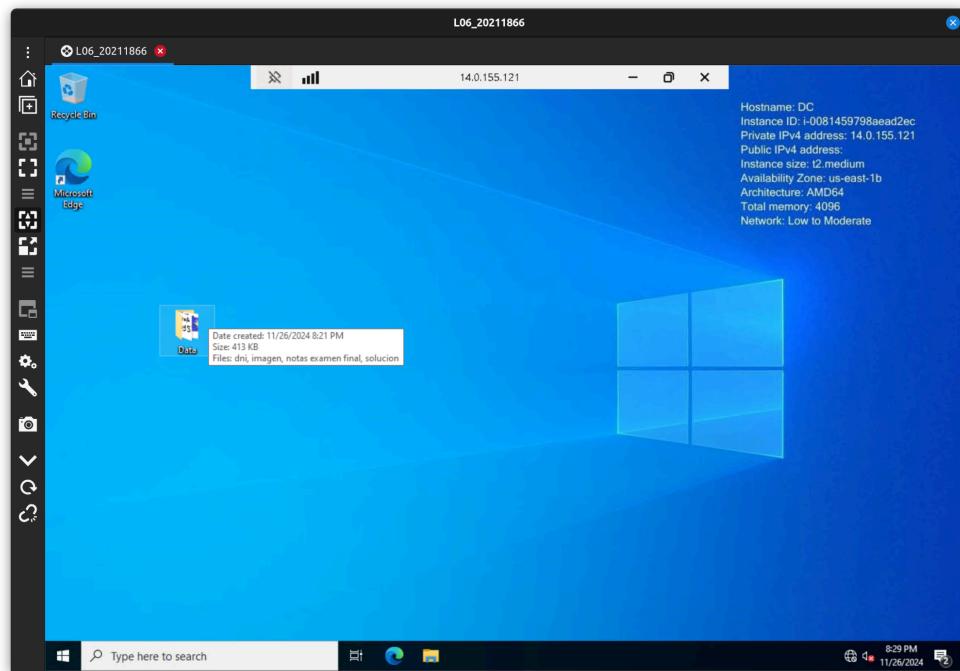


Figura 4.3. Directorio Data creado en el escritorio de DC

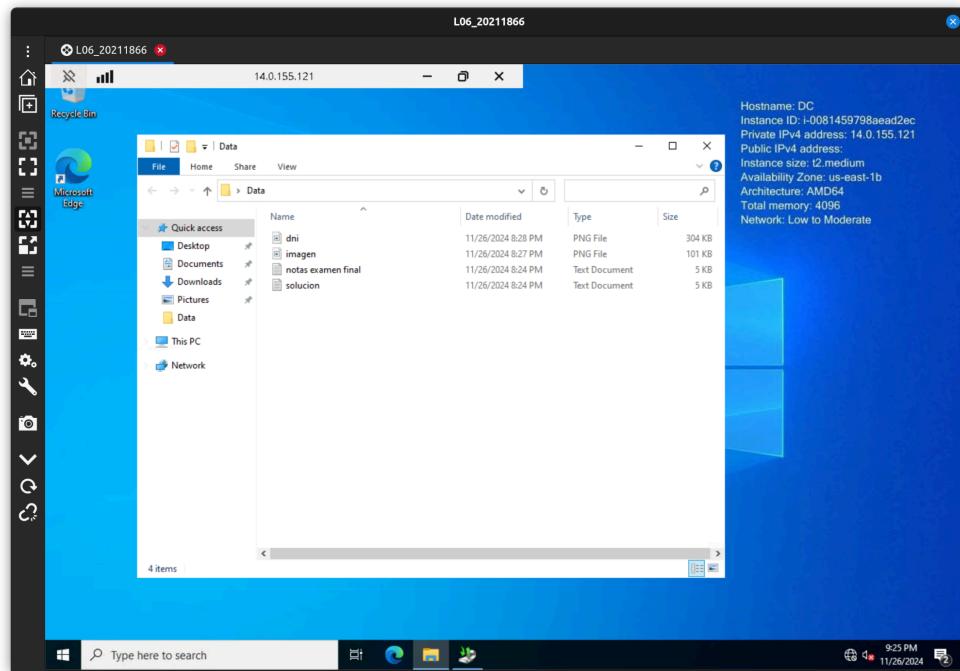


Figura 4.4. Contenido del directorio Data

Luego, se instala el servicio de Backup en el servidor desde Server Manager.

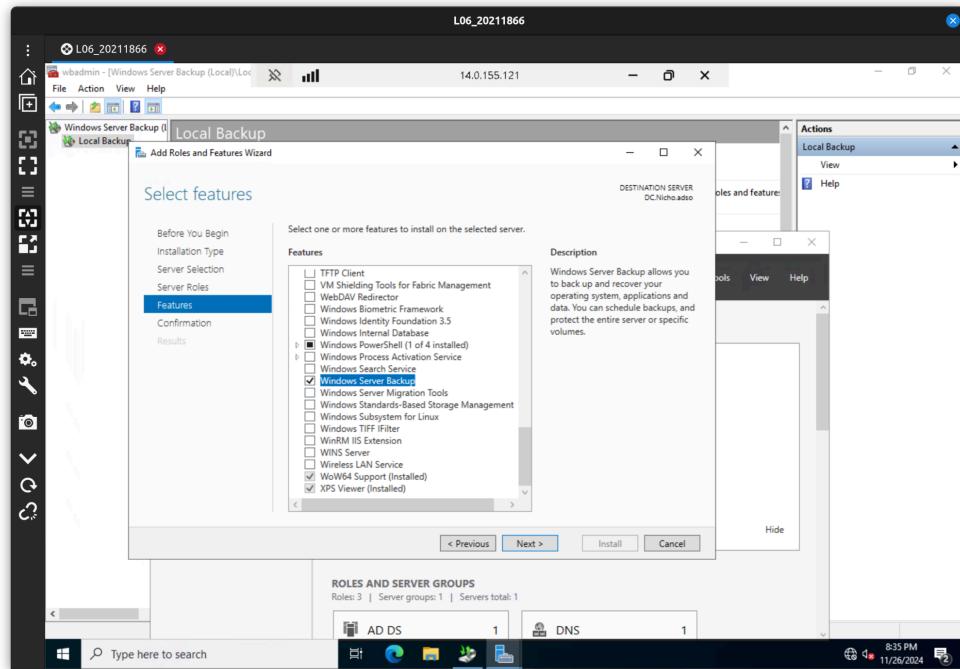


Figura 4.5. Instalación del servicio de backups

Con el servicio instalado, se procede a crear un custom backup.

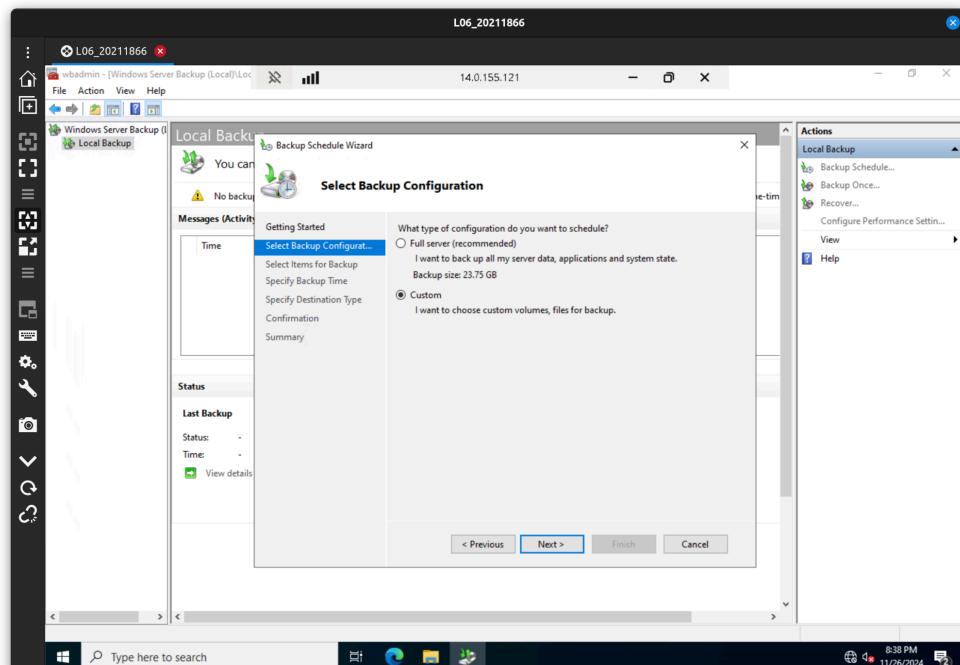


Figura 4.6. Creación de un custom backup

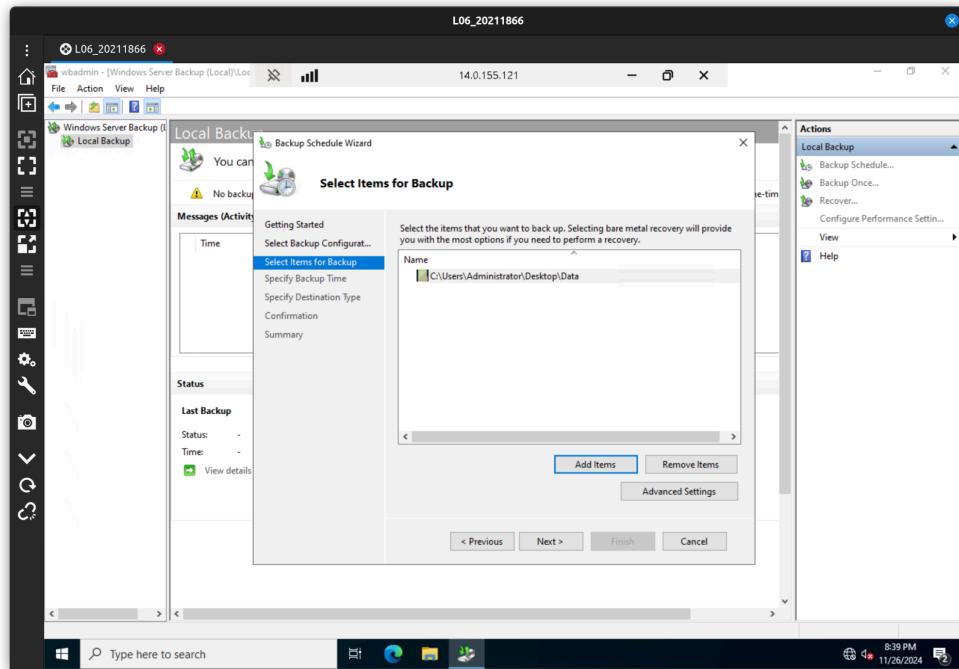


Figura 4.7. Se selecciona la carpeta Data para la copia a realizar

Además, se selecciona la opción para que el backup se realice una vez al día siempre a las 9PM.

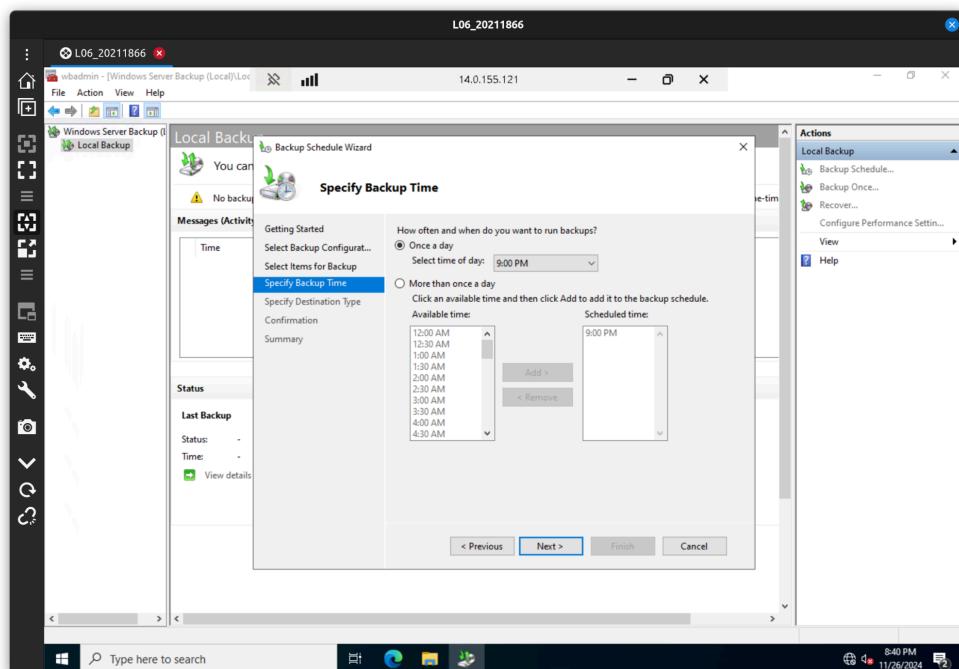


Figura 4.8. Configuración de la hora para los backups automáticos

Dado que la cantidad máxima de datos que se pueden perder en un desastre están dictaminados por el tiempo que se toma entre cada backup, el RPO está determinado por la frecuencia con que se realiza la copia de seguridad, la cual es 24 horas. Por otro lado, el RTO depende de la velocidad con que se restauren los datos, tras el desastre.

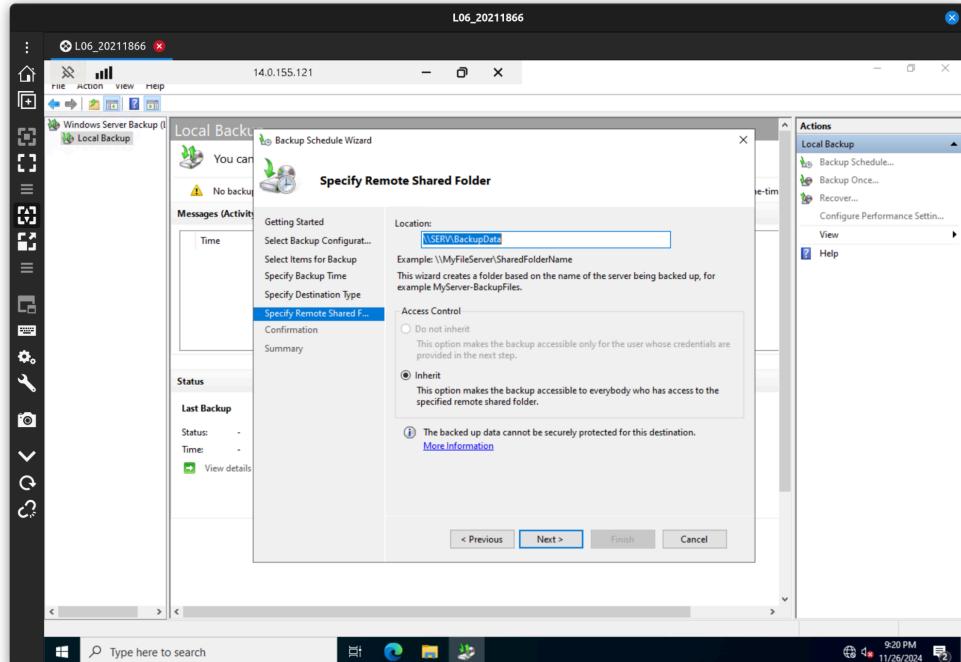


Figura 4.9. Configuración de la hora para los backups automáticos

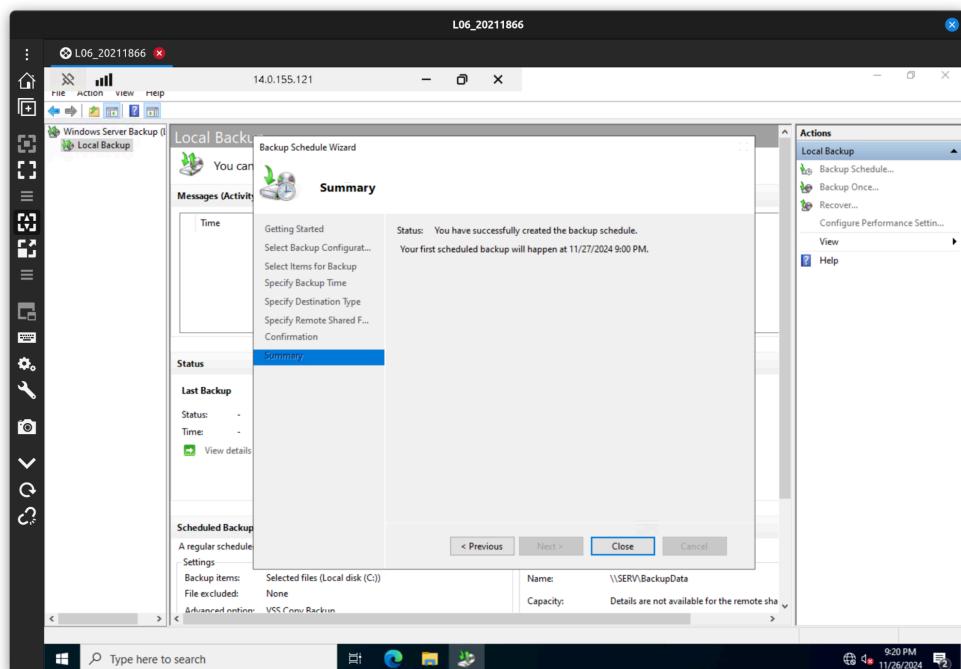


Figura 4.10. Backups automáticos configurados

Debido a que se necesita probar el servicio de backup dentro del horario de laboratorio se opta por realizar una copia de con la opción Backup Once.

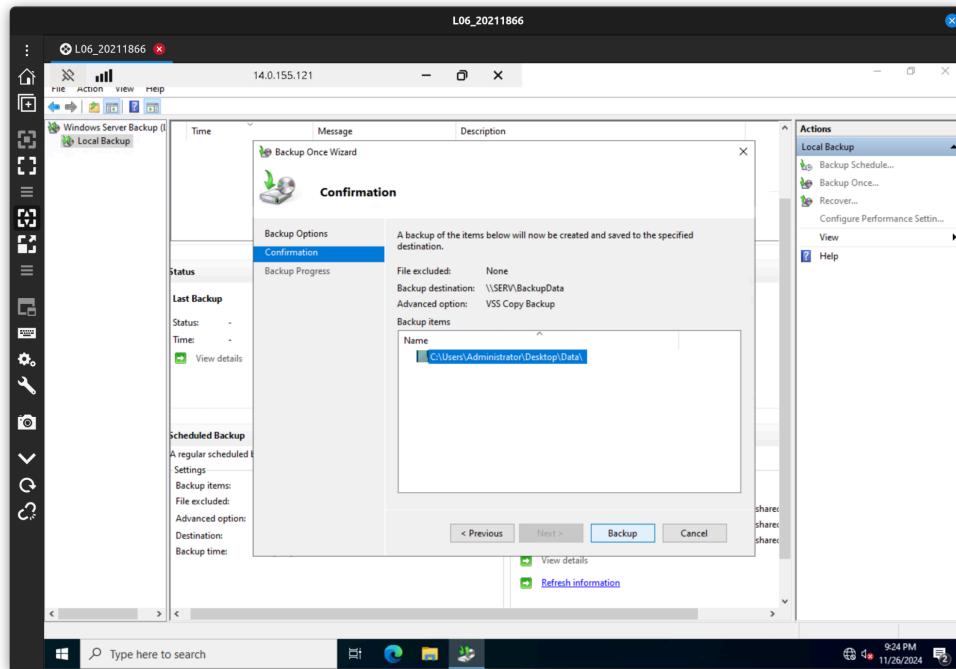


Figura 4.11. Creación de un Backup de forma manual

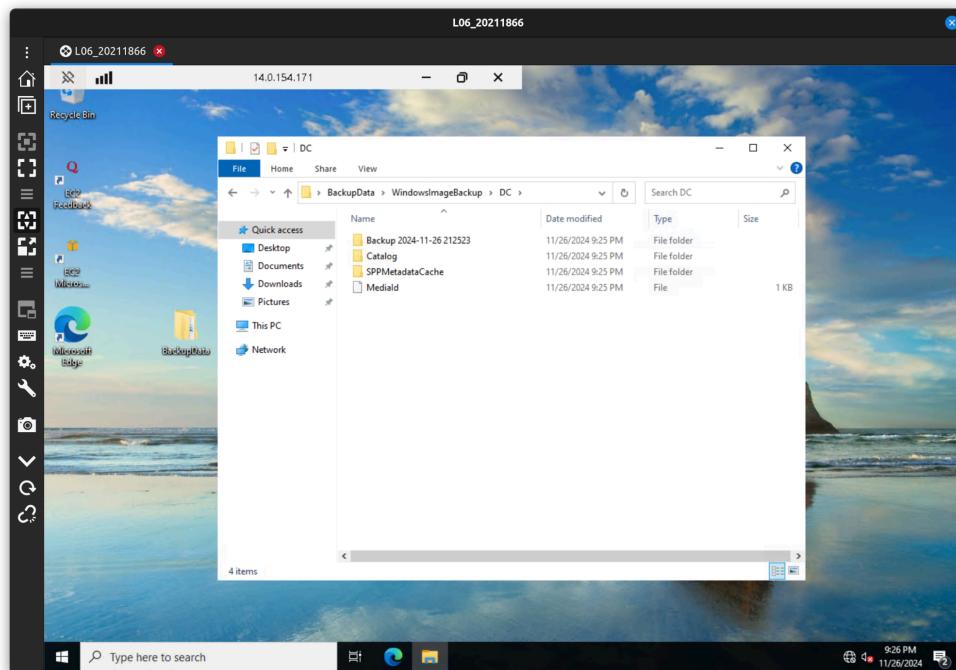


Figura 4.12. Resultado del backup creado en la instancia SERV

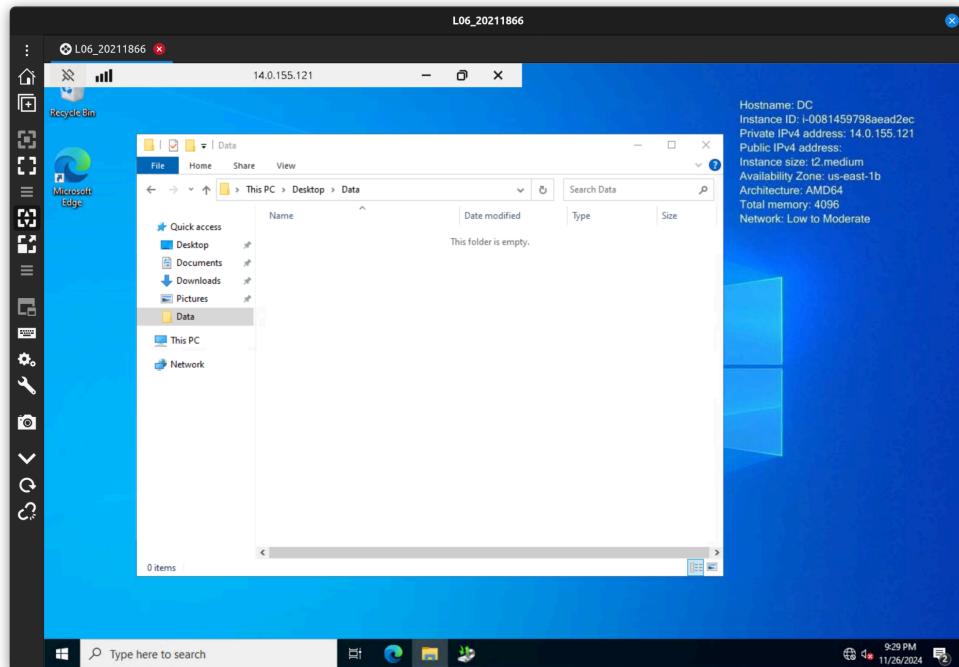


Figura 4.13. Se borra el contenido de Data para simular un evento no deseado

Se usa la opción de Recover.

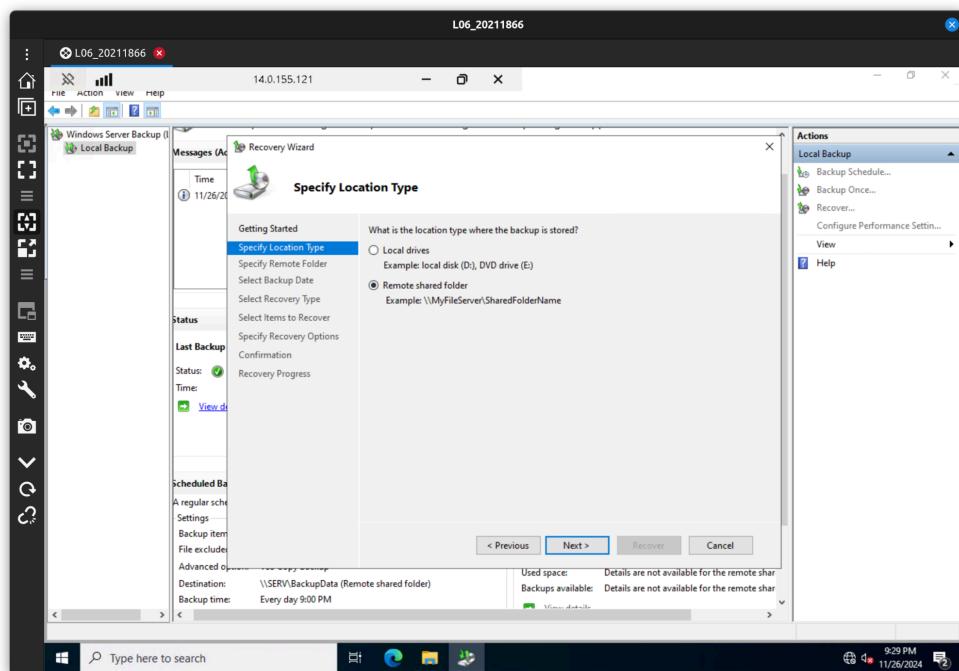


Figura 4.14. Se selecciona la opción de recuperar directorio remoto

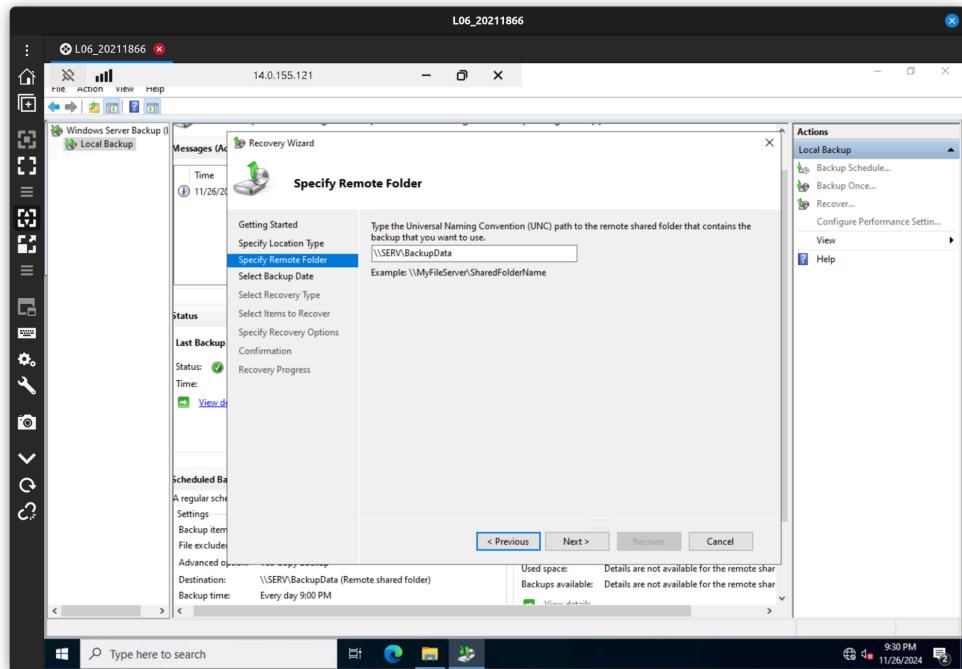


Figura 4.15. Se indica la dirección del directorio a recuperar

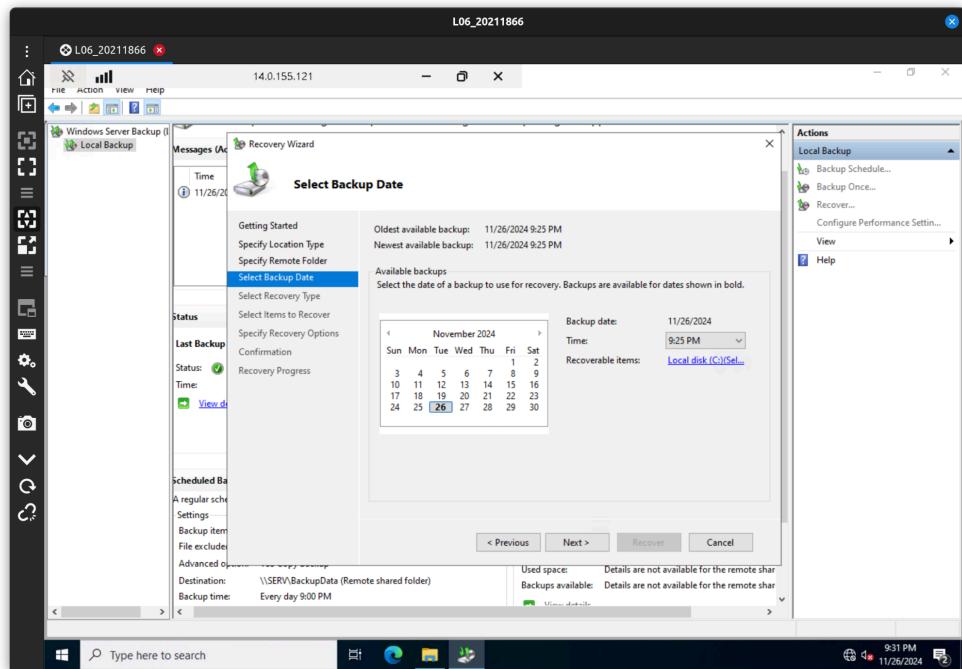


Figura 4.16. Se indica la fecha del backup realizado

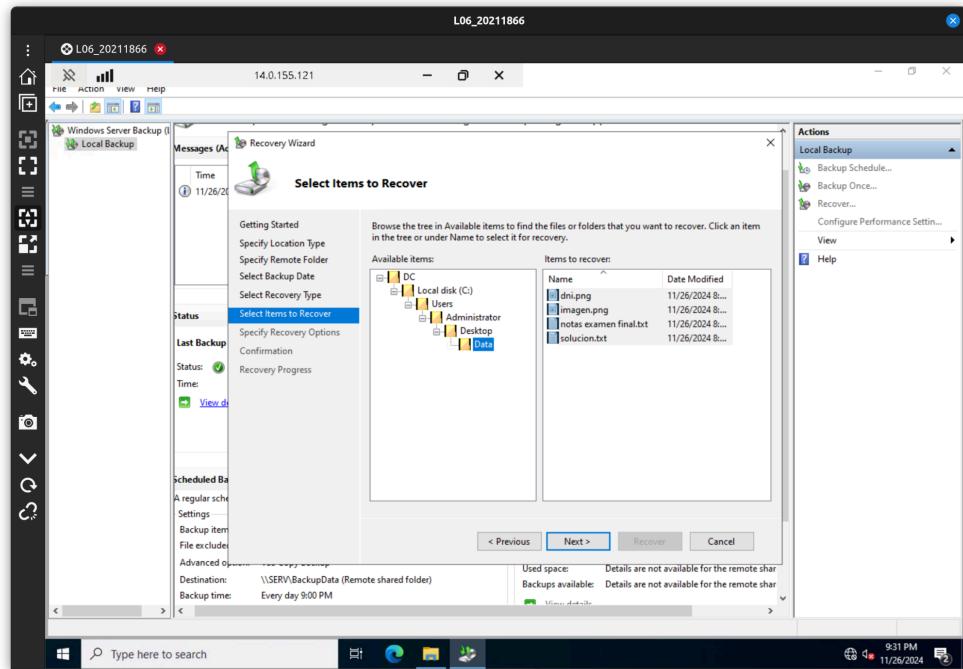


Figura 4.17. Se seleccionan los contenidos a recuperar

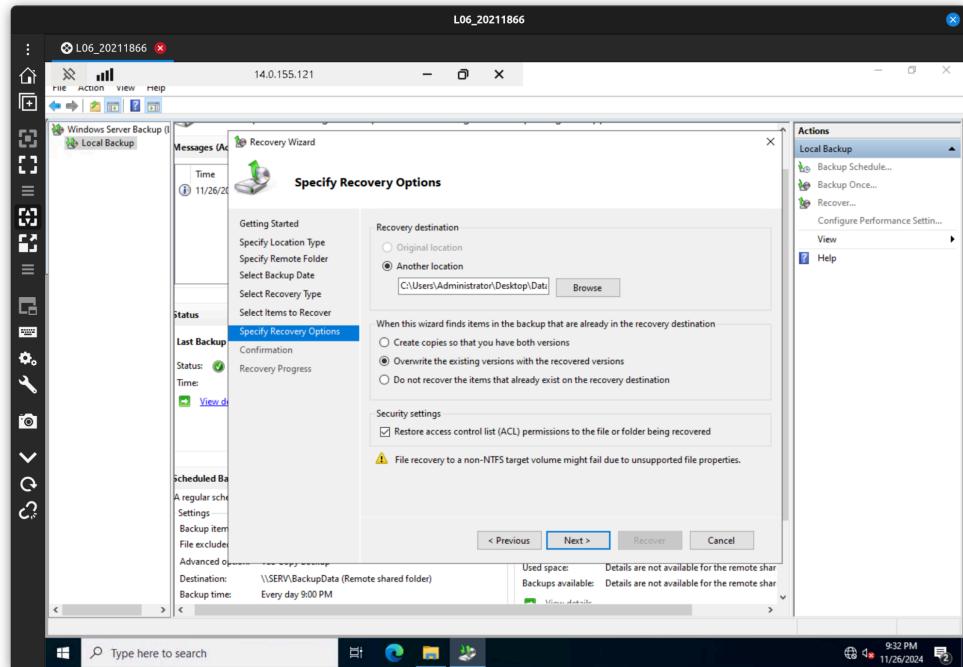


Figura 4.18. Se indica que los contenidos deben volver a la carpeta Data

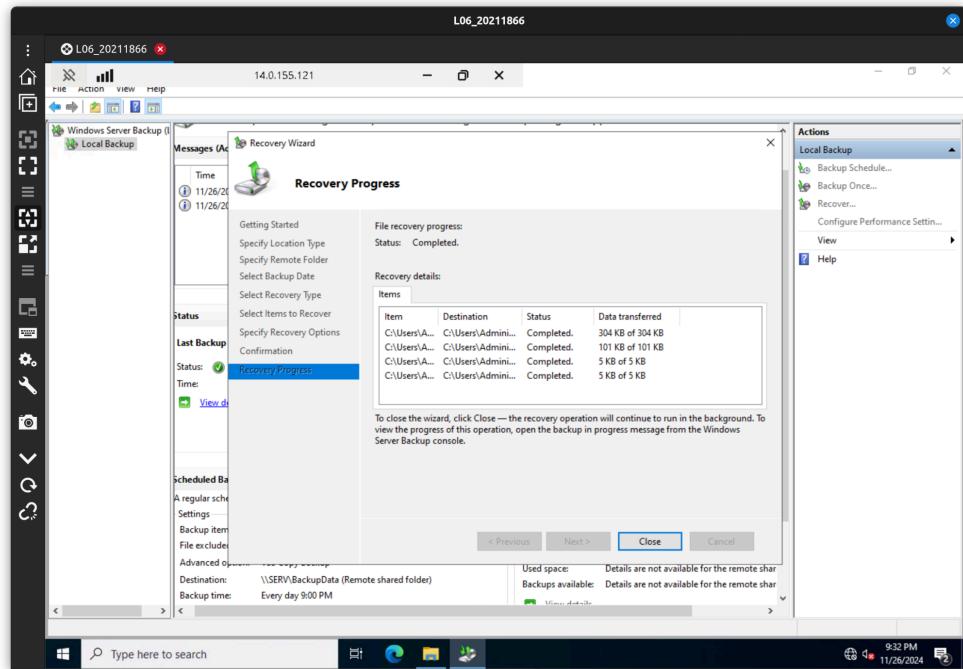


Figura 4.19. Resultado de la operación de recuperación

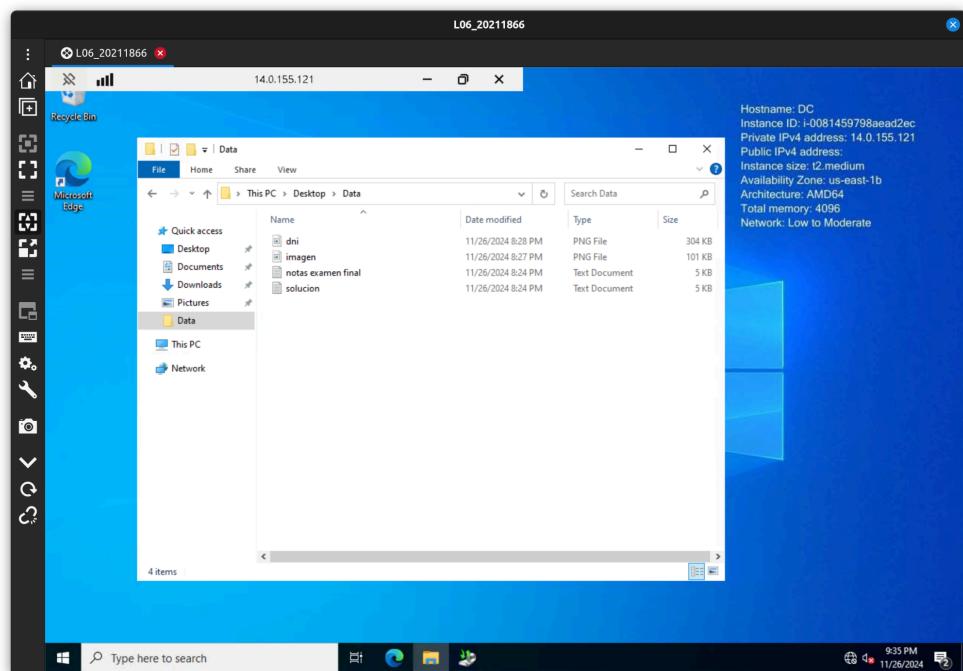


Figura 4.20. Archivos recuperados

# PREGUNTA 5

Se crea la instancia Ubuntu según las especificaciones indicadas.

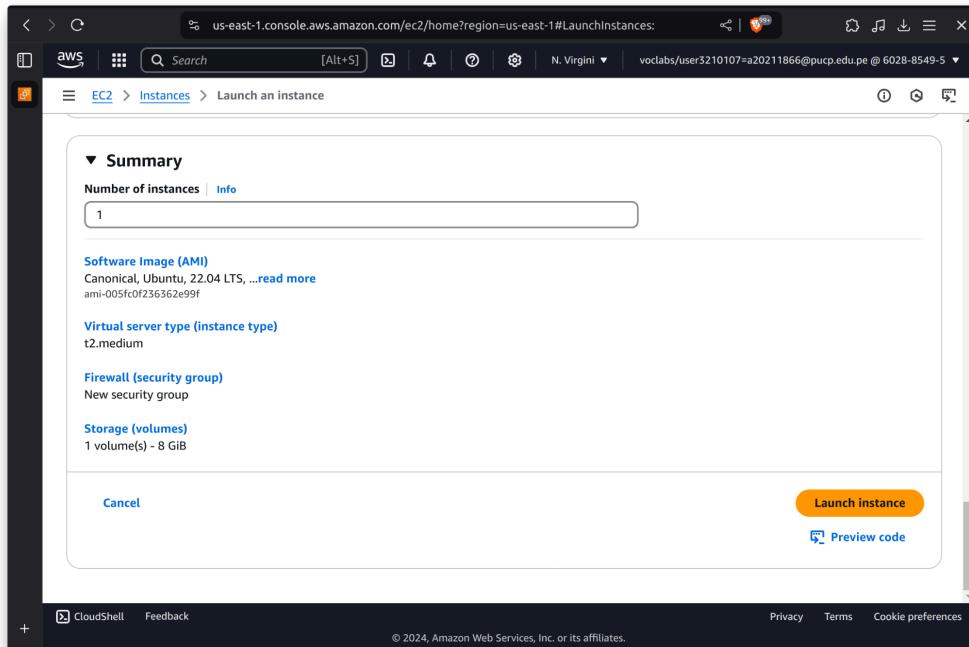


Figura 5.1. Resumen de creación de la instancia Ubuntu

```
ubuntu@ip-172-31-92-118: ~
System load:  0.41      Processes:           120
Usage of /:   22.9% of 6.71GB  Users logged in:       0
Memory usage: 5%
Swap usage:   0%
Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

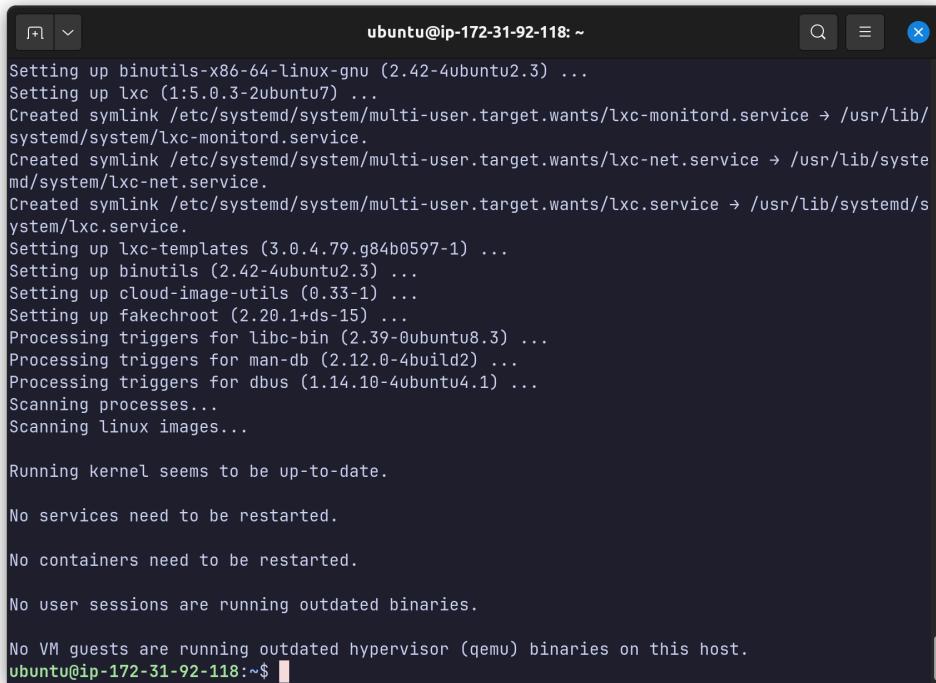
The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

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.

ubuntu@ip-172-31-92-118:~$
```

Figura 5.2. Conexión por ssh a la instancia Ubuntu



```

ubuntu@ip-172-31-92-118: ~
Setting up binutils-x86_64-linux-gnu (2.42-4ubuntu2.3) ...
Setting up lxc (1:5.0.3-2ubuntu7) ...
Created symlink /etc/systemd/system/multi-user.target.wants/lxc-monitor.service → /usr/lib/
systemd/system/lxc-monitor.service.
Created symlink /etc/systemd/system/multi-user.target.wants/lxc-net.service → /usr/lib/syste
md/system/lxc-net.service.
Created symlink /etc/systemd/system/multi-user.target.wants/lxc.service → /usr/lib/systemd/s
ystem/lxc.service.
Setting up lxc-templates (3.0.4.79.g84b0597-1) ...
Setting up binutils (2.42-4ubuntu2.3) ...
Setting up cloud-image-utils (0.33-1) ...
Setting up fakechroot (2.20.1+ds-15) ...
Processing triggers for libc-bin (2.39-0ubuntu8.3) ...
Processing triggers for man-db (2.12.0-4build2) ...
Processing triggers for dbus (1.14.10-4ubuntu4.1) ...
Scanning processes...
Scanning linux images...

Running kernel seems to be up-to-date.

No services need to be restarted.

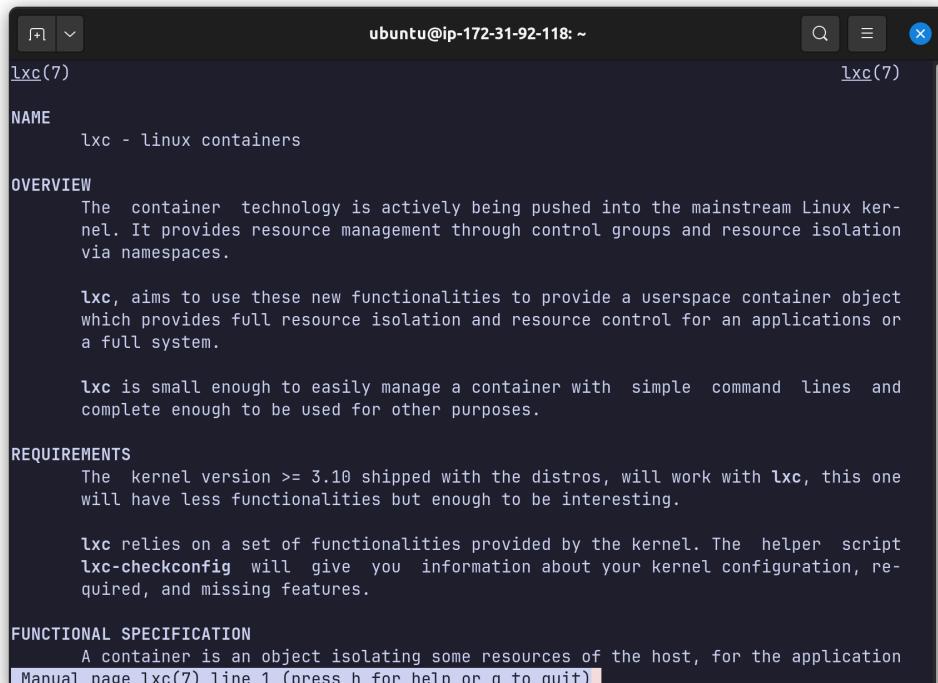
No containers need to be restarted.

No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.
ubuntu@ip-172-31-92-118:~$ █

```

Figura 5.3. Resultado de ejecución del comando `sudo apt install lxc`



```

ubuntu@ip-172-31-92-118: ~
lxc(7)
NAME
    lxc - linux containers
OVERVIEW
    The container technology is actively being pushed into the mainstream Linux kernel. It provides resource management through control groups and resource isolation via namespaces.

    lxc, aims to use these new functionalities to provide a userspace container object which provides full resource isolation and resource control for an applications or a full system.

    lxc is small enough to easily manage a container with simple command lines and complete enough to be used for other purposes.

REQUIREMENTS
    The kernel version >= 3.10 shipped with the distros, will work with lxc, this one will have less functionalities but enough to be interesting.

    lxc relies on a set of functionalities provided by the kernel. The helper script lxc-checkconfig will give you information about your kernel configuration, required, and missing features.

FUNCTIONAL SPECIFICATION
    A container is an object isolating some resources of the host, for the application
    Manual page lxc(7) line 1 (press h for help or q to quit).

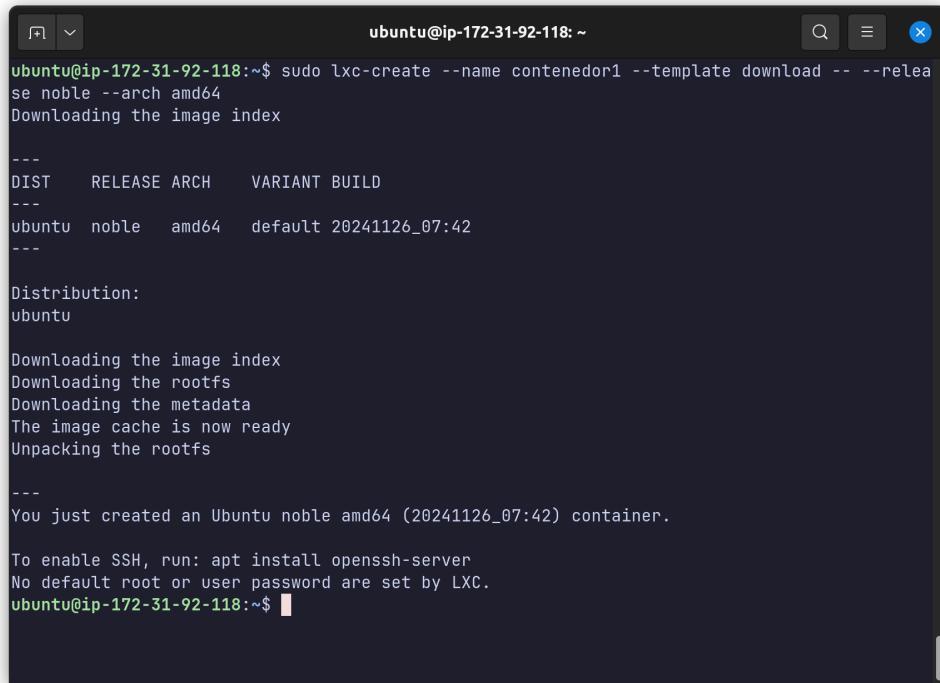
```

Figura 5.4. Manual de uso del comando `lxc`

Para la creación de los contenedores se usa el comando:

`sudo lxc-create --name contenedor1 --template download -- --release noble --arch amd64`

Lo único que cambia es el nombre del contenedor. Se usa download como template para no tener que instalar con apt todos los templates, sino en este caso solo instalar el de ubuntu.



```
ubuntu@ip-172-31-92-118:~$ sudo lxc-create --name contenedor1 --template download -- --release noble --arch amd64
Downloading the image index

---
DIST      RELEASE ARCH      VARIANT BUILD
---
ubuntu    noble   amd64    default 20241126_07:42
---

Distribution:
ubuntu

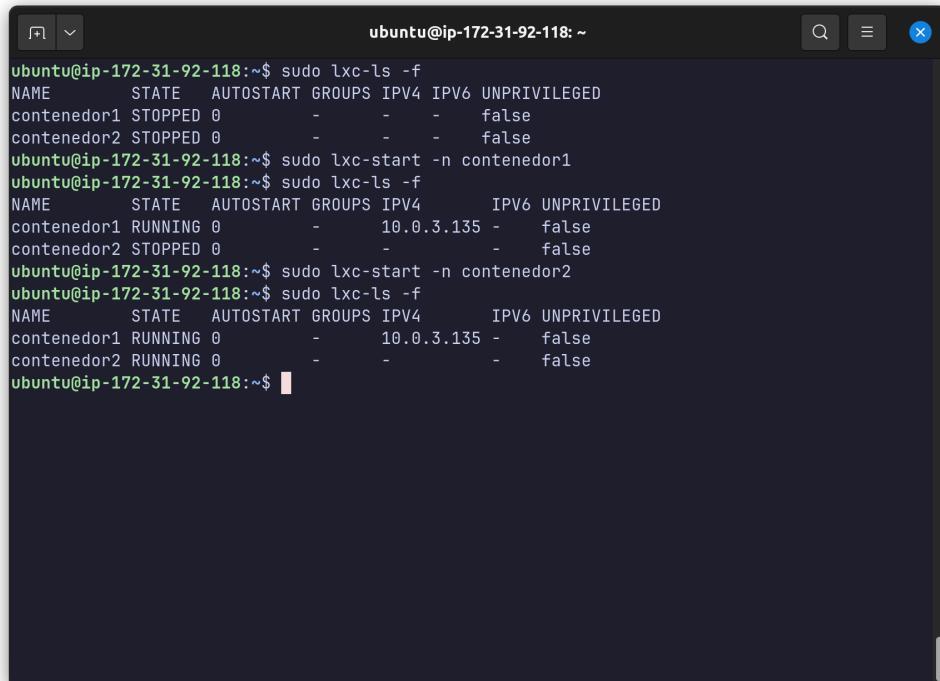
Downloading the image index
Downloading the rootfs
Downloading the metadata
The image cache is now ready
Unpacking the rootfs

---
You just created an Ubuntu noble amd64 (20241126_07:42) container.

To enable SSH, run: apt install openssh-server
No default root or user password are set by LXC.
ubuntu@ip-172-31-92-118:~$
```

Figura 5.5. Creación de un contenedor

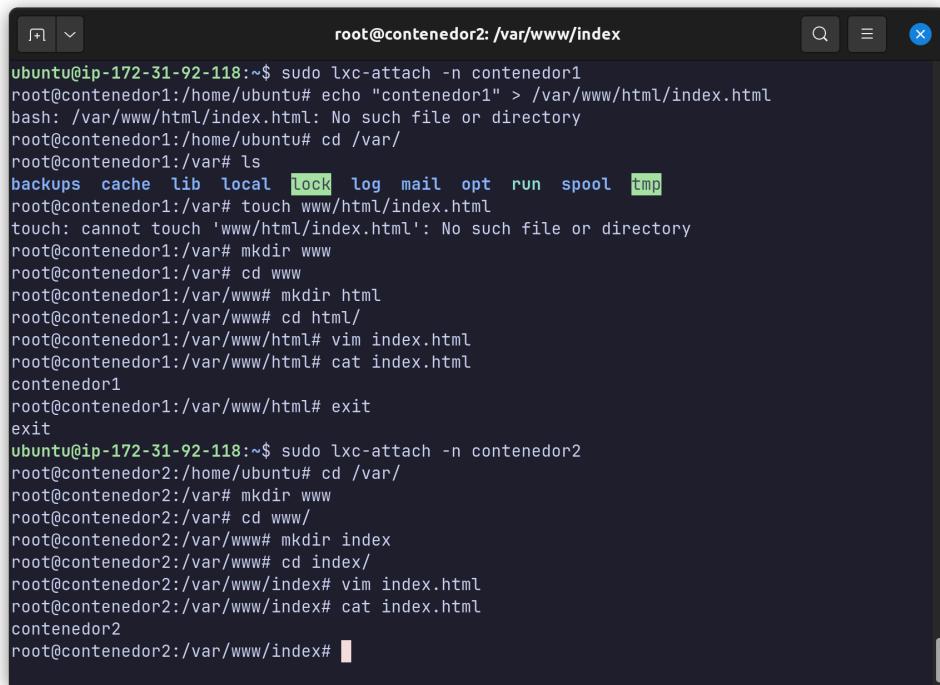
Se realiza esto para cada contenedor. A continuación, se listan los contenedores creados.



```
ubuntu@ip-172-31-92-118:~$ sudo lxc-ls -f
NAME      STATE  AUTOSTART GROUPS IPV4 IPV6 UNPRIVILEGED
contenedor1 STOPPED 0      -      -      -      false
contenedor2 STOPPED 0      -      -      -      false
ubuntu@ip-172-31-92-118:~$ sudo lxc-start -n contenedor1
ubuntu@ip-172-31-92-118:~$ sudo lxc-ls -f
NAME      STATE  AUTOSTART GROUPS IPV4      IPV6 UNPRIVILEGED
contenedor1 RUNNING 0      -      10.0.3.135 -      false
contenedor2 STOPPED 0      -      -      -      false
ubuntu@ip-172-31-92-118:~$ sudo lxc-start -n contenedor2
ubuntu@ip-172-31-92-118:~$ sudo lxc-ls -f
NAME      STATE  AUTOSTART GROUPS IPV4      IPV6 UNPRIVILEGED
contenedor1 RUNNING 0      -      10.0.3.135 -      false
contenedor2 RUNNING 0      -      -      -      false
ubuntu@ip-172-31-92-118:~$
```

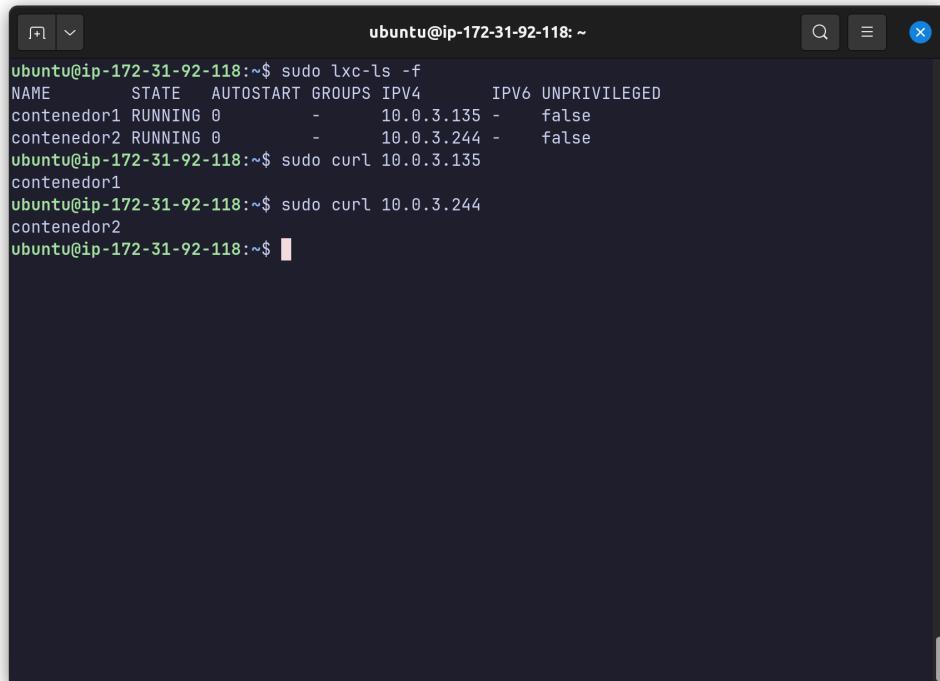
Figura 5.6. Listado de contenedores creados

Luego se instala el paquete de apache2 y se escriben los mensajes en los index.html, de manera respectiva dentro de cada contenedor.



```
root@contenedor2: /var/www/index
ubuntu@ip-172-31-92-118:~$ sudo lxc-attach -n contenedor1
root@contenedor1:/home/ubuntu# echo "contenedor1" > /var/www/html/index.html
bash: /var/www/html/index.html: No such file or directory
root@contenedor1:/home/ubuntu# cd /var/
root@contenedor1:/var# ls
backups cache lib local lock log mail opt run spool tmp
root@contenedor1:/var# touch www/html/index.html
touch: cannot touch 'www/html/index.html': No such file or directory
root@contenedor1:/var# mkdir www
root@contenedor1:/var# cd www
root@contenedor1:/var/www# vim index.html
root@contenedor1:/var/www/html# cat index.html
contenedor1
root@contenedor1:/var/www/html# exit
exit
ubuntu@ip-172-31-92-118:~$ sudo lxc-attach -n contenedor2
root@contenedor2:/home/ubuntu# cd /var/
root@contenedor2:/var# mkdir www
root@contenedor2:/var# cd www/
root@contenedor2:/var/www# mkdir index
root@contenedor2:/var/www# cd index/
root@contenedor2:/var/www/index# vim index.html
root@contenedor2:/var/www/index# cat index.html
contenedor2
root@contenedor2:/var/www/index#
```

Figura 5.7. Configuración de los archivos index.html para cada contenedor



```
ubuntu@ip-172-31-92-118:~$ sudo lxc-ls -f
NAME      STATE  AUTOSTART GROUPS IPV4          IPV6 UNPRIVILEGED
contenedor1 RUNNING 0      -      10.0.3.135 -    false
contenedor2 RUNNING 0      -      10.0.3.244 -    false
ubuntu@ip-172-31-92-118:~$ sudo curl 10.0.3.135
contenedor1
ubuntu@ip-172-31-92-118:~$ sudo curl 10.0.3.244
contenedor2
ubuntu@ip-172-31-92-118:~$
```

Figura 5.8. Conexión a cada página web con curl

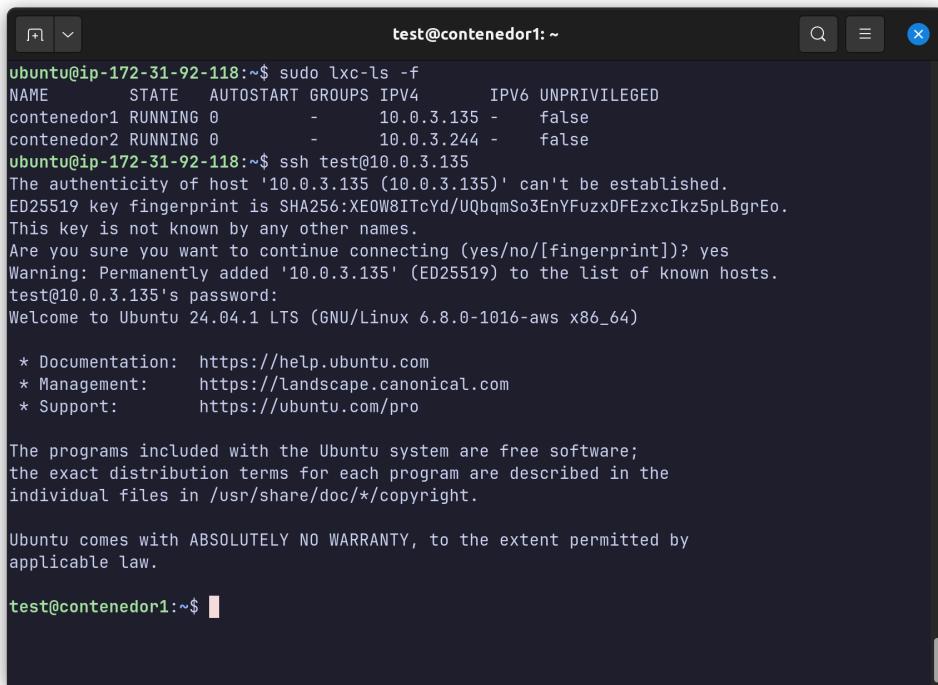
Tras esto, se crean los usuarios test dentro de cada contenedor y se instala el paquete de openssh con apt.

```
Creating SSH2 ED25519 key; this may take some time ...
256 SHA256:XE0W8ITcYd/UQbgmSo3EnYFuzxDFEzxcIkz5pLBgrEo root@contenedor1 (ED25519)
Created symlink /etc/systemd/system/sockets.target.wants/ssh.socket → /usr/lib/systemd/system/ssh.socket.
Created symlink /etc/systemd/system/ssh.service.requires/ssh.socket → /usr/lib/systemd/system/ssh.socket.
Processing triggers for libc-bin (2.39-0ubuntu8.3) ...
root@contenedor1:/home/ubuntu# adduser test
info: Adding user `test' ...
info: Selecting UID/GID from range 1000 to 59999 ...
info: Adding new group `test' (1001) ...
info: Adding new user `test' (1001) with group `test (1001)' ...
info: Creating home directory `/home/test' ...
info: Copying files from `/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for test
Enter the new value, or press ENTER for the default
      Full Name []: test
      Room Number []:
      Work Phone []:
      Home Phone []:
      Other []:
Is the information correct? [Y/n]
info: Adding new user `test' to supplemental / extra groups `users' ...
info: Adding user `test' to group `users' ...
root@contenedor1:/home/ubuntu#
```

Figura 5.9. Instalación de openssh-server en el contenedor1

```
256 SHA256:tp+99yWMbXUnb2PS6ApaK3/4bilLqE1BuoufNdQEXl98 root@contenedor2 (ED25519)
Created symlink /etc/systemd/system/sockets.target.wants/ssh.socket → /usr/lib/systemd/system/ssh.socket.
Created symlink /etc/systemd/system/ssh.service.requires/ssh.socket → /usr/lib/systemd/system/ssh.socket.
Processing triggers for libc-bin (2.39-0ubuntu8.3) ...
root@contenedor2:/home/ubuntu# adduser test
info: Adding user `test' ...
info: Selecting UID/GID from range 1000 to 59999 ...
info: Adding new group `test' (1001) ...
info: Adding new user `test' (1001) with group `test (1001)' ...
info: Creating home directory `/home/test' ...
info: Copying files from `/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for test
Enter the new value, or press ENTER for the default
      Full Name []: test
      Room Number []:
      Work Phone []:
      Home Phone []:
      Other []
Is the information correct? [Y/n]
info: Adding new user `test' to supplemental / extra groups `users' ...
info: Adding user `test' to group `users' ...
root@contenedor2:/home/ubuntu# systemctl restart ssh
root@contenedor2:/home/ubuntu#
```

Figura 5.10. Instalación de openssh-server en el contenedor2



```
test@contenedor1:~$ sudo lxc-ls -f
NAME      STATE  AUTOSTART GROUPS IPV4      IPV6 UNPRIVILEGED
contenedor1 RUNNING 0      -      10.0.3.135 -  false
contenedor2 RUNNING 0      -      10.0.3.244 -  false
ubuntu@ip-172-31-92-118:~$ ssh test@10.0.3.135
The authenticity of host '10.0.3.135 (10.0.3.135)' can't be established.
ED25519 key fingerprint is SHA256:XE0W8ITcYd/UQbqmSo3EnYFuzxDFEzxcIkz5pLBgrEo.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.0.3.135' (ED25519) to the list of known hosts.
test@10.0.3.135's password:
Welcome to Ubuntu 24.04.1 LTS (GNU/Linux 6.8.0-1016-aws x86_64)

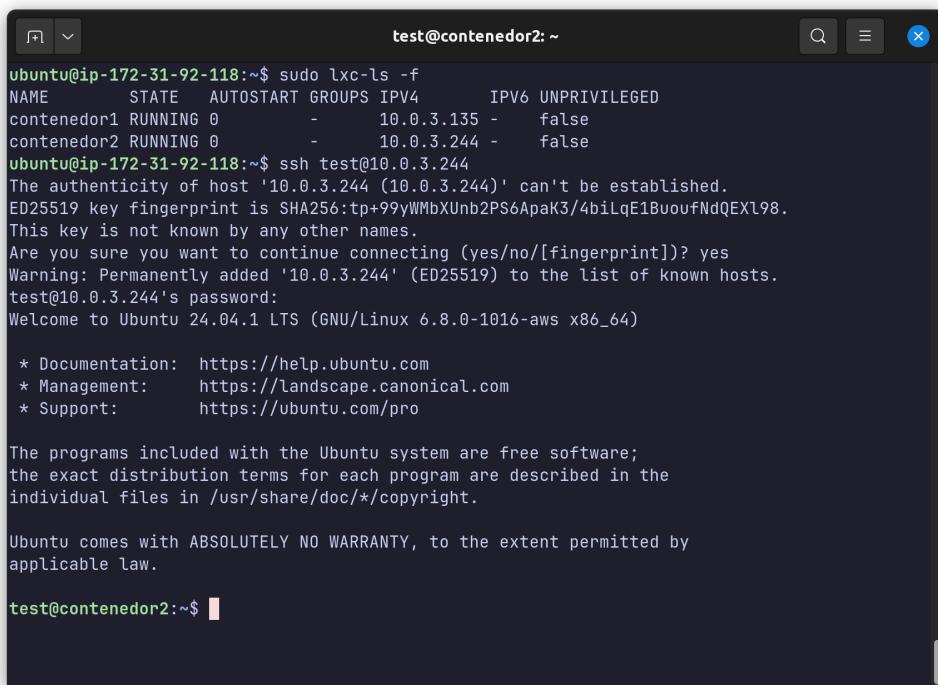
 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/pro

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.

test@contenedor1:~$
```

Figura 5.11. Conexión al contenedor1 con el usuario test



```
test@contenedor2:~$ sudo lxc-ls -f
NAME      STATE  AUTOSTART GROUPS IPV4      IPV6 UNPRIVILEGED
contenedor1 RUNNING 0      -      10.0.3.135 -  false
contenedor2 RUNNING 0      -      10.0.3.244 -  false
ubuntu@ip-172-31-92-118:~$ ssh test@10.0.3.244
The authenticity of host '10.0.3.244 (10.0.3.244)' can't be established.
ED25519 key fingerprint is SHA256:tp+99yWMbXUnb2PS6ApaK3/4biLqE1BuoufNdQEXl98.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.0.3.244' (ED25519) to the list of known hosts.
test@10.0.3.244's password:
Welcome to Ubuntu 24.04.1 LTS (GNU/Linux 6.8.0-1016-aws x86_64)

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

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.

test@contenedor2:~$
```

Figura 5.12. Conexión al contenedor2 con el usuario test