PRÁCTICA 5 SSH

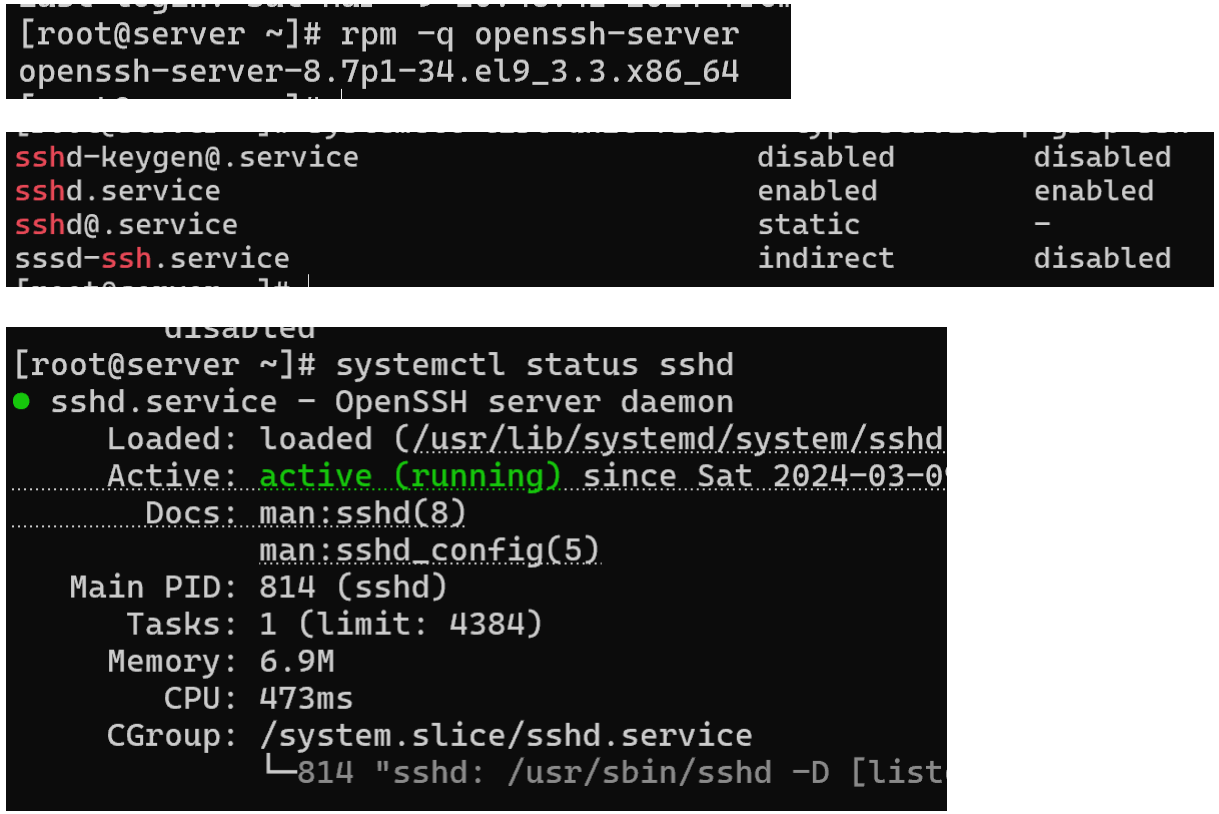
Paloma Pérez de Madrid

ÍNDICE

ÍNDICE

1. COMPROBAR QUE EL SERVIDOR OPENSSH ESTÁ INSTALADO EN EL EQUIPO

- Comprobar que el servidor OpenSSH está instalado en el equipo (en caso contrario, hacer la instalación con el paquete openssh-server). Identificar su nombre simbólico para systemd (sshd o ssh) e iniciar o reiniciar el servicio. Dar de alta en el servidor alguna cuenta de usuario (con su correspondiente directorio home) y asignar contraseñas. ¿Qué comandos ha usado para estas tareas?



1. COMPROBAR QUE EL SERVIDOR OPENSSH ESTÁ INSTALADO EN EL EQUIPO

- Comprobar que el servidor OpenSSH está instalado en el equipo (en caso contrario, hacer la instalación con el paquete openssh-server). Identificar su nombre simbólico para systemd (sshd o ssh) e iniciar o reiniciar el servicio. Dar de alta en el servidor alguna cuenta de usuario (con su correspondiente directorio home) y asignar contraseñas. ¿Qué comandos ha usado para estas tareas?

```
[root@server ~]# useradd -m palomassh
```

```
[root@server ~]# passwd palomassh
Cambiando la contraseña del usuario palomassh.
Nueva contraseña:
Vuelva a escribir la nueva contraseña:
passwd: todos los tokens de autenticación se actual
izaron exitosamente.
[root@server ~]#
```

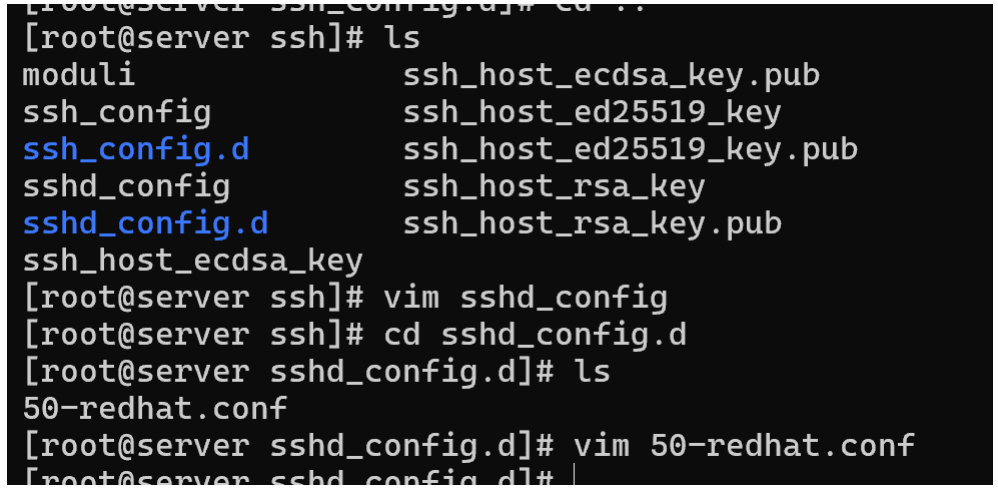
Contraseña: Practica1234+

ÍNDICE

2. ESTUDIAR EL ARCHIVO DE CONFIGURACIÓN DEL SERVIDOR

- Estudiar el archivo de configuración del servidor. Realizar algunas modificaciones sobre la configuración por defecto y comprobar su efecto reiniciando el servidor: cambiar el puerto de escucha, impedir el acceso remoto al usuario root, no realizar resoluciones DNS al inicio, y permitir conexiones sólo a algunos usuarios concretos.

Como es el servidor tenemos que modificar el archivo sshd_config



Se observa que el archivo sshd_config contiene los arvhicos de la carpeta sshd_config.d

Sin embargo, la directiva PermitRootLogin está en sshd_config al igual que la de Port. Añadiremos en este archivo las directivas

2. ESTUDIAR EL ARCHIVO DE CONFIGURACIÓN DEL SERVIDOR

```
Port 2222
UseDNS no
AllowUsers palomassh

#AddressFamily any
#ListenAddress 0.0.0.0
#ListenAddress ::

#HostKey /etc/ssh/ssh_host_rsa_key
#HostKey /etc/ssh/ssh_host_ecdsa_key
#HostKey /etc/ssh/ssh_host_ed25519_key

# Ciphers and keying
#RekeyLimit default none

# Logging
#SyslogFacility AUTH
#LogLevel INFO

# Authentication:

#LoginGraceTime 2m
#PermitRootLogin prohibit-password
PermitRootLogin no
```

/etc/ssh/sshd_config

```
[root@server ssh]# systemctl restart sshd
```

Verificación cambio de Puerto

```
PS C:\Users\ppere> ssh palomassh@192.168.119.138 -p
 2222
palomassh@192.168.119.138's password:
[palomassh@server ~]$ ls
[palomassh@server ~]$ pwd
/home/palomassh
[palomassh@server ~]$
```

Verificación que no se puede iniciar sesión con root

```
PS C:\Users\ppere> ssh root@192.168.119.138 -p 2222

root@192.168.119.138's password:
Permission denied, please try again.
```

2. ESTUDIAR EL ARCHIVO DE CONFIGURACIÓN DEL SERVIDOR

Comprobar que no haya mensajes relacionados con DNS en los registros del sistema al iniciar sesión

```
PS C:\Users\ppere> ssh palomassh@192.168.119.138 -p 2222
palomassh@192.168.119.138's password:
Last login: Tue Mar 19 20:26:08 2024 from 192.168.119.1
[palomassh@server ~]$ journalctl | grep DNS
Hint: You are currently not seeing messages from other users and the system.
      Users in groups 'adm', 'systemd-journal', 'wheel' can see all messages.
      Pass -q to turn off this notice.
```

ÍNDICE

3. USAR EL CLIENTE SSH PARA ACCEDER A NUESTRO SERVIDOR OPENSSH

Utilizar el cliente SSH desde nuestra máquina anfitriona (ssh y/o putty) para acceder a nuestro servidor OpenSSH. Utilizar cuentas de usuario válidas en el servidor remoto y autentificación por contraseña.

Línea de Comandos Máquina Anfitriona (Windows)

```
C:\Users\ppere>ssh root@192.168.119.138
ssh: connect to host 192.168.119.138 port 22: Connection refused

C:\Users\ppere>ssh root@192.168.119.138
root@192.168.119.138's password:
Permission denied, please try again.
root@192.168.119.138's password:
Permission denied, please try again.
root@192.168.119.138's password:
root@192.168.119.138: Permission denied (publickey,gssapi-keyex,gssapi-with-mic,password).

C:\Users\ppere>ssh root@192.168.119.138
root@192.168.119.138's password:
Last failed login: Wed Mar 20 11:32:23 CET 2024 from 192.168.119.1 on ssh:notty
There were 5 failed login attempts since the last successful login.
Last login: Tue Mar 19 19:44:54 2024 from 192.168.119.1
[root@server ~]#
```

```
AllowUsers palomassh root
```

```
#PermitRootLogin no
PermitRootLogin yes
```

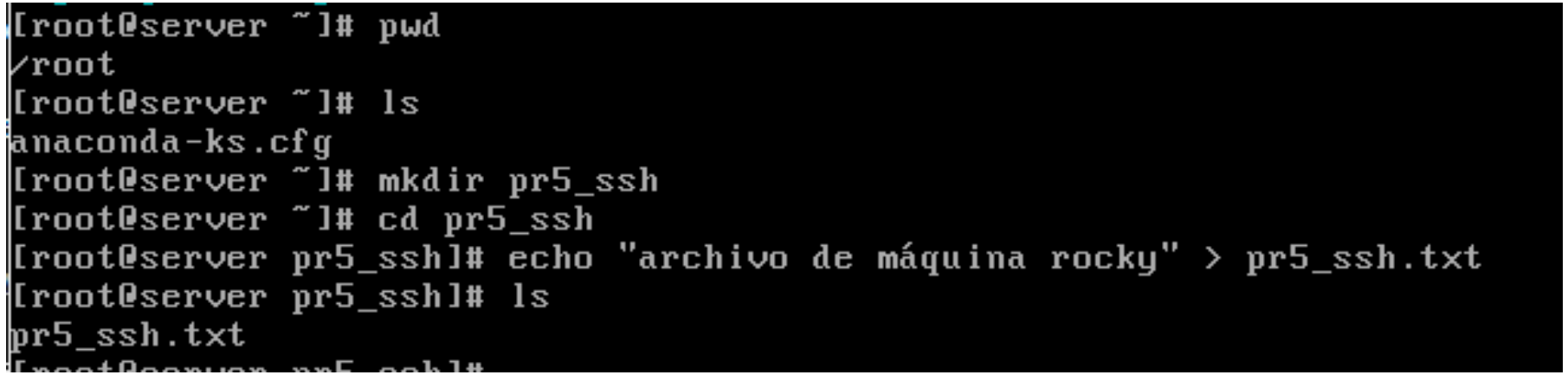
Cambiamos el archivo /etc/ssh/sshd_config y reiniciamos el servicio sshd para ver que cambia

ÍNDICE

4. UTILIZAR LOS COMANDOS SCP Y SFTP Y EL EXPLORADOR DE ARCHIVOS O EL PROGRAMA WINSCP.EXE

Utilizar los comandos scp y sftp (consola Linux/Mac o PowerShell de Windows) y el Explorador de Archivos o el programa WinSCP.exe para intercambiar archivos de forma segura entre el cliente y un servidor SSH.

Creamos un archivo en nuestra máquina Rocky:



4. UTILIZAR LOS COMANDOS SCP Y SFTP Y EL EXPLORADOR DE ARCHIVOS O EL PROGRAMA WINSCP.EXE

Utilizar los comandos scp y sftp (consola Linux/Mac o PowerShell de Windows) y el Explorador de Archivos o el programa WinSCP.exe para intercambiar archivos de forma segura entre el cliente y un servidor SSH.

Con scp desde la máquina anfitriona:

```
PS C:\Users\ppere\Desktop\AdminSist> scp root@192.168.119.138:/root/pr5_ssh/pr5_ssh.txt .
root@192.168.119.138's password:
pr5_ssh.txt                                                   100%   26     4.0KB/s   00:00
PS C:\Users\ppere\Desktop\AdminSist> dir


    Directorio: C:\Users\ppere\Desktop\AdminSist


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-----        20/03/2024     11:55                Practicas
-a----        04/03/2024      7:41      828791296 CentOS-6.10.ova
-a----        20/03/2024     11:55             26 pr5_ssh.txt
-a----        08/02/2024     11:02     1803672576 RockyLinux-9.3.ova
-a----        08/02/2024     11:06     2637218304 UbuntuServer-22.04-3.ova
```

4. UTILIZAR LOS COMANDOS SCP Y SFTP Y EL EXPLORADOR DE ARCHIVOS O EL PROGRAMA WINSCP.EXE

Utilizar los comandos scp y sftp (consola Linux/Mac o PowerShell de Windows) y el Explorador de Archivos o el programa WinSCP.exe para intercambiar archivos de forma segura entre el cliente y un servidor SSH.

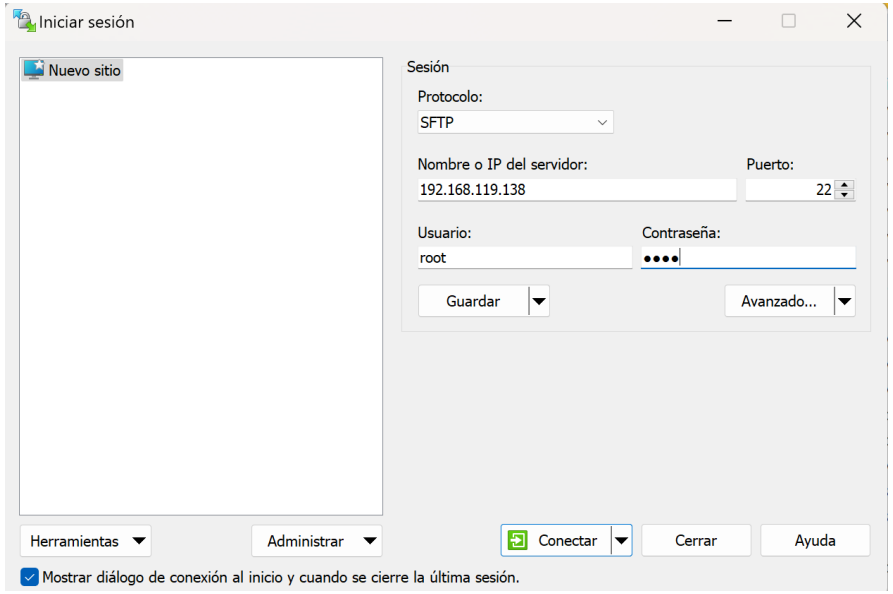
Con sftp desde la máquina anfitriona:

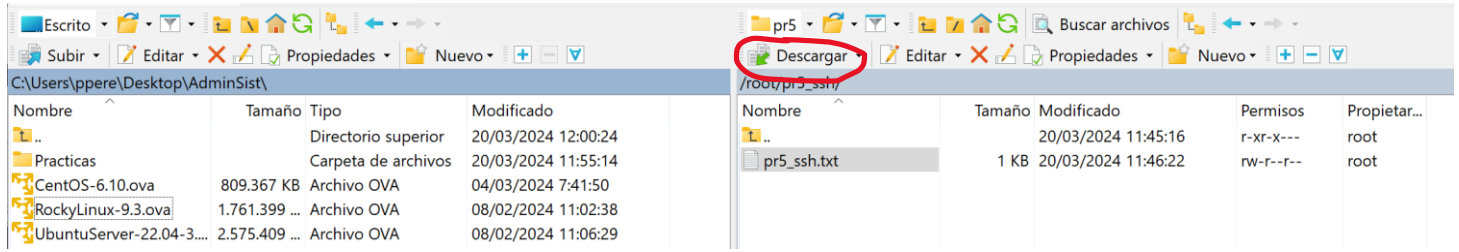
```
PS C:\Users\ppere\Desktop\AdminSist> sftp root@192.168.119.138
root@192.168.119.138's password:
Connected to 192.168.119.138.
sftp> cd /root/pr5_ssh
sftp> ls
pr5_ssh.txt
sftp> get pr5_ssh.txt .
Fetching /root/pr5_ssh/pr5_ssh.txt to ./pr5_ssh.txt
/root/pr5_ssh/pr5_ssh.txt                                                   100%   26     1.9KB/s   00:00
sftp> exit
PS C:\Users\ppere\Desktop\AdminSist> dir


    Directorio: C:\Users\ppere\Desktop\AdminSist


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-----        20/03/2024     11:55                Practicas
-a----        04/03/2024      7:41      828791296 CentOS-6.10.ova
-a----        20/03/2024     11:58             26 pr5_ssh.txt
-a----        08/02/2024     11:02     1803672576 RockyLinux-9.3.ova
-a----        08/02/2024     11:06     2637218304 UbuntuServer-22.04-3.ova
```

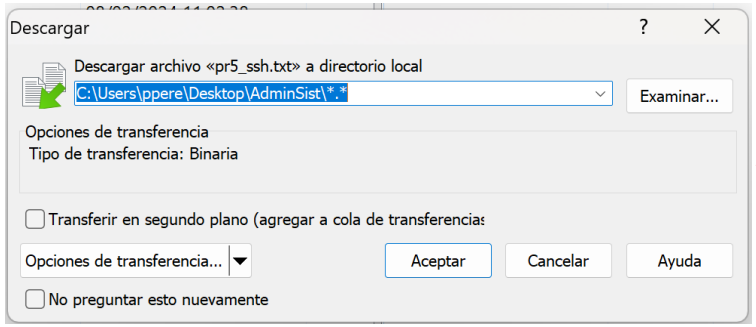
4. UTILIZAR LOS COMANDOS SCP Y SFTP Y EL EXPLORADOR DE ARCHIVOS O EL PROGRAMA WINSCP.EXE

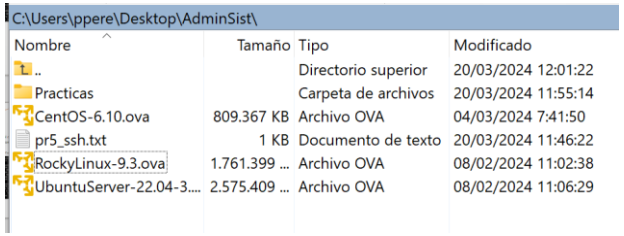
Utilizar los comandos scp y sftp (consola Linux/Mac o PowerShell de Windows) y el Explorador de Archivos o el programa WinSCP.exe para intercambiar archivos de forma segura entre el cliente y un servidor SSH.

Con WinSCP.exe:





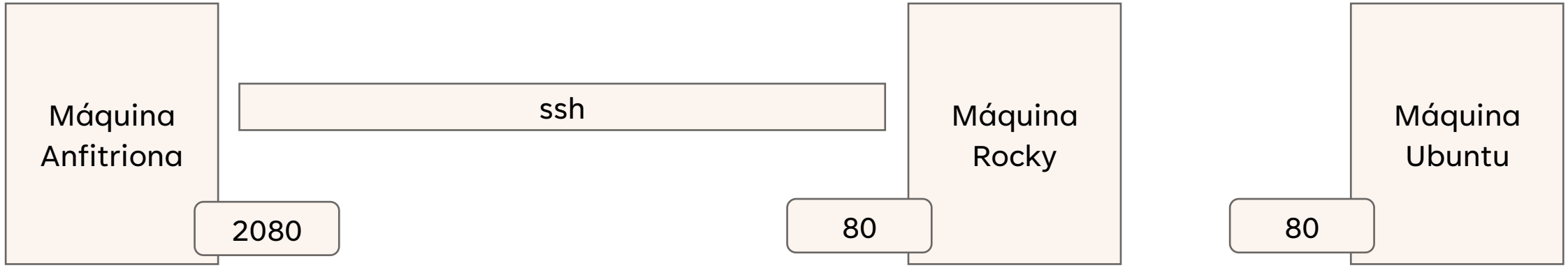




ÍNDICE

5. CONFIGURAR UN TÚNEL LOCAL SSH DESDE NUESTRO CLIENTE A UN SERVICIO ACTIVO DEL PROPIO SERVIDOR

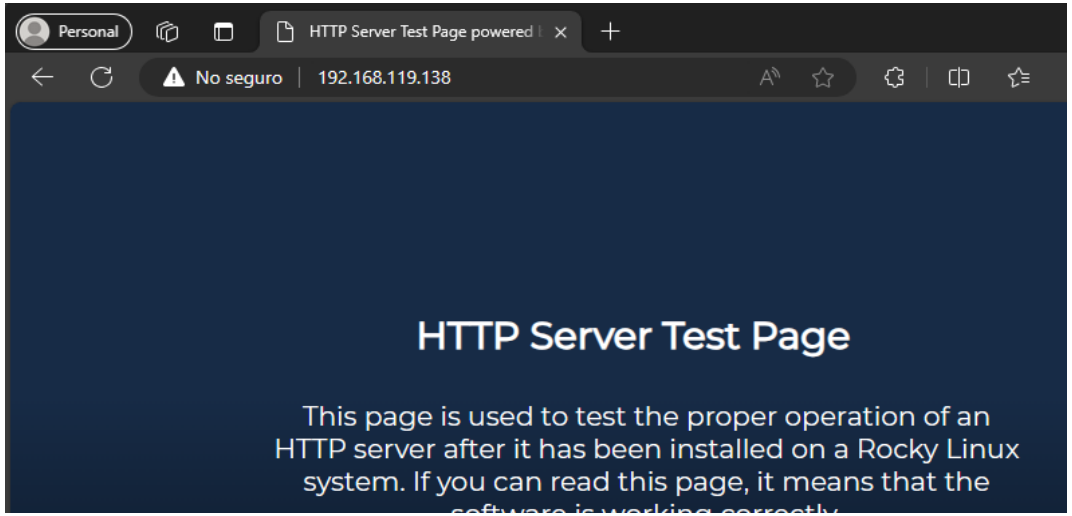
Configurar un túnel local SSH desde nuestro cliente (ssh y/o putty) a un servicio activo del propio servidor (HTTP, POP3, ...). A continuación, abrir la conexión SSH y comprobar que podemos acceder al servicio remoto a través del túnel. Probar igualmente a redirigir las peticiones que entran por el túnel hacia otro servidor (por ejemplo, abrir un túnel desde el puerto cliente 2080 a través de nuestro servidor SSH y que alcance otro host (otra máquina virtual) con el servicio web arrancado en su puerto 80).

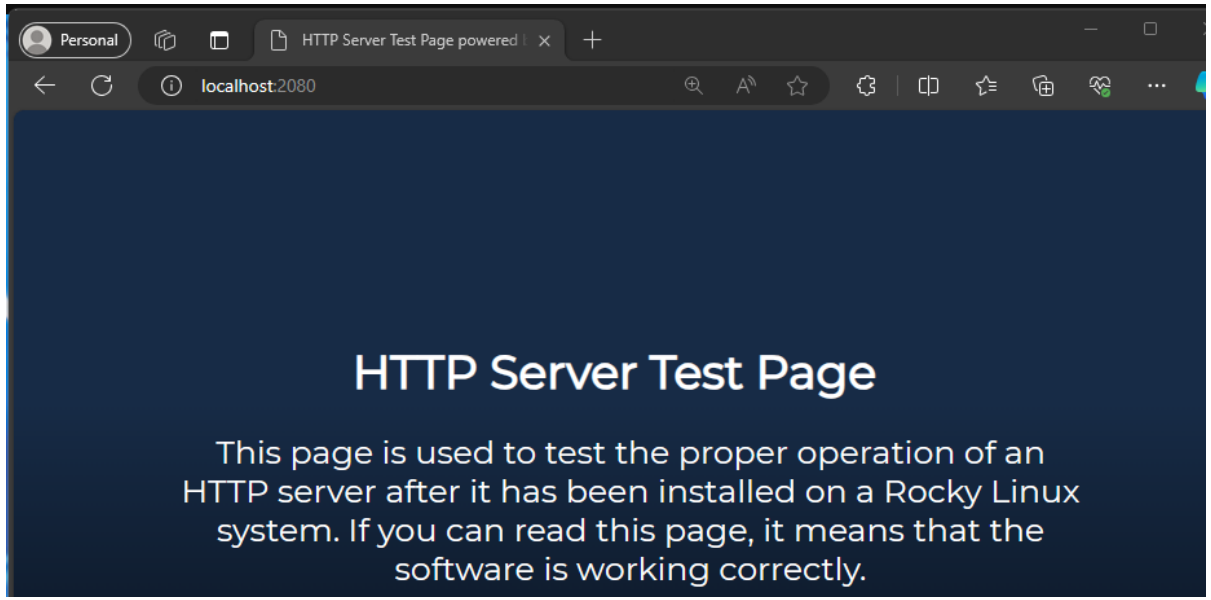


5. CONFIGURAR UN TÚNEL LOCAL SSH DESDE NUESTRO CLIENTE A UN SERVICIO ACTIVO DEL PROPIO SERVIDOR

Para comprobar que el túnel se ha hecho de forma correcta, instalaremos httpd en la máquina Rocky. De esta forma, si en la máquina anfitriona buscamos http://localhost:2080 nos redigirá automáticamente al servidor web configurado en la máquina rocky (puerto 80)

```
PS C:\Users\ppere> ssh -L 2080:192.168.119.138:80 root@192.168.119.138
root@192.168.119.138's password:
Last login: Wed Mar 20 12:51:35 2024 from 192.168.119.1
[root@server ~]#
```



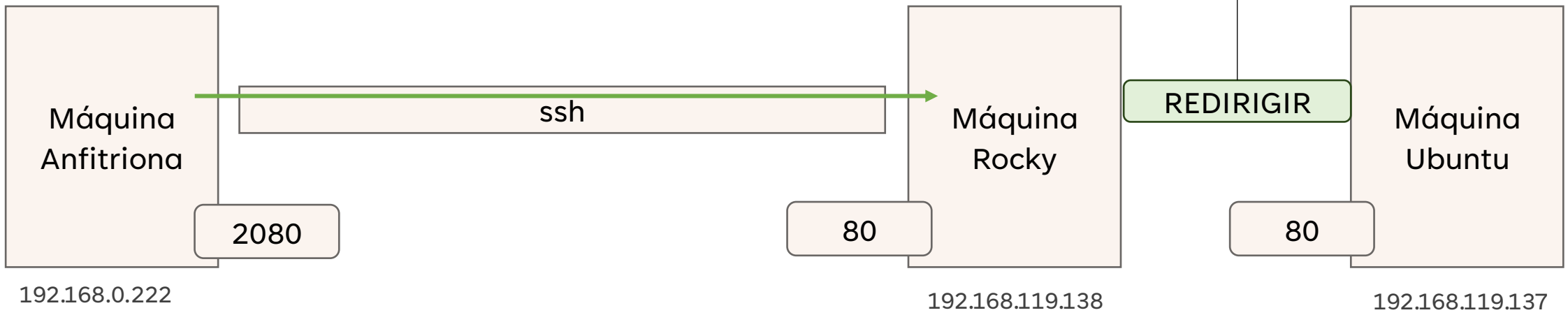


5. CONFIGURAR UN TÚNEL LOCAL SSH DESDE NUESTRO CLIENTE A UN SERVICIO ACTIVO DEL PROPIO SERVIDOR

Probar igualmente a redirigir las peticiones que entran por el túnel hacia otro servidor (por ejemplo, abrir un túnel desde el puerto cliente 2080 a través de nuestro servidor SSH y que alcance otro host (otra máquina virtual) con el servicio web arrancado en su puerto 80)

Redirigir las peticiones desde la VM Rocky hacia la VM Ubuntu

```
iptables -t nat -A PREROUTING -p tcp --dport 80
-j DNAT --to-destination 192.168.119.137:80
```



Nota Teo: El segundo túnel es correcto, y funcionará sin necesidad de la regla iptables de prerouting.

5. CONFIGURAR UN TÚNEL LOCAL SSH DESDE NUESTRO CLIENTE A UN SERVICIO ACTIVO DEL PROPIO SERVIDOR

```
PS C:\Users\ppere> ssh -L 2080:192.168.119.137:80 root@192.168.119.138
root@192.168.119.138's password:
Last login: Wed Mar 20 15:24:32 2024 from 192.168.119.1
[root@server ~]#
```

```
[root@server ~]# iptables -t nat -A PREROUTING -p tcp --dport 80 -j DNAT --to-destination 192
.168.119.137:80
```

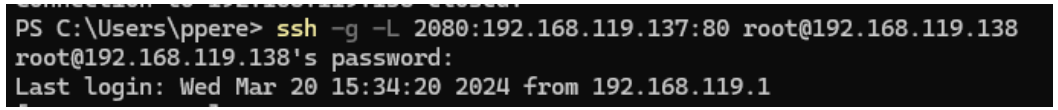
Comprobamos que dicha página corresponde al servidor web de Ubuntu del puerto 80

Desde la Máquina Anfitriona (Windows)

ÍNDICE

6. REPETIR EL APARTADO ANTERIOR PERMITIENDO CONEXIONES A NUESTRO TÚNEL DESDE OTROS EQUIPOS

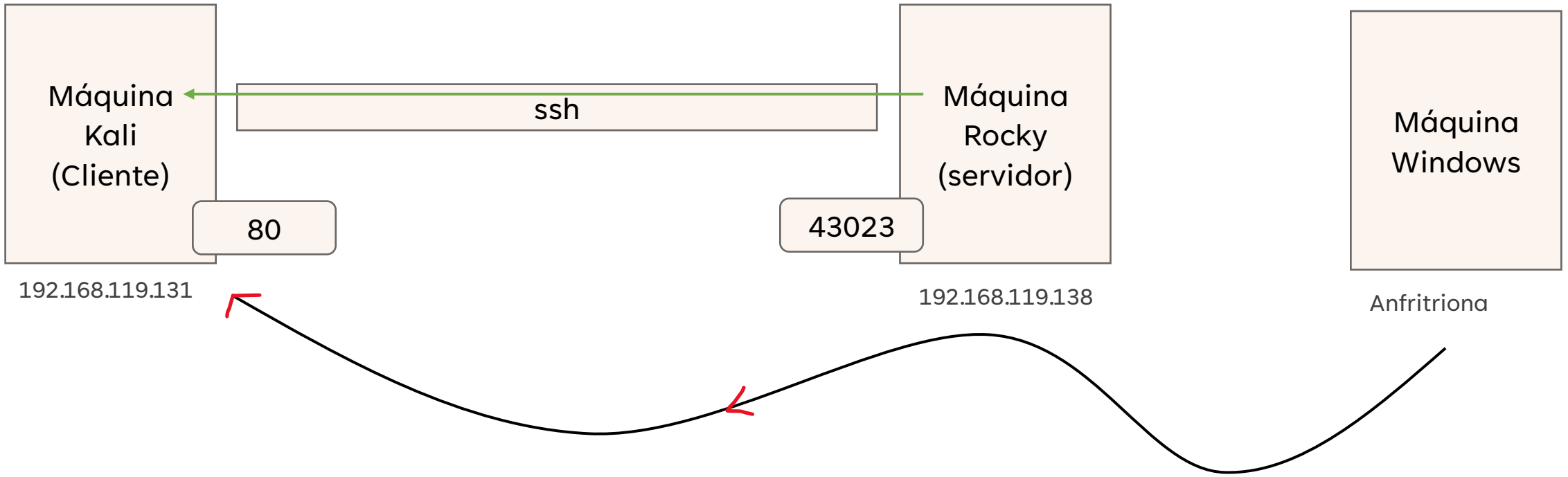
Repetir el apartado anterior permitiendo conexiones a nuestro túnel desde otros equipos (opción -g en SSH o Connection → SSH → Tunnels → Local ports accept connection from other hosts en Putty). Usar la máquina anfitriona o una nueva máquina virtual para acceder a través del túnel al servicio ofrecido.



Desde otro dispositivo dentro de la red buscamos la dirección IP de la máquina anfitriona con el puerto (192.168.0.222:2080)

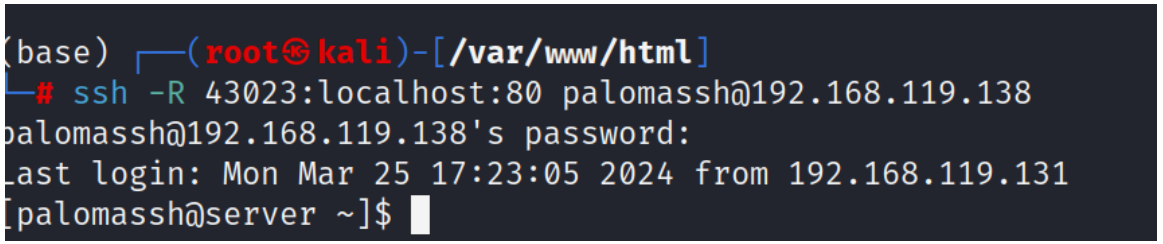
ÍNDICE

7. ESTABLECER UN POSIBLE CONTEXTO PARA PROBAR EL FUNCIONAMIENTO DE UN TÚNEL INVERSO SSH (REVERSE)



7. ESTABLECER UN POSIBLE CONTEXTO PARA PROBAR EL FUNCIONAMIENTO DE UN TÚNEL INVERSO SSH (REVERSE)

ssh -R 43023:localhost:80 palomassh@192.168.119.138 (kali)



Túnel SSH inverso desde un puerto específico en el servidor remoto (`192.168.119.138`) hacia un servicio en tu máquina local.

- **ssh**
- **-R 43023:localhost:80**: "Redirige el tráfico que llega al puerto 43023 en el servidor remoto hacia el puerto 80 de la máquina local".

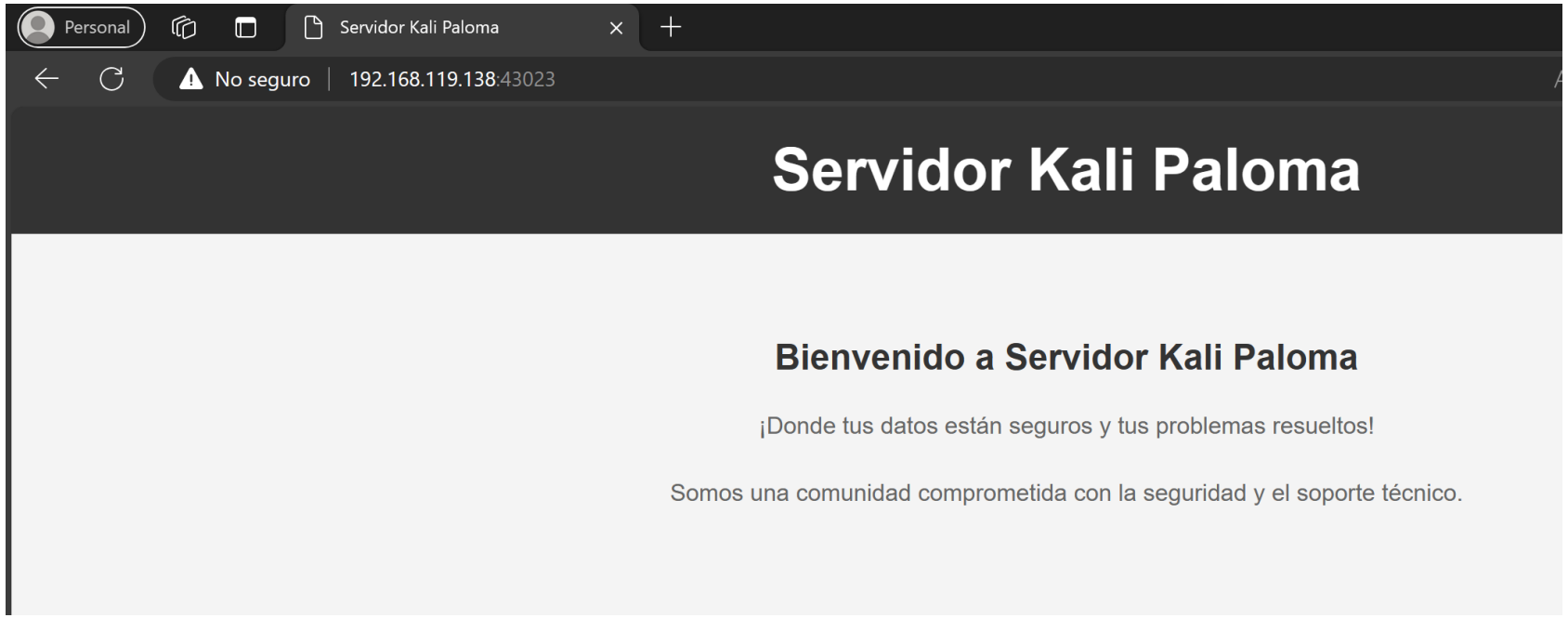
- **43023**: Es el puerto en el servidor remoto (Rocky) al que se conectará cualquier tráfico entrante.
- **localhost:80**: Es el destino (máquina local, Kali) al que se redirigirá el tráfico desde el servidor remoto.

- **palomassh@192.168.119.138**: Especifica el usuario y la dirección IP del servidor remoto (`192.168.119.138`, Rocky) al que se conectará SSH.

7. ESTABLECER UN POSIBLE CONTEXTO PARA PROBAR EL FUNCIONAMIENTO DE UN TÚNEL INVERSO SSH (REVERSE)

Desde la máquina anfitriona comprobamos que desde el puerto 43023 de la máquina Rocky (Servidor ssh) se puede acceder al puerto 80 de la máquina Kali (um servidor web)

Máquina Anfitriona: 192.168.119.138:43023

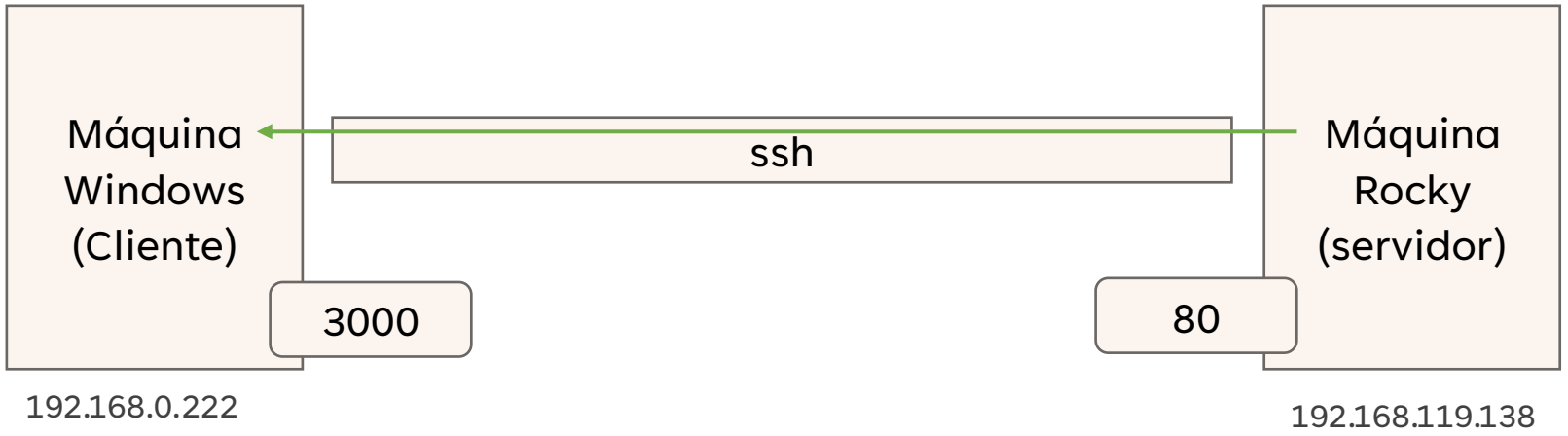


ÍNDICE

8. PARA UN USUARIO DE PRUEBA EN EL SERVIDOR SSH CONFIGURAR SU AUTENTIFICACIÓN BASADA EN CLAVES PÚBLICA/PRIVADA

Para un usuario de prueba en el servidor SSH configurar su autentificación basada en claves pública/privada. Para generar las claves en el cliente Linux utilizar el comando ssh keygen. En clientes Windows puede utilizar el programa puttygen.exe (ojo: se usa una clave pública RSA de 2048 bits. La clave pública obtenida en formato OpenSSH aparece en la ventana superior). Comprobar el efecto que tiene no proteger con frase de paso la clave privada.

Nota: es posible cambiar la frase de paso de una clave privada con la opción –p de ssh-keygen (ssh-keygen –p –f id_rsa)



8. PARA UN USUARIO DE PRUEBA EN EL SERVIDOR SSH CONFIGURAR SU AUTENTIFICACIÓN BASADA EN CLAVES PÚBLICA/PRIVADA

Activar la autenticación basada en claves en el servidor (/etc/ssh/sshd_config en Rocky):

```
PubkeyAuthentication yes
```

Systemctl restart sshd

Creamos las claves con ssh-keygen en el cliente (Ubuntu)

```
root@server:~# ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa
Your public key has been saved in /root/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:OCk9rKzMPnIBKJChhL0FKbbnyr9PNRbM6iQx3py+lGc root@server
The key's randomart image is:
+---[RSA 3072]----+
|o=.o             |
|B.o . o          |
|=..=   +         |
|+.o.=ooo.        |
|..oo.*B+S        |
|  .o=o+o.        |
|. ..o* E         |
|.=o.o +          |
| +*+oo           |
+----[SHA256]-----+
```

Frase de Paso: palomassh

Ahora copiaremos la clave pública en el servidor (Rocky) , lo guardará en un archivo llamado 'authority_keys'

```
root@server:/etc/ssh# ssh-copy-id -p22 palomassh@192.168.119.138
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/root/.ssh/id_rsa.pub"
The authenticity of host '192.168.119.138 (192.168.119.138)' can't be established.
ED25519 key fingerprint is SHA256:H4SZXenDPKX2NRqNaDLMpE+TA8SCLlWeTTaMp4T4J8A.
This host key is known by the following other names/addresses:
    ~/.ssh/known_hosts:1: [hashed name]
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to install the new keys
palomassh@192.168.119.138's password:
Permission denied, please try again.
palomassh@192.168.119.138's password:

Number of key(s) added: 1

Now try logging into the machine, with:   "ssh -p '22' 'palomassh@192.168.119.138'"
and check to make sure that only the key(s) you wanted were added.
```

Comprobamos que el usuario palomassh tiene la clave pública

```
[palomassh@server ~]$ cat ~/.ssh/authorized_keys
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABgQC2GFTP6NOsNPQCDm12ehNjnvXpKC0PIsXaBneE8b20O5S
ccEI2hUFoHnKriiLdqAfWSYRdV56hAR2gTDNPc83rkHAJJHJMu5ZBso+jfUkQrEEdkTDq6VjWrGnXszFU8Q
bGWUEThz/GzyUWXXJJ7VLqgaCD+fcZsVNE2af1dz+wBWKf/M0PEkQgrDNXEFvteDq9O/ZeDhL3E6jwK6PPh
FQ8imWkDyj/YJveIuXURzXDQQ13LGeKqsKz8msOz1eFBViUgM2vjx4/rVvF2YGV3vWJZvggGf6zqnq8NBJw
F8AULLMxcuVirK1qbK9rOyUW5+5syzQuTRUjKqB5nUhXbyNHT/Tu1lhLDLL00RR78K2TZ6oO7A97ZeX02q1
wkLQJISJoE12lO3fA2dhIR9lyDG+D4QPtSYTZJbbu5hoNpbBIbdzO8SinYdVbTDdqcqZzGaSZQvN19BAJNm
+iz4fzsFXiyhS+ViOFqUuiwXPKPmeIBHjbwBvfQuyqZhuBWxxnUW8= root@server
[palomassh@server ~]$
```

8. PARA UN USUARIO DE PRUEBA EN EL SERVIDOR SSH CONFIGURAR SU AUTENTIFICACIÓN BASADA EN CLAVES PÚBLICA/PRIVADA

Si hacemos una conexión desde el cliente al servidor (Rocky) nos pedirá la frase de paso

```
root@server:/etc/ssh# ssh palomassh@192.168.119.138 -p22
Enter passphrase for key '/root/.ssh/id_rsa':
Last login: Wed Mar 20 19:28:44 2024 from 192.168.119.1
[palomassh@server ~]$ uname -a
Linux server 5.14.0-362.18.1.el9_3.0.1.x86_64 #1 SMP PREEMPT_DYNAMIC Sun Feb 11 13:
49:23 UTC 2024 x86_64 x86_64 x86_64 GNU/Linux
[palomassh@server ~]$
```

Queda demostrado que desde Ubuntu podemos acceder a Rocky con una frase de paso

Frase de Paso: palomassh

ÍNDICE

9. BUSCAR ALGUNA SOLUCIÓN QUE PERMITA AUTOMATIZAR LA EJECUCIÓN REMOTA SIN TENER QUE INTRODUCIR LA CONTRASEÑA CADA VEZ QUE SE EJECUTA EL COMANDO.

Considere que necesita crear un script para ejecutar algún comando en un equipo remoto, y que la conexión a ese equipo se realiza a través de ssh (tenemos una cuenta de usuario en el servidor). Buscar alguna solución que permita automatizar la ejecución remota sin tener que introducir la contraseña cada vez que se ejecuta el comando. Es posible usar la autentificación basada en clave pública/privada del apartado anterior (sin proteger la clave privada con frase de paso), pero también existen utilidades que nos permiten pasar el usuario y contraseña de autentificación en la llamada a la ejecución remota (buscar por automate SSH login).

Una opción es con **sshpass**

COMENTARIOS TEO

5. El segundo túnel es correcto, y funcionará sin necesidad de la regla iptables de prerouting.