Práctica 5: Entorno seguro para usuarios SSH

Completa los comandos necesarios en los huecos en blanco

Supongamos que administramos un servidor para alojamiento compartido profesional (hosting) en el que los usuarios sólo van a poder acceder a su directorio “home”. Estos usuarios no van a poder leer ni acceder a ninguna parte del sistema de ficheros que no esté bajo su directorio home

Creamos un grupo nuevo llamado “hostusers”.

Los usuarios de este grupo tendrán automáticamente restringido el sistema de directorios a su propio directorio cerrado. No pudiendo salir ni ver más allá de su directorio raíz.

groupadd hostusers

```
root@ladynightmare-E5-522-4154:~# groupadd hostusers
```

Creamos el usuario.

Este usuario tiene la particularidad que no podrá acceder a un terminal y sólo podrá acceder mediante un cliente sftp a sus directorios cerrados. Le asignamos una contraseña a este usuario

useradd -g hostusers -d /hosting/testuser/home -s /sbin/nologin testuser

```
root@ladynightmare-E5-522-4154:~# useradd -g hostusers -d /hosting/testuser/home -s /sbin/nologin testuser
```

Verificamos que el usuario se ha creado correctamente consultando si ha incluido la línea del usuario en el fichero /etc/passwd/

grep testuser /etc/passwd

```
root@ladynightmare-E5-522-4154:~# grep testuser /etc/passwd
testuser:x:1001:1001::/hosting/testuser/home:/sbin/nologin
```

Configuramos el servicio sftp cerrado.

Para evitar que el usuario sólo pueda usar el protocolo sftp y no otras opciones disponibles en SSH, tenemos que cambiar la configuración de OpenSSH.

Para ello editamos el fichero sshd_config donde se configura el servicio del servidor ssh. En este archivo preparamos ssh para funcionar como queramos. Cambiando el sftp-server que viene configurado por defecto por el “internal-sftp”, ya que es compatible con usuarios de entorno cerrado.

```
sudo nano /etc/ssh/sshd_config
```

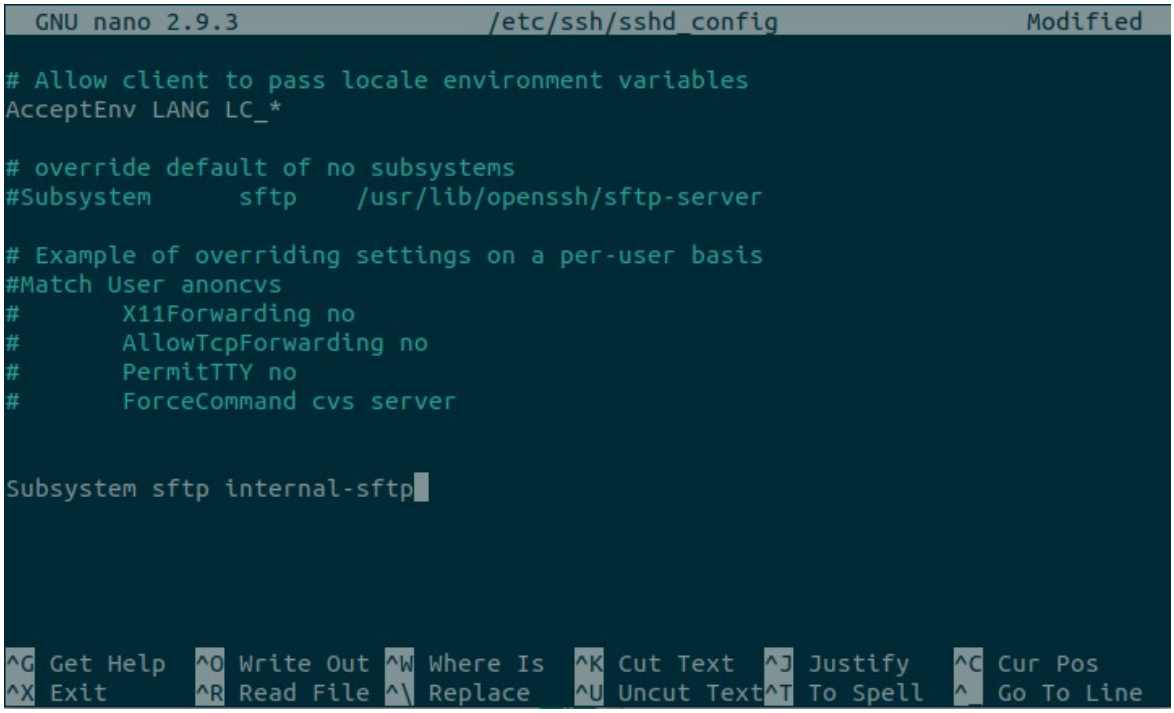
Comentamos la línea para deshabilitarla:

```
#Subsystem sftp /usr/lib/openssh/sftp-server
```

```
  GNU nano 2.9.3                  /etc/ssh/sshd_config                  Modified

#UseDNS no
#PidFile /var/run/sshd.pid
#MaxStartups 10:30:100
#PermitTunnel no
#ChrootDirectory none
#VersionAddendum none

# no default banner path
#Banner none

# Allow client to pass locale environment variables
AcceptEnv LANG LC_*

# override default of no subsystems
#Subsystem      sftp    /usr/lib/openssh/sftp-server

# Example of overriding settings on a per-user basis
#Match User anoncvs
#       X11Forwarding no

^G Get Help  ^O Write Out ^W Where Is  ^K Cut Text  ^J Justify   ^C Cur Pos
^X Exit      ^R Read File ^\ Replace   ^U Uncut Text^T To Spell  ^_ Go To Line
```

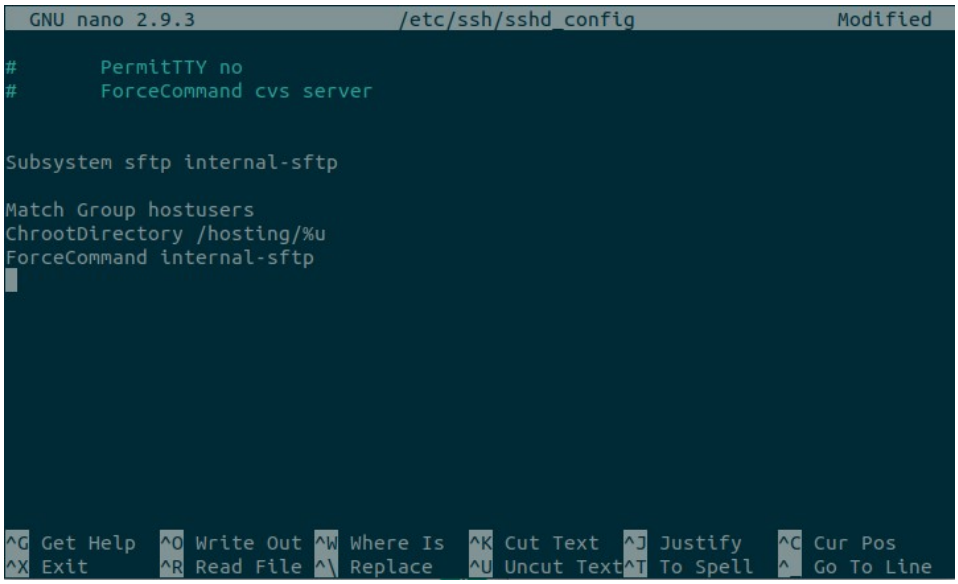
Al **final del archivo de configuración** añadimos una línea nueva para que use el sftp interno:

```
Subsystem sftp internal-sftp
```



Y después añadimos el siguiente texto para instruir al servicio ssh cual será el directorio cerrado.

```
Match Group hostusers
ChrootDirectory /hosting/%u
ForceCommand internal-sftp
```



La primera línea es una condicional e indica que las siguientes sólo aplican a los usuarios del grupo *hostusers*.

La segunda línea indica que los usuarios estarán encapsulados en el directorio especificado.

Y la tercera indica que se debe forzar para estos usuarios el uso del sftp interno e Impide al usuario usar otro comando.

Creamos el directorio cerrado.

```
sudo mkdir /hosting
```

```
root@ladynightmare-E5-522-4154:~# sudo mkdir /hosting
```

Creamos los directorios para el usuario.

```
sudo mkdir /hosting/testuser
sudo mkdir /hosting/testuser/home
sudo mkdir /hosting/testuser/home/public_html
```

```
root@ladynightmare-E5-522-4154:~# sudo mkdir /hosting/testuser
root@ladynightmare-E5-522-4154:~# sudo mkdir /hosting/testuser/home
root@ladynightmare-E5-522-4154:~# sudo mkdir /hosting/testuser/home/public_html
```

Para estos usuarios especiales *"/hosting/testuser"* será como *"/"* para el usuario *testuser*. Este usuario no podrá ver nada por encima de su directorio raíz. El

directorio “*public_html*” queda ahí para poder usarlo con Apache y orientar los dominios virtuales a ese directorio. De modo que cuando esté configurado Apache, lo que cada usuario suba ahí se publicará en Internet.

Aplicamos los permisos correctos.

Como estos directorios son creados por *root* debemos cambiarles el propietario para que el usuario *testuser* pueda usarlos.

chown testuser:hostusers /hosting/testuser/home

```
root@ladynightmare-E5-522-4154:~# chown testuser:hostusers /hosting/testuser/home
```

Reiniciar el servicio SSH

Finalmente reiniciamos el servicio ssh para que los cambios aplicados tengan efecto.

`sudo service ssh restart`

```
root@ladynightmare-E5-522-4154:~# service ssh restart
```

Comprobación del funcionamiento

```
root@ladynightmare-E5-522-4154:~# ssh testuser@localhost
testuser@localhost's password:
This service allows sftp connections only.
Connection to localhost closed.
root@ladynightmare-E5-522-4154:~# sftp testuser@localhost
testuser@localhost's password:
Connected to localhost.
sftp> pwd
Remote working directory: /
sftp> ls
home
sftp> cd ../
sftp> ls
home
```

Creación de nuevos usuarios en este entorno

Para crear cada usuario adicional en este entorno seguro tendremos que realizar los siguientes pasos.

Creamos el usuario con ‘useradd -g hostusers -d /hosting/<nombre de usuario>/home -s /sbin/nologin <nombre de usuario>’
Creamos el directorio home del usuario con ‘mkdir /hosting/<nombre de usuario>-p /home/public_html’
Cambiamos los permisos al directorio con ‘chown <nombre de usuario>:hostusers /hosting/<nombre de usuario>/home’
Reiniciamos el servicio con ‘service ssh restart’