

Usaremos dos máquinas Ubuntu 18.04 en RED NAT

En la maquina que utilizaremos como servidor:

1. Instalar openssh-server. Comprobar el estado del servicio.

Usamos el comando `sudo apt install ssh`, aceptamos para empezar la descarga.

```
abc@abc-VirtualBox:~$ sudo apt install ssh
Leyendo lista de paquetes... Hecho
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes adicionales:
  ncurses-term openssh-client openssh-server openssh-sftp-server ssh-import
Paquetes sugeridos:
  keychain libpam-ssh monkeysphere ssh-askpass molly-guard
Se instalarán los siguientes paquetes NUEVOS:
  ncurses-term openssh-server openssh-sftp-server ssh ssh-import-id
Se actualizarán los siguientes paquetes:
  openssh-client
1 actualizados, 5 nuevos se instalarán, 0 para eliminar y 299 no actualizados
Se necesita descargar 693 kB/1.364 kB de archivos.
```

Revisamos el estado del servicio con el comando `systemctl status ssh`:

```
abc@abc-VirtualBox:~$ systemctl status ssh
● ssh.service - OpenBSD Secure Shell server
   Loaded: loaded (/lib/systemd/system/ssh.service; enabled; vendor preset: enabled)
   Active: active (running) since Wed 2022-01-26 11:36:11 CET; 1min 4s ago
     Docs: man:sshd(8)
           man:sshd_config(5)
   Main PID: 3243 (sshd)
     Tasks: 1 (limit: 2703)
    Memory: 1.0M
   CGroup: /system.slice/ssh.service
           └─3243 sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups

ene 26 11:36:11 abc-VirtualBox systemd[1]: Starting OpenBSD Secure Shell server...
ene 26 11:36:11 abc-VirtualBox sshd[3243]: Server listening on 0.0.0.0 port 22.
ene 26 11:36:11 abc-VirtualBox sshd[3243]: Server listening on :: port 22.
ene 26 11:36:11 abc-VirtualBox systemd[1]: Started OpenBSD Secure Shell server.
abc@abc-VirtualBox:~$
```

Si no está activado usaríamos `sudo systemctl start ssh` y para desactivarlo con `sudo systemctl stop ssh`.

En la máquina que utilizaremos como cliente

2. Instalar openssh-client y realizar una conexión

Usamos el comando `sudo apt-get install openssh-client`

```

abc@abc-Cliente:~$ sudo apt-get install openssh-client
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Paquetes sugeridos:
  keychain libpam-ssh monkeysphere ssh-askpass
Se actualizarán los siguientes paquetes:
  openssh-client
1 actualizados, 0 nuevos se instalarán, 0 para eliminar y 299 no actualizados
Se necesita descargar 0 B/671 kB de archivos.
Se utilizarán 0 B de espacio de disco adicional después de esta operación.
(Leyendo la base de datos ... 160336 ficheros o directorios instalados...)

```

### 3. Generar el par de claves

Clave cliente:

Se usa el comando ssh-keygen

```

abc@abc-Cliente:~$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/abc/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/abc/.ssh/id_rsa
Your public key has been saved in /home/abc/.ssh/id_rsa.pub
Your terminal fingerprint is:
SHA256:++mdm8UmNSK4BfUX8SMtHHlaQDIagZy0zGke7Fd+UVU abc@abc-Cliente
The key's randomart image is:
+-----[RSA 3072]-----+
| 0.0000.+==OE|
| ++0.0.+.= |
| 0.. ..==0.|
| + .00 0+ .|
| oS.0...0 |
| .+ ..+ .|
| 0 . + |
| . 0 * |
| .+ =. |
+-----[SHA256]-----+

```

Comprobamos donde están ubicadas las claves:

```

abc@abc-Cliente:~$ ls .ssh
id_rsa  id_rsa.pub  known_hosts

```

### 4. Copiar la clave publica en el servidor usando ssh-copy-id

```

abc@abc-Cliente:~$ ssh-copy-id abc@192.168.2.2
The authenticity of host '192.168.2.2 (192.168.2.2)' can't be established.
ECDSA key fingerprint is SHA256:6rWMYsVln9orMS2rFSuLd+zCQu04UjbfS+0vXkBLWUI.
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 promp
ted now it is to install the new keys
abc@192.168.2.2's password:

Number of key(s) added: 1

Now try logging into the machine, with: "ssh 'abc@192.168.2.2'"
and check to make sure that only the key(s) you wanted were added.

```

5. Comprobar que podemos realizar una conexión sin necesidad de la contraseña de la cuenta remota

Para acceder la primera vez nos pedirá la passphrase

```

Now try logging into the machine, with: "ssh 'abc@192.168.2.2'"
and check to make sure that only the key(s) you wanted were added.

abc@abc-Cliente:~$ ssh abc@192.168.2.2
Welcome to Ubuntu 20.04.2 LTS (GNU/Linux 5.13.0-27-generic x86_64)

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

309 updates can be installed immediately.
147 of these updates are security updates.

```

Nuevamente en la máquina del servidor

6. Inhabilitar la autenticación con contraseña

Modificamos el archivo /etc/ssh/sshd\_config

```

GNU nano 4.8                               /etc/ssh/sshd_config      Modificado
#AuthorizedKeysCommand none
#AuthorizedKeysCommandUser nobody

# For this to work you will also need host keys in /etc/ssh/ssh_known_hosts
#HostbasedAuthentication no
# Change to yes if you don't trust ~/.ssh/known_hosts for
# HostbasedAuthentication
#IgnoreUserKnownHosts no
# Don't read the user's ~/.rhosts and ~/.shosts files
#IgnoreRhosts yes

# To disable tunneled clear text passwords, change to no here!
# PasswordAuthentication yes
#PermitEmptyPasswords no

# Change to yes to enable challenge-response passwords (beware issues with
# some PAM modules and threads)
ChallengeResponseAuthentication no

```

Descomentamos esa línea y en vez de un yes ponemos un no

```

# Don't read the user's ~/.rhosts and ~/.shosts files
#IgnoreRhosts yes

# To disable tunneled clear text passwords, change to no here!
PasswordAuthentication no
#PermitEmptyPasswords no

# Change to yes to enable challenge-response passwords (beware issues with
# some PAM modules and threads)
ChallengeResponseAuthentication no

# Kerberos options
#KerberosAuthentication no

```

Con otra máquina virtual comprobamos que no se puede acceder con contraseña:

```

abc@abc-VirtualBox:~$ ssh abc@192.168.2.2
ssh: connect to host 192.168.2.2 port 22: Connection refused
abc@abc-VirtualBox:~$

```

Busquedas:

- Configuración ssh servidor en ubuntu->  
<https://jcastaneda.com/servidores/configurar-ssh-en-ubuntu-server-20-04/?cn-reloaded=1>

- Instalar cliente: <https://howtoinstall.co/es/openssh-client>
- Generar claves: <https://noviello.it/es/como-configurar-claves-ssh-en-ubuntu-20-04-lts/>
- Uso de ssh sin contraseña: <https://www.hostinger.es/tutoriales/configurar-ssh-sin-contrasena-linux>
- Desactivar autenticación por contraseña para ssh:  
<https://www.enmimaquinafunciona.com/pregunta/151963/como-desactivar-la-autenticacion-por-contrasena-para-ssh>