



Práctica 10. Configurar servicio SSH entre cliente y servidor linux.

Fecha de realización: 10 - 11 - 22

Instrucciones

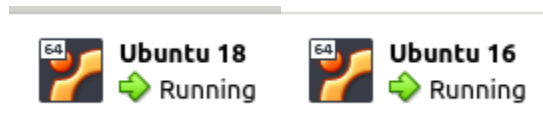
- 1.** Instala el servidor y cliente
- 2.** Configura el servidor y cliente para ejecutar aplicaciones gráficas y ejecuta xclock o libre office desde el cliente.
- 3.** Configura la autenticación de un usuario cliente por sistema de clave RSA al servidor.
- 4.** Realiza copia de archivos segura bajo ssh con scp desde cliente al servidor y desde el servidor al cliente.



Realiza una práctica de configuración del servicio ssh entre dos equipos linux y realiza un pequeño manual con capturas de pantallas y explicaciones de los pasos realizados:

1. Instala el servidor y cliente

Mi servidor ssh lo haré en un sistema operativo Ubuntu 18 y el cliente lo montaré en un Ubuntu 16.



En el servidor voy a utilizar el siguiente comando en nuestra terminal linux, para instalar el cliente y el servidor:

```
serverubu@serverubu:~$ sudo apt install ssh
```

En nuestro cliente no hace falta, porque ya viene preinstalado el cliente ssh, por lo que no necesitaremos instalar nada. Algo que es muy importante es tener nuestros equipos en la misma red, y mi servidor ssh, tiene también el servicio isc-dhcp-server corriendo, para dar servicio al cliente.

Configuración IP de mi servidor:

```
serverubu@serverubu:~$ ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.10.13.1 netmask 255.255.255.0 broadcast 10.10.13.255
    inet6 fe80::8777:c2e0:c434:41bb prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:08:6d:b8 txqueuelen 1000 (Ethernet)
    RX packets 234 bytes 36553 (36.5 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 279 bytes 38357 (38.3 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```



Configuración IP de mi cliente

```
cliente@cliente:~$ ifconfig
enp0s3    Link encap:Ethernet  HWaddr 08:00:27:c7:b7:1e
          inet addr:10.10.13.100  Bcast:10.10.13.255  Mask:255.255.255.0
          inet6 addr: fe80::851:da0c:d30a:383f/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:215 errors:0 dropped:0 overruns:0 frame:0
          TX packets:244 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:31619 (31.6 KB)  TX bytes:37381 (37.3 KB)
```

Ahora, vamos a comprobar que tenemos conectividad, iniciando sesión en nuestro cliente y conectándonos al servidor

```
cliente@cliente:~$ sudo ssh serverubu@10.10.13.1
serverubu@10.10.13.1's password:
Welcome to Ubuntu 18.04.6 LTS (GNU/Linux 5.4.0-84-generic x86_64)

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

224 updates can be applied immediately.
205 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

Failed to connect to https://changelogs.ubuntu.com/meta-release-lts. Check your
Internet connection or proxy settings

Your Hardware Enablement Stack (HWE) is supported until April 2023.
Last login: Sun Nov 13 12:24:08 2022 from 10.10.13.100
serverubu@serverubu:~$
```

Para hacer la conexión utilizaremos el nombre de usuario de nuestro server ("serverubu") y su IP (10.10.13.1). Como podemos comprobar en la imagen anterior ya tenemos conexión entre el cliente y el servidor.



2. Configura el servidor y cliente para ejecutar aplicaciones gráficas y ejecuta xclock o libre office desde el cliente.

Para poder hacer uso de aplicaciones gráficas necesitaremos acceder al archivo de configuración del servidor ssh (/etc/ssh/sshd_config). Descomentamos el siguiente parámetro y escribimos "yes":

```
X11Forwarding yes
```

En mi caso, le he dado unos permisos al archivo .xauthority de nuestro cliente y servidor, en concreto el 0600, ya que este archivo necesitaba esos permisos para poder redirigir el modo gráfico, haremos lo siguiente.

```
serverubu@serverubu:~$ ls -la
```

```
-rw----- 1 serverubu serverubu 55 nov 13 12:33 .Xauthority
```

```
serverubu@serverubu:~$ sudo chmod 0600 .Xauthority
```

Lo que acabamos de hacer lo realizaremos en el cliente también, de la misma manera.

Si no sabes dónde se encuentra tu archivo, ejecuta el siguiente comando y copiate el archivo a tu directorio de conexión:

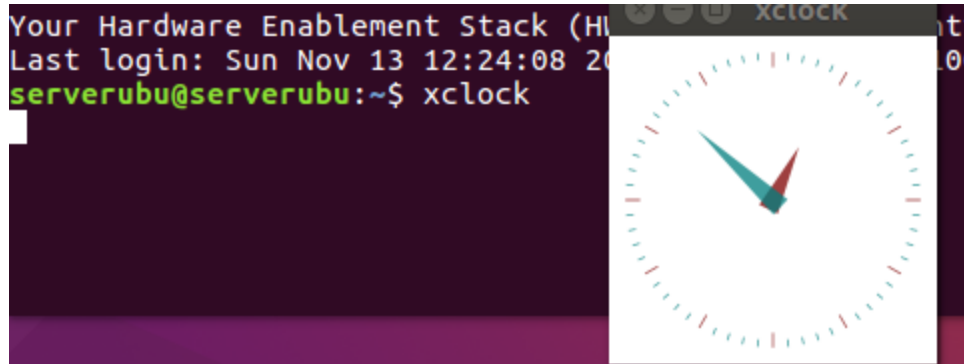
```
serverubu@serverubu:~$ xauth
Using authority file /run/user/1000/gdm/Xauthority
xauth> 
```

Por último reiniciamos el servicio y veremos si está ok:

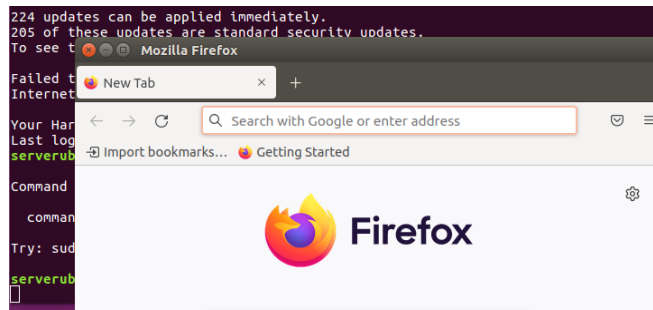
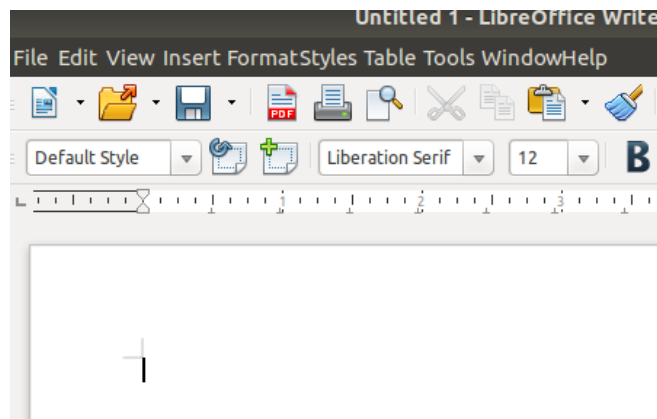
```
serverubu@serverubu:~$ sudo /etc/init.d/ssh restart && /etc/init.d/ssh status
```



Ahora vamos a probar si ha funcionado lo que hemos hecho, iniciaremos sesión en nuestro cliente, nos conectaremos al servidor y abriremos xclock y libreoffice:



Como hemos podido comprobar, ya podemos abrir aplicaciones con interfaz gráfica, por lo que toda la configuración hasta ahora, está funcionando a la perfección.





3. Configura la autenticación de un usuario cliente por sistema de clave RSA al servidor.

Para crear un certificado que permita autenticar al cliente, debemos de ejecutar el siguiente comando en el cliente:

```
cliente@cliente:~$ ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/home/cliente/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/cliente/.ssh/id_rsa.
Your public key has been saved in /home/cliente/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:Y0E4MNYksDQboA0vxorhFxxv4SzUq50bFMTWe5Vqtd2g cliente@cliente
The key's randomart image is:
+---[RSA 2048]---+
|o.=.+...|
|oo * o+.|
|o + ..|
|.O. . .|
|o= + + .S|
|B.+ B =...|
|oB O o o ..|
|o B . o .E .|
| . . . .O .|
+-----[SHA256]-----+
cliente@cliente:~/.ssh$ ls
id_rsa id_rsa.pub known_hosts serverubu@10.10.13.1
```

Se nos habrá guardado en el directorio .ssh, como vemos en la anterior imagen.

Para poder identificarnos en el servidor desde el cliente, debemos copiar el archivo id_rsa.pub que hemos creado en el cliente, lo copiaremos en el servidor dentro de una carpeta llamada '.ssh' en un archivo llamado authorized_keys. Para copiar dicho archivo del cliente al servidor, podemos hacerlo con scp.

```
cliente@cliente:~$ scp /home/cliente/.ssh/id_rsa.pub serverubu@10.10.13.1:~/.ssh
/authorized_keys
serverubu@10.10.13.1's password:
id_rsa.pub 100% 397 0.4KB/s 00:00
```



Ahora probaremos la conexión y veremos si nos pide contraseña:

```
cliente@cliente:~$ sudo ssh serverubu@10.10.13.1
Enter passphrase for key '/root/.ssh/id_rsa':
Welcome to Ubuntu 18.04.6 LTS (GNU/Linux 5.4.0-84-generic x86_64)

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

224 updates can be applied immediately.
205 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

Failed to connect to https://changelogs.ubuntu.com/meta-release-lts. Check your
Internet connection or proxy settings

Your Hardware Enablement Stack (HWE) is supported until April 2023.
Last login: Sun Nov 13 13:08:49 2022 from 10.10.13.100
serverubu@serverubu:~$
```

Ahora solo nos pide la frase que antes configuramos

4. Realiza copia de archivos segura bajo ssh con scp desde cliente al servidor y desde el servidor al cliente.

Para realizar copia de archivos segura desde el cliente al servidor ejecutaremos el siguiente comando:

```
cliente@cliente:~$ sudo scp ejemplo1.txt serverubu@10.10.13.1:~/Desktop
```

Como podemos comprobar en la siguiente imagen, hemos hecho la copia de ese archivo al servidor a la perfección

```
serverubu@serverubu:~/Desktop$ ls
ejemplo1.txt
```



Ahora realizaremos una transferencia del servidor al cliente, con un archivo de ejemplo ("ejemplo2.txt").

```
serverubu@serverubu:~$ scp ejemplo2.txt cliente@10.10.13.100:~/Desktop
cliente@10.10.13.100's password:
ejemplo2.txt                                100%   0   0.0KB/s   00:00
```

Como podemos comprobar en la siguiente imagen, hemos hecho la copia de ese archivo al cliente a la perfección

```
cliente@cliente:~/Desktop$ ls
ejemplo2.txt
```

Aquí termina la práctica 10 de hoy, hemos visto cómo instalar el servidor y cliente, hacer conexión entre ellos, generar una clave privada y otra pública, realizar transferencia de datos y ver cómo todo ha funcionado a la perfección.

Gracias por su tiempo.