

Manual SSH

Álvaro Delgado y Samuel Escudero

27/11/2015

Manual de SSH completado con las competencias aprendidas en el primer trimestre sobre el funcionamiento del protocolo SSH en el Teide IV.

Índice

Ubuntu

Instalación de servidor SSH.	2
Configuración del servidor.	3
Creación clave privada en el servidor.....	4
Primera conexión.....	5
Controlar conexiones al servidor.....	6
Autenticación con clave pública del cliente.	7
Ejecutar aplicaciones gráficas.....	9
Transferencia de archivos sobre SSH.	10
Crear túnel SSH.....	11

Windows

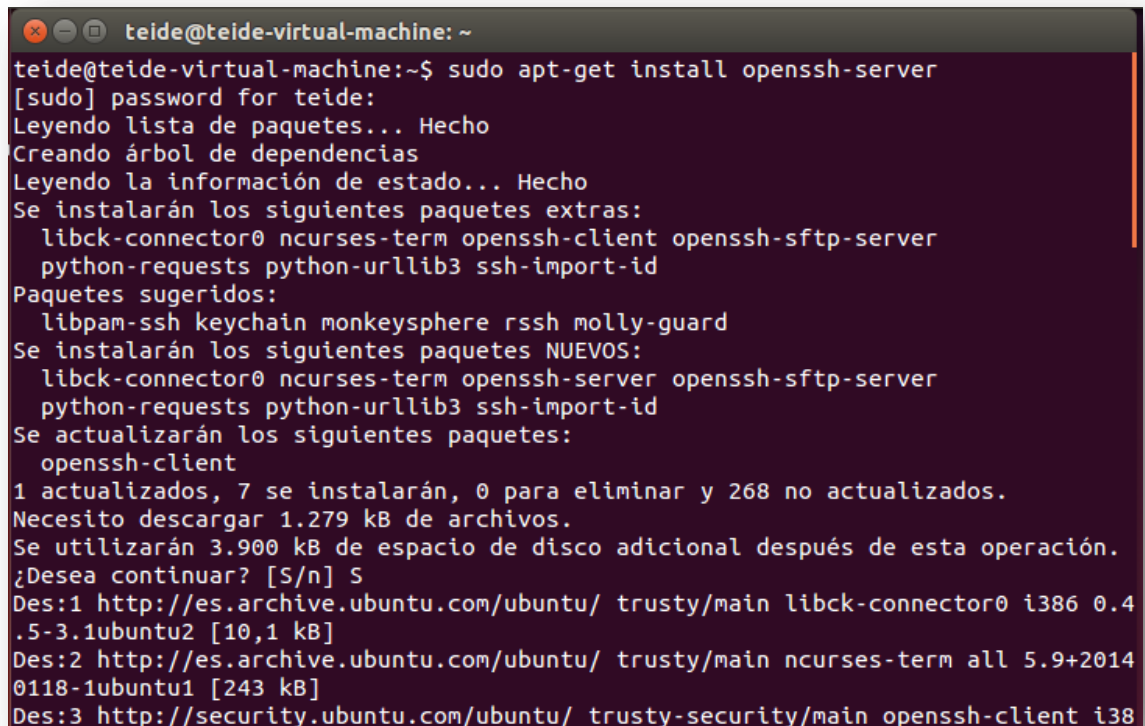
Instalación del cliente SSH.....	12
Primera conexión.....	13
Borrado de las claves públicas de los diferentes servidores.....	14
Crear clave privada.	15
Autenticación mediante clave privada.....	16
Ejecutar aplicaciones gráficas.....	18
Transferencia de archivos sobre SSH.	20
Crear túnel SSH.....	21

Ubuntu

Instalación del servidor SSH

Para instalar el servidor SSH en Ubuntu deberemos ejecutar el comando:

Sudo apt-get install open_ssh-server

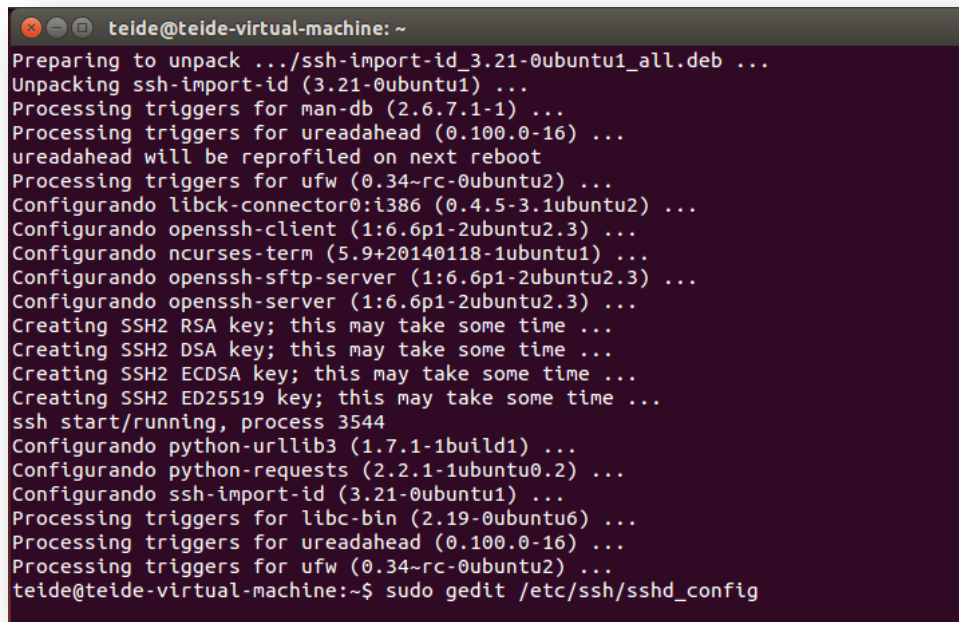
A terminal window titled 'teide@teide-virtual-machine: ~' showing the command 'sudo apt-get install openssh-server' and its output. The output includes package lists, dependencies, and disk space requirements.

```
teide@teide-virtual-machine:~$ sudo apt-get install openssh-server
[sudo] password for teide:
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes extras:
  libck-connector0 ncurses-term openssh-client openssh-sftp-server
  python-requests python-urllib3 ssh-import-id
Paquetes sugeridos:
  libpam-ssh keychain monkeysphere rssh molly-guard
Se instalarán los siguientes paquetes NUEVOS:
  libck-connector0 ncurses-term openssh-server openssh-sftp-server
  python-requests python-urllib3 ssh-import-id
Se actualizarán los siguientes paquetes:
  openssh-client
1 actualizados, 7 se instalarán, 0 para eliminar y 268 no actualizados.
Necesito descargar 1.279 kB de archivos.
Se utilizarán 3.900 kB de espacio de disco adicional después de esta operación.
¿Desea continuar? [S/n] S
Des:1 http://es.archive.ubuntu.com/ubuntu/ trusty/main libck-connector0 i386 0.4
.5-3.1ubuntu2 [10,1 kB]
Des:2 http://es.archive.ubuntu.com/ubuntu/ trusty/main ncurses-term all 5.9+2014
0118-1ubuntu1 [243 kB]
Des:3 http://security.ubuntu.com/ubuntu/ trusty-security/main openssh-client i38
```

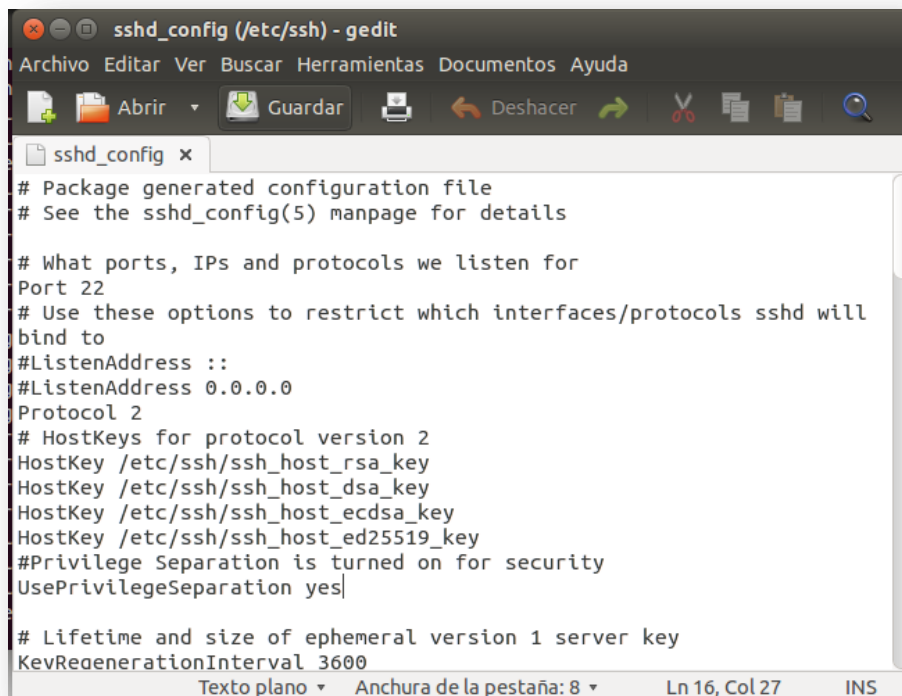
Configuración del servidor

La configuración se almacena en el fichero `sshd_config`, podemos modificarlo con el siguiente commando:

`Sudo gedit /etc/ssh/sshd_config`



```
teide@teide-virtual-machine: ~
Preparing to unpack .../ssh-import-id_3.21-0ubuntu1_all.deb ...
Unpacking ssh-import-id (3.21-0ubuntu1) ...
Processing triggers for man-db (2.6.7.1-1) ...
Processing triggers for ureadahead (0.100.0-16) ...
ureadahead will be reprofiled on next reboot
Processing triggers for ufw (0.34-rc-0ubuntu2) ...
Configurando libck-connector0:i386 (0.4.5-3.1ubuntu2) ...
Configurando openssh-client (1:6.6p1-2ubuntu2.3) ...
Configurando ncurses-term (5.9+20140118-1ubuntu1) ...
Configurando openssh-sftp-server (1:6.6p1-2ubuntu2.3) ...
Configurando openssh-server (1:6.6p1-2ubuntu2.3) ...
Creating SSH2 RSA key; this may take some time ...
Creating SSH2 DSA key; this may take some time ...
Creating SSH2 ECDSA key; this may take some time ...
Creating SSH2 ED25519 key; this may take some time ...
ssh start/running, process 3544
Configurando python-urllib3 (1.7.1-1build1) ...
Configurando python-requests (2.2.1-1ubuntu0.2) ...
Configurando ssh-import-id (3.21-0ubuntu1) ...
Processing triggers for libc-bin (2.19-0ubuntu6) ...
Processing triggers for ureadahead (0.100.0-16) ...
Processing triggers for ufw (0.34-rc-0ubuntu2) ...
teide@teide-virtual-machine:~$ sudo gedit /etc/ssh/sshd_config
```



```
sshd_config (/etc/ssh) - gedit
Archivo Editar Ver Buscar Herramientas Documentos Ayuda
Abrir Guardar Deshacer
sshd_config x
# Package generated configuration file
# See the sshd_config(5) manpage for details

# What ports, IPs and protocols we listen for
Port 22
# Use these options to restrict which interfaces/protocols sshd will
bind to
#ListenAddress ::
#ListenAddress 0.0.0.0
Protocol 2
# HostKeys for protocol version 2
HostKey /etc/ssh/ssh_host_rsa_key
HostKey /etc/ssh/ssh_host_dsa_key
HostKey /etc/ssh/ssh_host_ecdsa_key
HostKey /etc/ssh/ssh_host_ed25519_key
#Privilege Separation is turned on for security
UsePrivilegeSeparation yes

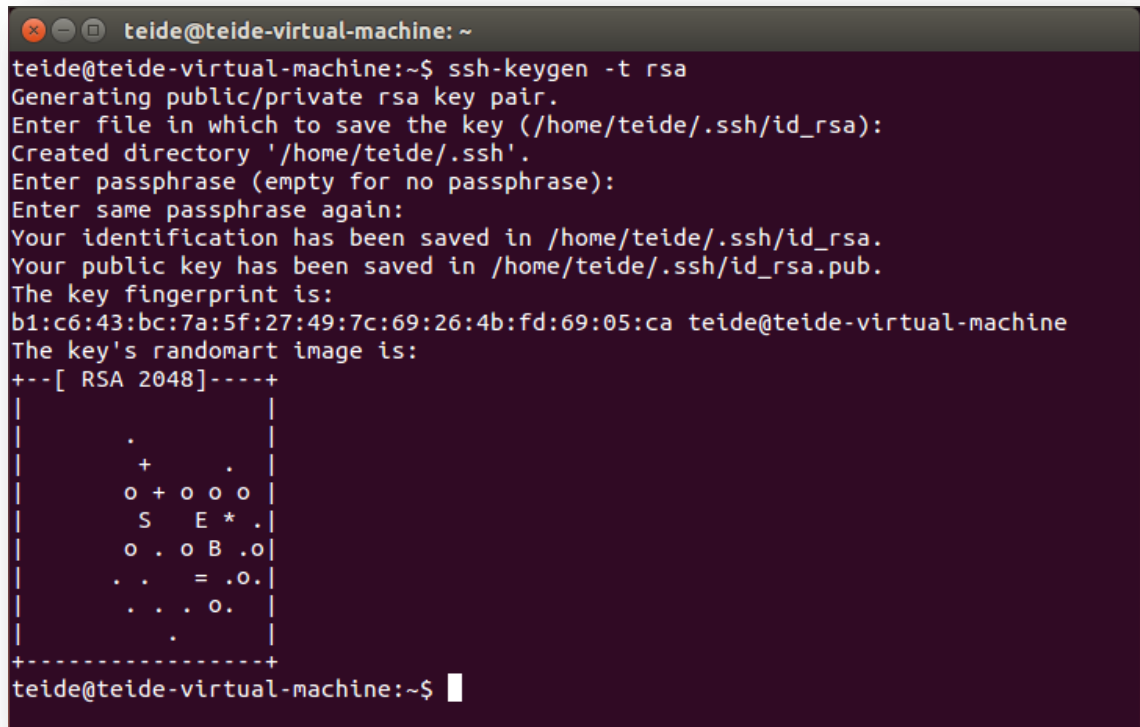
# Lifetime and size of ephemeral version 1 server key
KevRegenerationInterval 3600
Texto plano Anchura de la pestaña: 8 Ln 16, Col 27 INS
```

Creación de la clave privada en el servidor

Para generar el par de clave utilizamos el comando:

```
sudo ssh-keygen -t rsa
```

Guardara las claves generadas en ~/.ssh/



```
teide@teide-virtual-machine: ~
teide@teide-virtual-machine:~$ ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/home/teide/.ssh/id_rsa):
Created directory '/home/teide/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/teide/.ssh/id_rsa.
Your public key has been saved in /home/teide/.ssh/id_rsa.pub.
The key fingerprint is:
b1:c6:43:bc:7a:5f:27:49:7c:69:26:4b:fd:69:05:ca teide@teide-virtual-machine
The key's randomart image is:
+--[ RSA 2048 ]-----+
|          .           |
|        + + .        |
|       o + o o o     |
|      S   E * .      |
|     o . o B .o     |
|    . .   = .o.     |
|   . . . o.         |
|          .          |
+-----+
teide@teide-virtual-machine:~$
```

Primera conexión

Para conectarnos con el cliente ejecutamos el comando:

```
ssh <ip> -l <usuario>
```

Nos pedirá la clave y la introducimos.

```
teide@teide-virtual-machine:~$ ssh 127.0.0.1
The authenticity of host '127.0.0.1 (127.0.0.1)' can't be established.
ECDSA key fingerprint is 4e:6f:3a:f5:3f:64:d1:2e:e3:a9:79:a1:7a:fe:07:44.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '127.0.0.1' (ECDSA) to the list of known hosts.
teide@127.0.0.1's password:
Welcome to Ubuntu 14.04 LTS (GNU/Linux 3.13.0-24-generic i686)

 * Documentation:  https://help.ubuntu.com/

0 packages can be updated.
0 updates are security updates.

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.

teide@teide-virtual-machine:~$ exit
```

Controlar las conexiones al servidor

Para ver que usuarios están conectados ejecutamos el comando:

Who

Las conexiones externas serían las pts diferentes de 0, en este caso pts/12 si queremos cerrar esa sesión ejecutamos:

Sudo pkill -t pts/12.

```
teide@teide-virtual-machine:~$ who
teide    :0                2015-11-27 16:05 (:0)
teide    pts/0            2015-11-27 16:18 (:0)
teide    pts/12           2015-11-27 16:25 (localhost)
teide    pts/14           2015-11-27 16:29 (:0)
teide@teide-virtual-machine:~$ sudo pkill -t pts/12
[sudo] password for teide:
teide@teide-virtual-machine:~$ who
teide    :0                2015-11-27 16:05 (:0)
teide    pts/0            2015-11-27 16:18 (:0)
teide    pts/14           2015-11-27 16:29 (:0)
teide@teide-virtual-machine:~$
```

Autenticación con clave pública del cliente.

Lo primero que debemos hacer es generar el par de claves en el cliente con el comando :

```
Sudo ssh-keygen -t rsa
```

Una vez nos las ha creado las exportamos al servidor mediante el comando:

```
ssh-copy-ip <clave pública>@<ip_servidor>
```

Otra opción es copiarla con un pen drive y copiar su contenido a ~/.ssh/authorized keys.

Ahora realizamos la conexión:

```
ssh <usuario>@<servidor> -i <clave_privada>
```



```
root@teide-virtual-machine: /home/teide/ssh
root@teide-virtual-machine:/home/teide/.ssh# ssh-copy-id -i id_rsa.pub teide@127.0.0.1
The authenticity of host '127.0.0.1 (127.0.0.1)' can't be established.
ECDSA key fingerprint is 4e:6f:3a:f5:3f:64:d1:2e:e3:a9:79:a1:7a:fe:07:44.
Are you sure you want to continue connecting (yes/no)? 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
teide@127.0.0.1's password:

Number of key(s) added: 1

Now try logging into the machine, with:  "ssh 'teide@127.0.0.1'"
and check to make sure that only the key(s) you wanted were added.

root@teide-virtual-machine:/home/teide/.ssh#
```

```
teide@teide-virtual-machine: ~
teide@teide-virtual-machine: ~/.ssh$ ssh teide@127.0.0.1 -i id_rsa
Welcome to Ubuntu 14.04 LTS (GNU/Linux 3.13.0-24-generic i686)

* Documentation:  https://help.ubuntu.com/

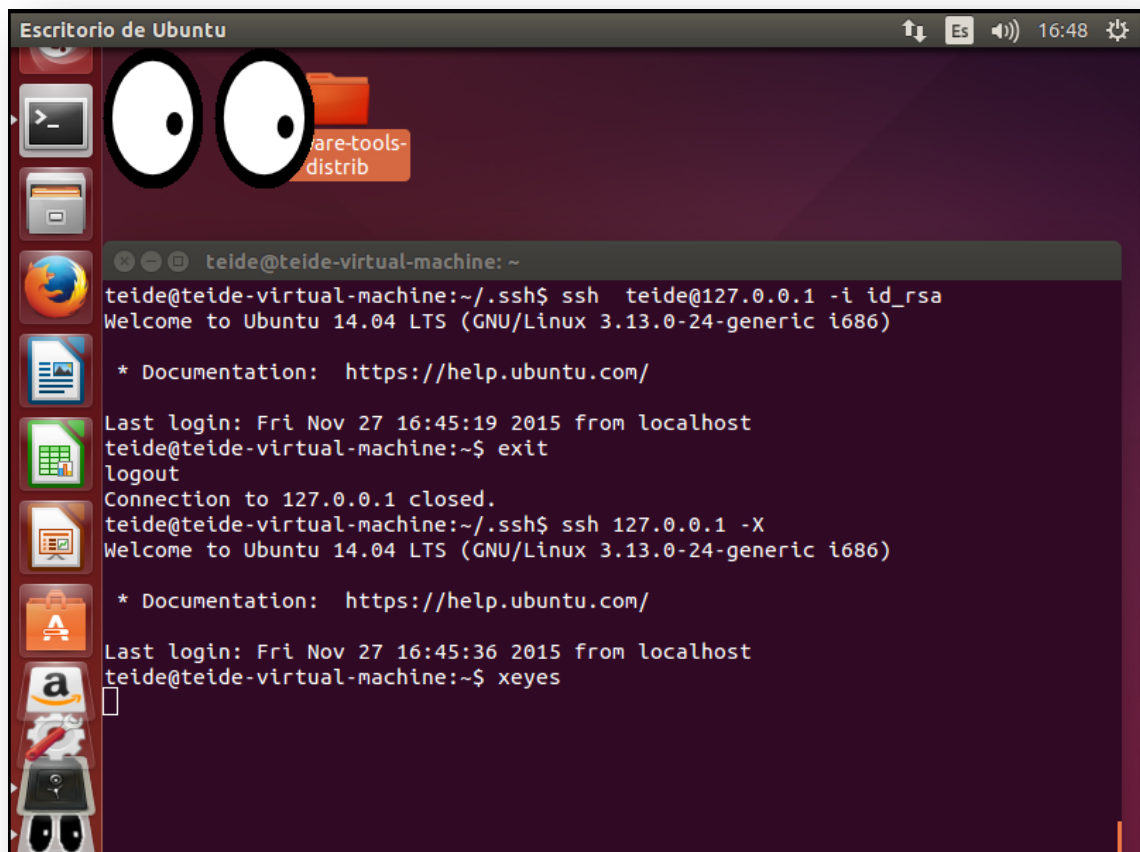
Last login: Fri Nov 27 16:45:19 2015 from localhost
teide@teide-virtual-machine:~$
```

Ejecutar aplicaciones gráficas

Podemos ejecutar aplicaciones gráficas al conectarnos por ssh usando la variable -X.

```
ssh <usuario>@<ip_servidor> -X
```

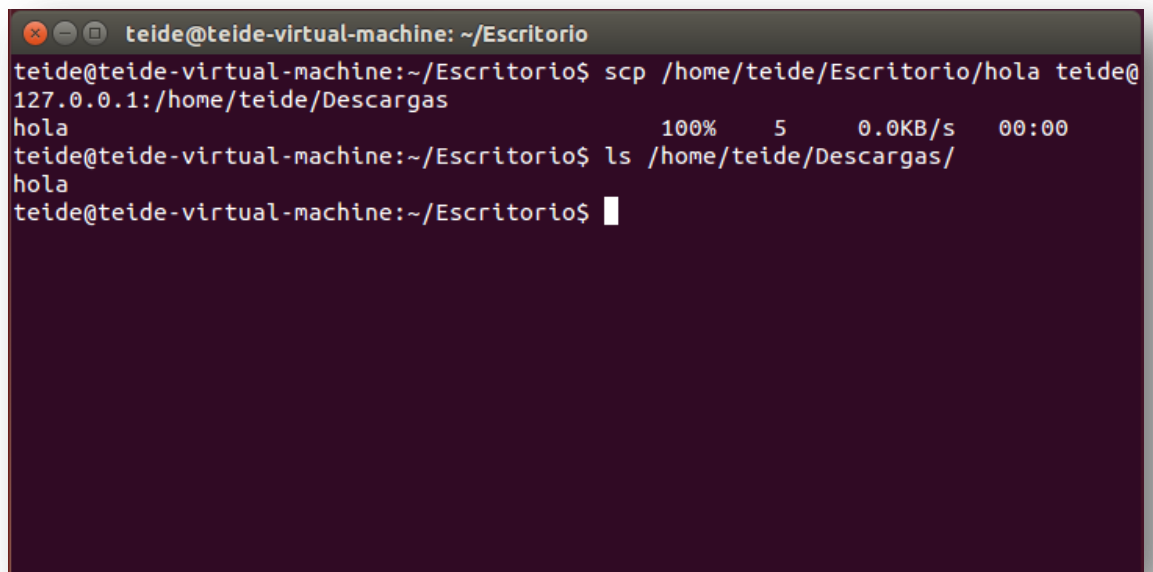
Ejecutamos xeyes



Transferencia de archivos sobre SSH

Para copiar archivos entre el cliente y el servidor ejecutamos el comando scp:

scp <ruta_fichero_local> <usuario>@<ip_servidor>:<ruta_destino_remoto>

A terminal window titled 'teide@teide-virtual-machine: ~/Escritorio' shows the execution of the scp command. The command 'scp /home/teide/Escritorio/hola teide@127.0.0.1:/home/teide/Descargas' is entered. The output shows the file 'hola' being transferred at 100% completion, with a size of 5 bytes and a speed of 0.0KB/s. The user then runs 'ls /home/teide/Descargas/' which lists the file 'hola'.

```
teide@teide-virtual-machine: ~/Escritorio
teide@teide-virtual-machine:~/Escritorio$ scp /home/teide/Escritorio/hola teide@
127.0.0.1:/home/teide/Descargas
hola                               100%    5    0.0KB/s   00:00
teide@teide-virtual-machine:~/Escritorio$ ls /home/teide/Descargas/
hola
teide@teide-virtual-machine:~/Escritorio$
```

Crear túnel SSH

Podemos crear un túnel ssh mediante el comando:

```
ssh -L <puerto_local>:<localhost>:<puerto_remoto> <usuario>@<ip_servidor>
```

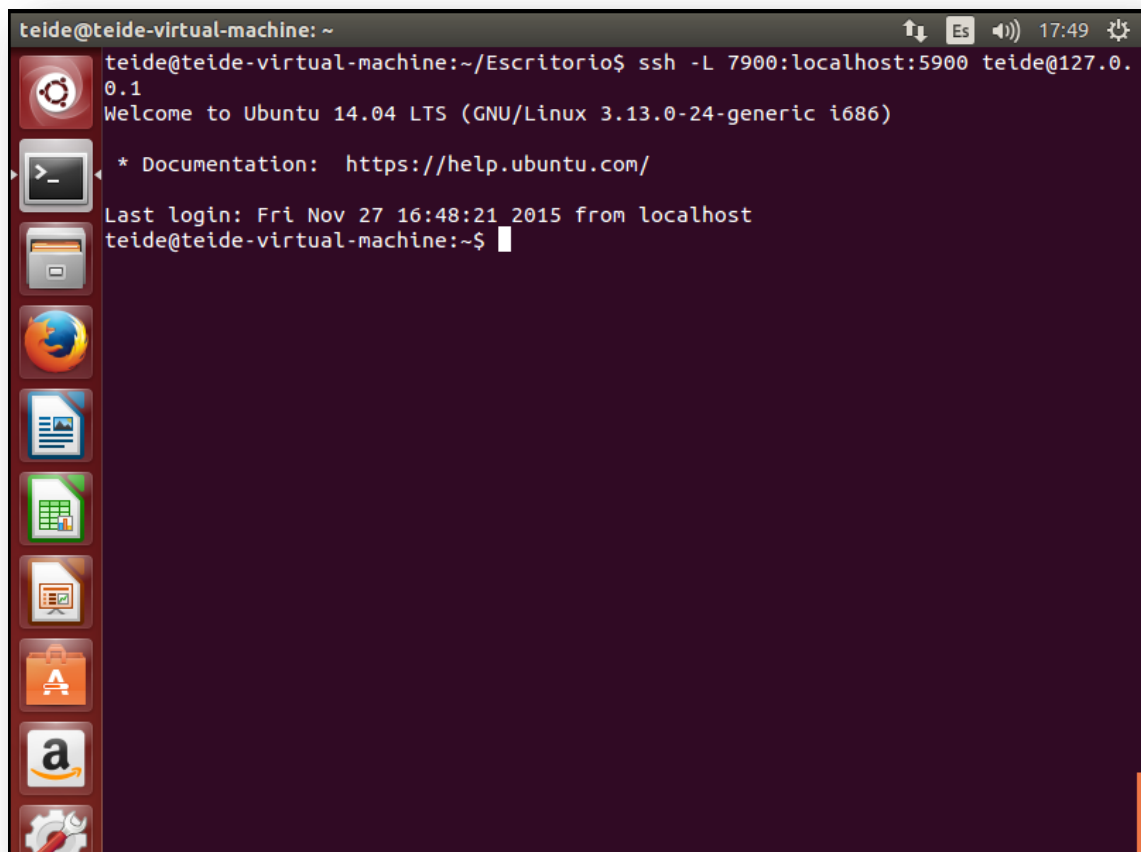
Para usarlo debemos redirigir el tráfico de la aplicación a nuestro puerto local:

Es un comando muy útil dado que nos permite cifrar el tráfico de un puerto ajeno a ssh.

Ejemplo:

```
ssh -L 7900:interno2:5900 usuario@interno1
```

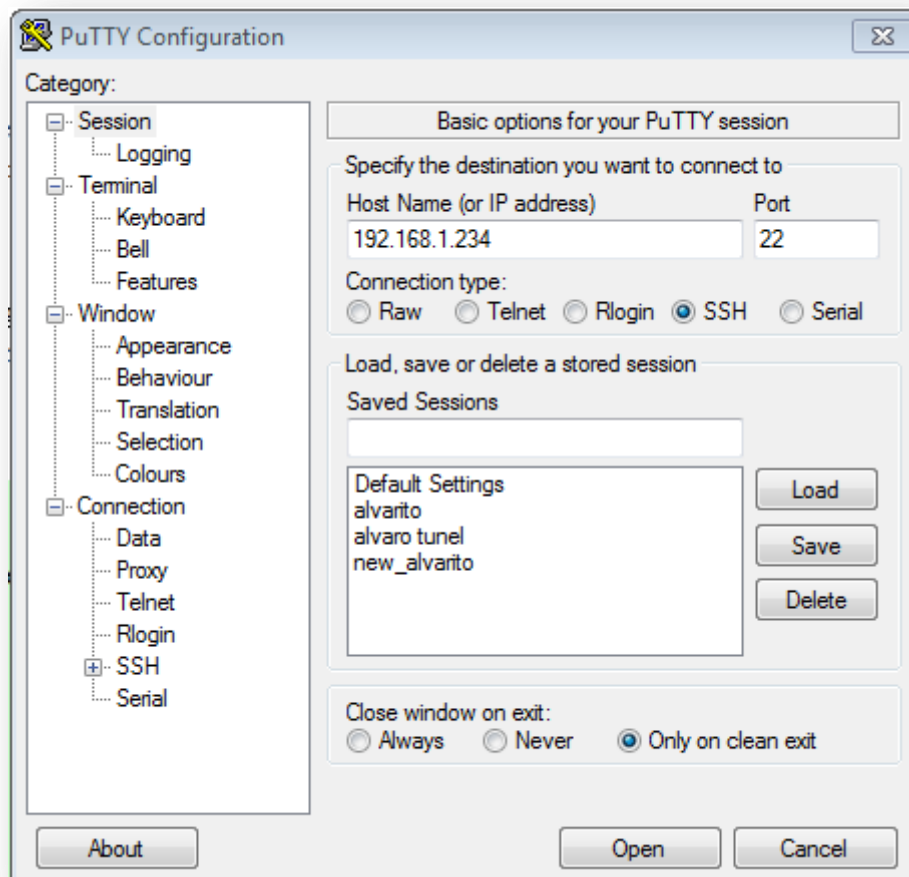
```
vncviewer localhost::7900
```



Windows

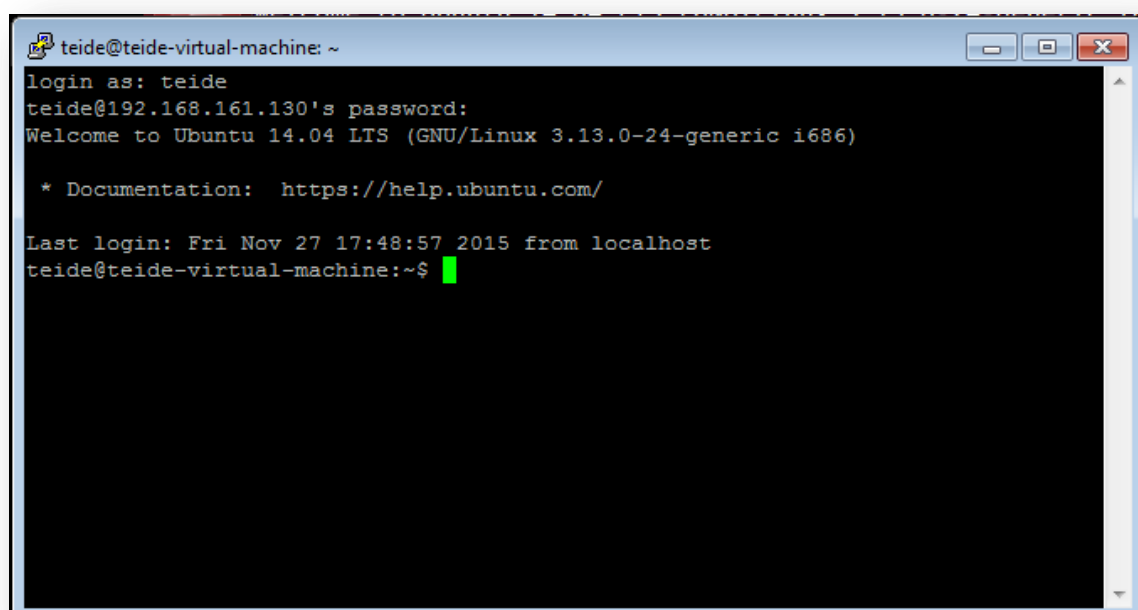
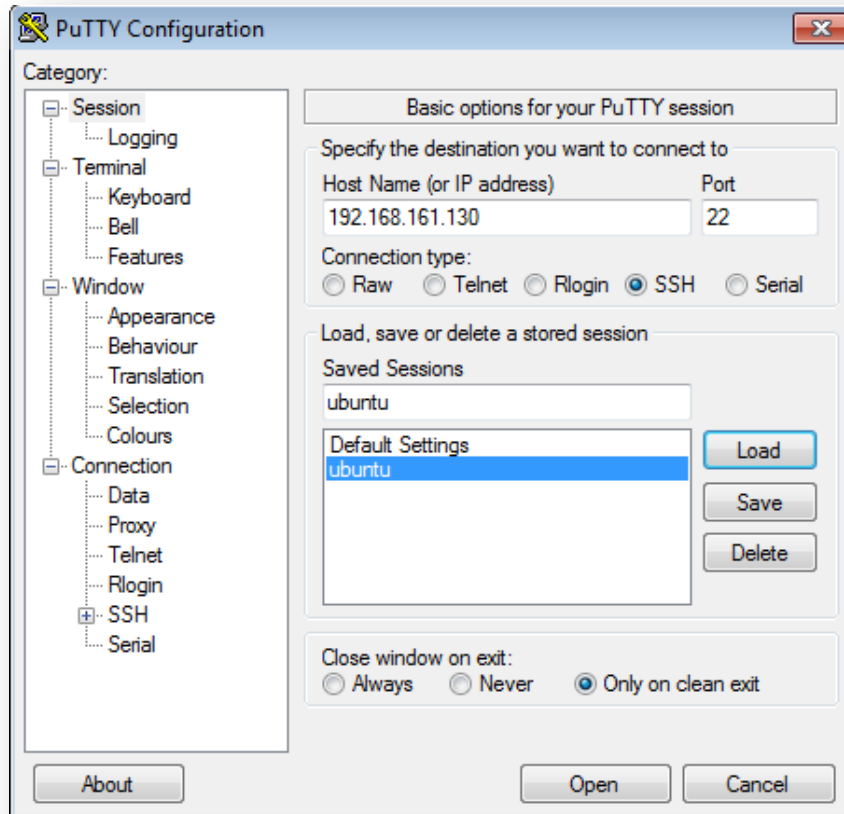
Instalación del cliente SSH.

El cliente que usamos en Windows Putty, no necesita instalación, lo ejecutamos veremos la siguiente ventana:



Primera conexión.

Introducimos la ip del servidor y le damos a conectar:

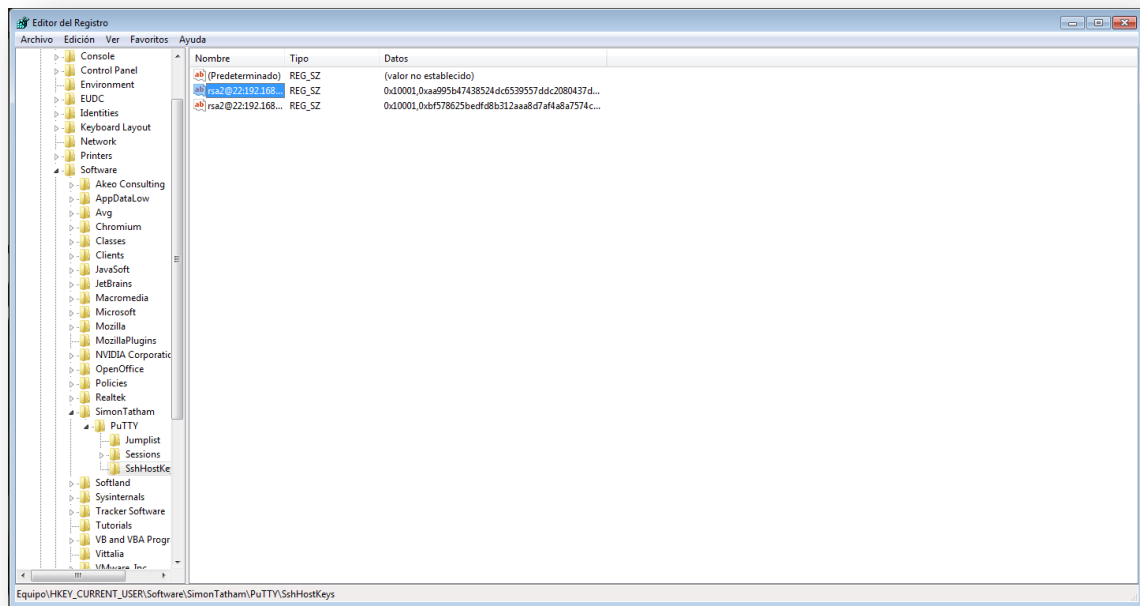


Borrado de las claves públicas de los diferentes servidores.

Abrimos regedit y seguimos la siguiente ruta:

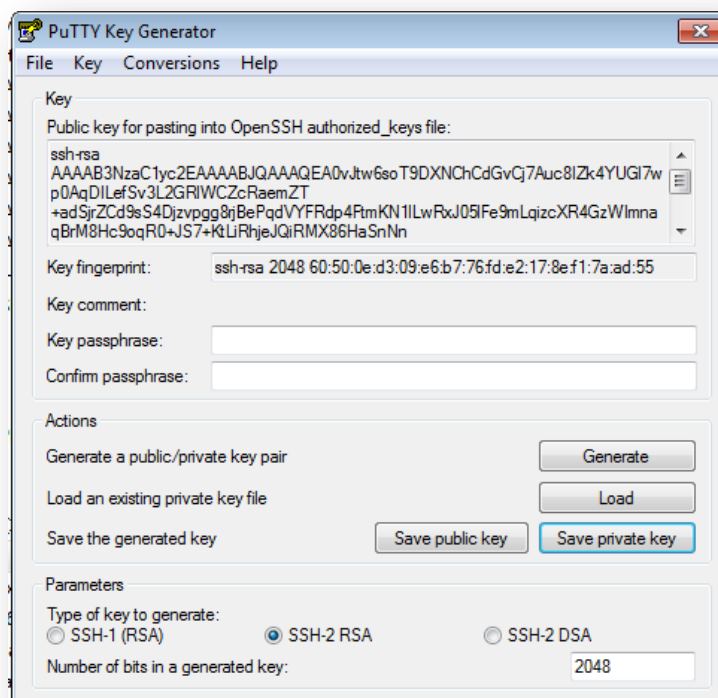
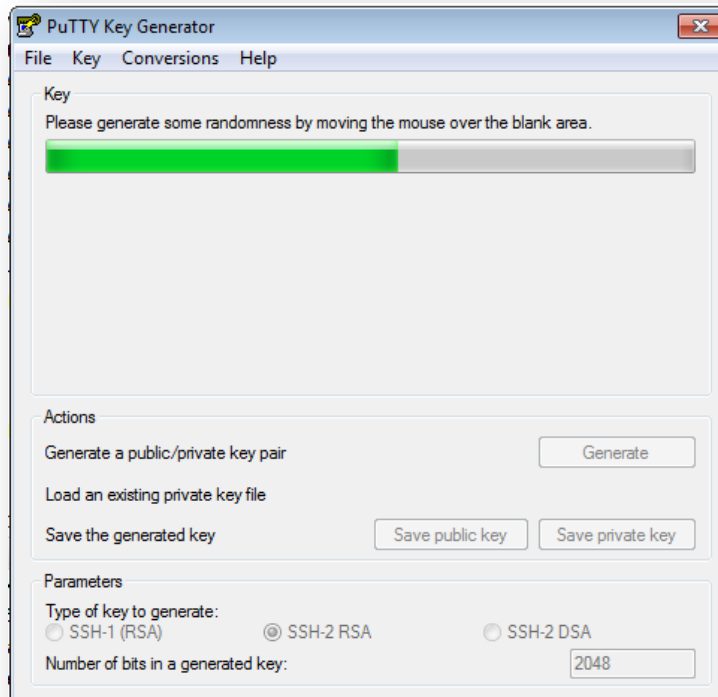
Equipo\HKEY_CURRENTUSER\Software\SimonThatam\Putty\sshHostKeys

Aquí borramos las rsa2 que queramos.



Crear clave privada

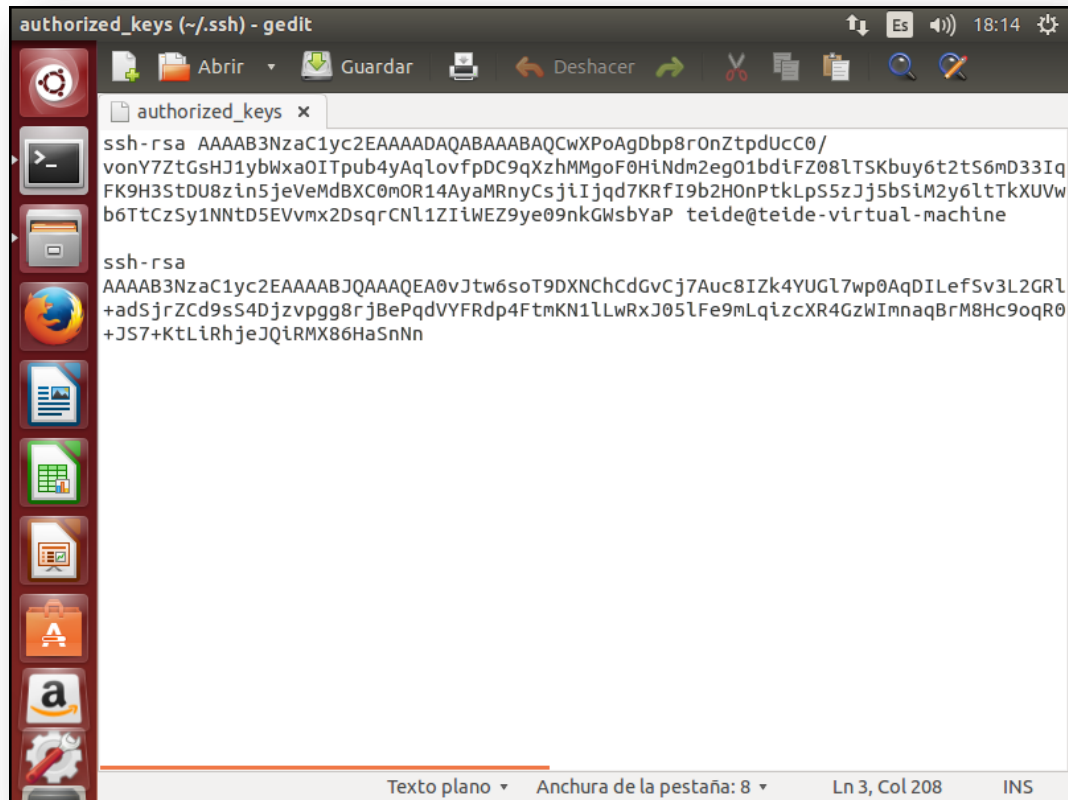
Deberemos descargarnos el PuttyGen y lo ejecutamos.



Autenticación mediante clave privada.

Debes copiar el texto obtenido al crear la clave privada y copiarlo en `~/ssh/authorized_keys`

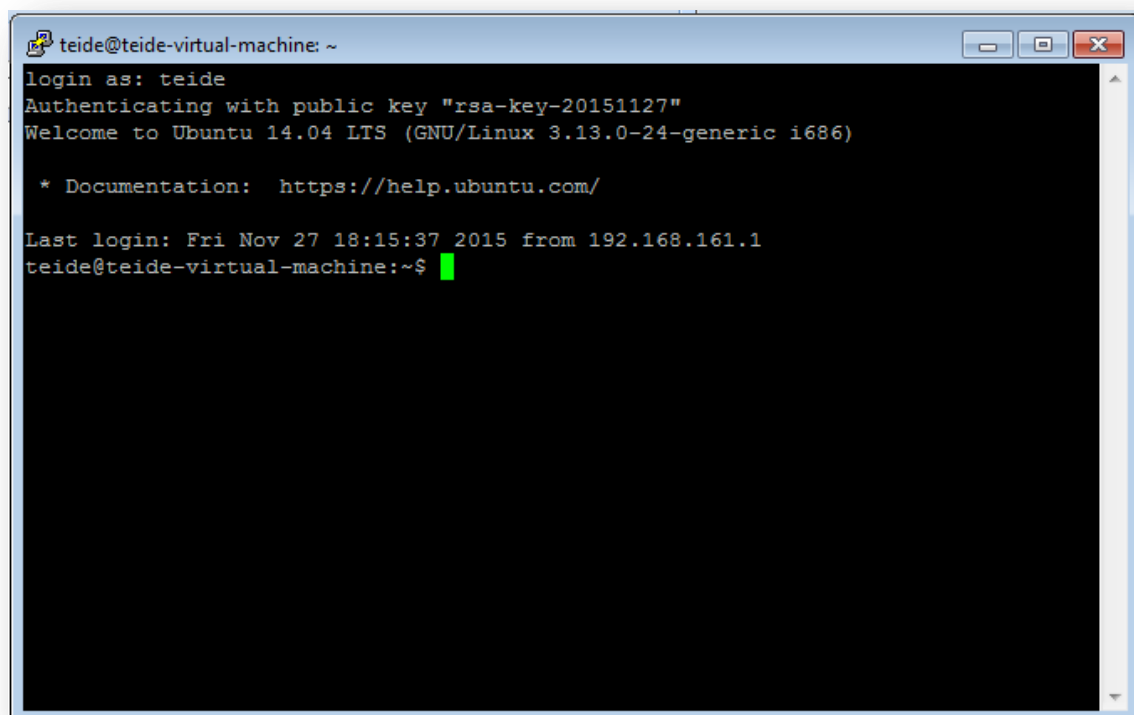
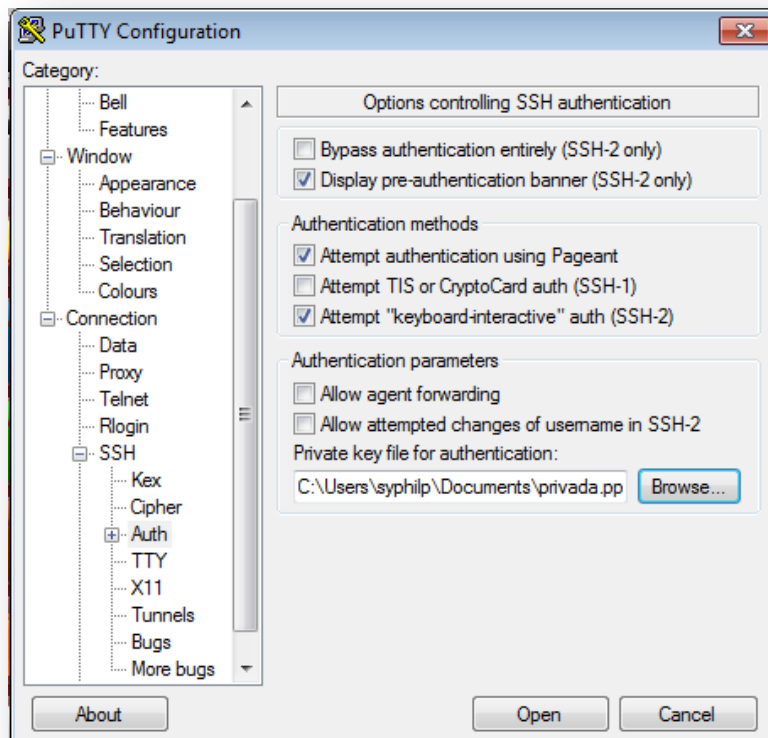
Vamos al Putty > ssh > Auth y cargamos nuestra clave privada, le damos a conectar, nos pedirá la passphrase de la clave privada.



The screenshot shows a gedit window titled "authorized_keys (~/.ssh) - gedit". The window contains two entries for SSH keys. The first entry is an ssh-rsa key with a long base64-encoded string, followed by the comment "teide@teide-virtual-machine". The second entry is another ssh-rsa key with a long base64-encoded string. The window has a standard Ubuntu-style interface with a sidebar on the left showing icons for various applications. The status bar at the bottom indicates "Texto plano", "Anchura de la pestaña: 8", "Ln 3, Col 208", and "INS".

```
authorized_keys x
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQAwXAgDbb8rOnZtpdUcC0/
vonY7ZtGsHJ1ybWxa0ITpub4yAqlvfpDC9qXzhMMgoF0HiNdm2eg01bdiFZ08lTSKbuy6t2tS6mD33Iq
FK9H3StDU8zln5jeVeMdBXC0mOR14AyaMRnyCsjiIjqd7KRfI9b2H0nPtKlpS5zJj5bSiM2y6ltTkXUVw
b6TtCzSy1NNtD5EVvmx2DsqrCNl1ZIiWEZ9ye09nkGwsbyaP teide@teide-virtual-machine

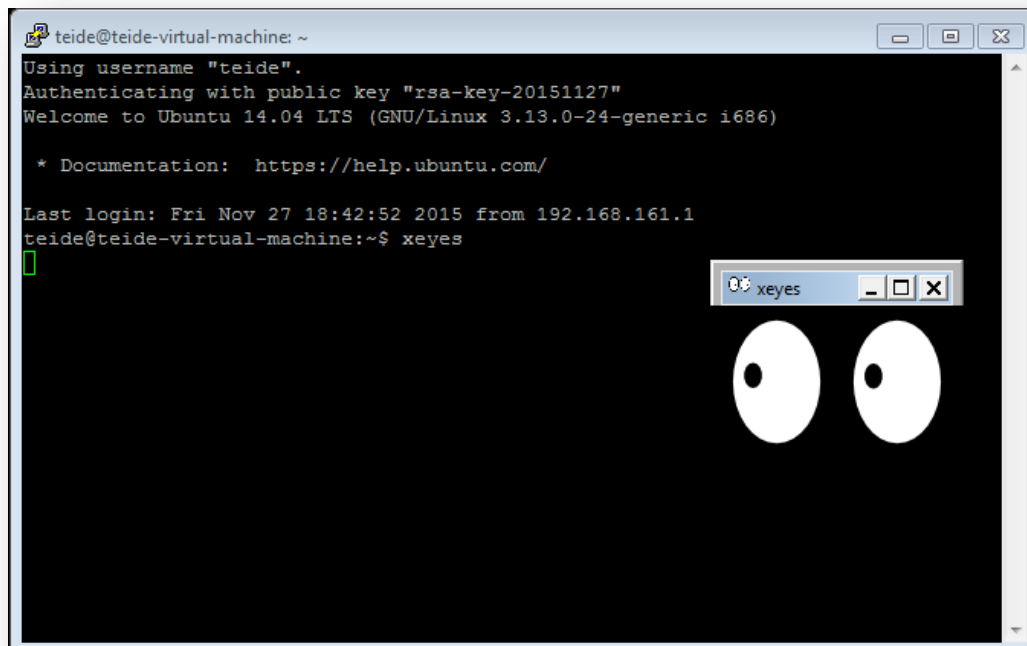
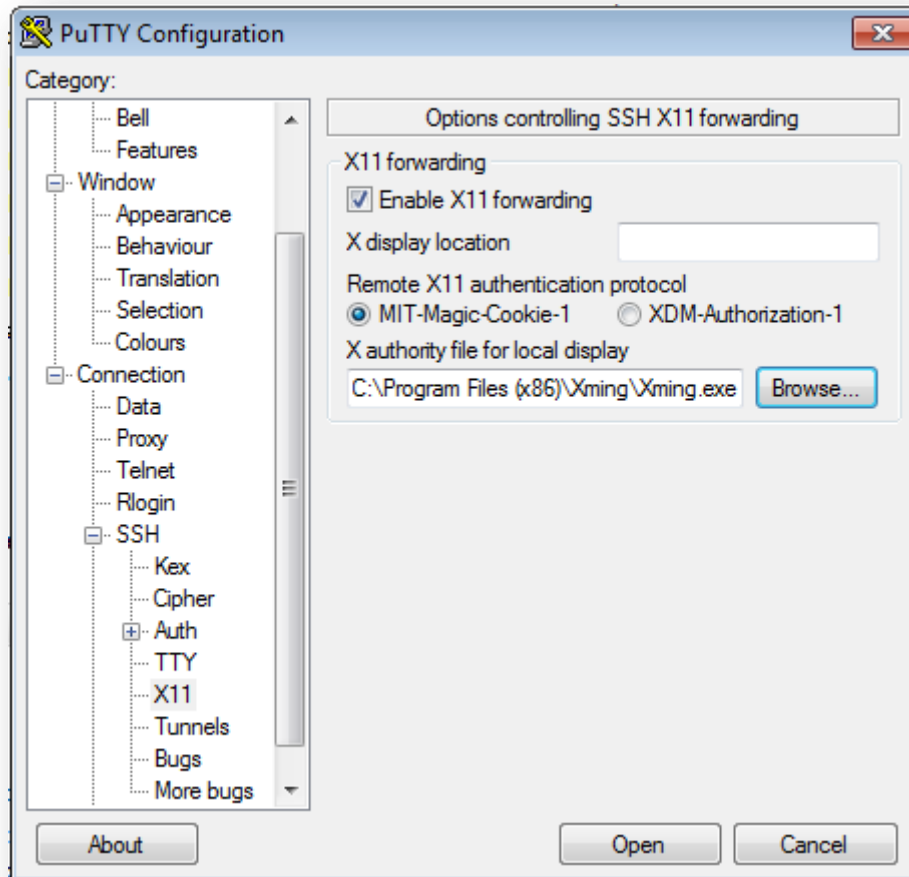
ssh-rsa
AAAAB3NzaC1yc2EAAAABJQAAQEA0vJtw6soT9DXNChCdGvcj7Auc8IZk4YUGl7wp0AqDIlefSv3L2GRl
+adSjrZCd9sS4Djzvpgg8rjBePqdVYFRdp4FtmKN1LLwRxJ05lFe9mLqizcXR4GzWImnaqBrM8Hc9oqR0
+JS7+KtLiRhjeJQiRMX86HaSnNn
```



Ejecutar aplicaciones gráficas

Deberemos instalar el programa xming y ejecutar putty con la opción `ssh > x11 > enable x11` activada. Y le damos a conectar.

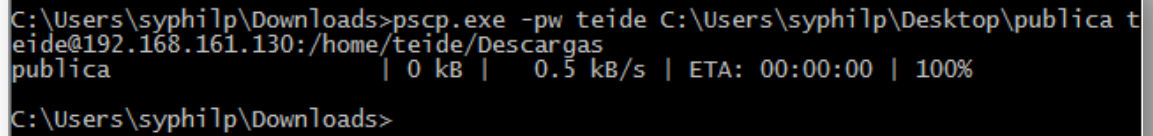




Transferencia de archivos sobre ssh

Desde cmd ejecutamos el comando:

`pscp.exe -pw <usuario> <archivo_local> <usuario>@<ip_server>:<archivo_remoto>`



```
C:\Users\syphilp\Downloads>pscp.exe -pw teide C:\Users\syphilp\Desktop\publica t
eide@192.168.161.130:/home/teide/Descargas
publica | 0 kB | 0.5 kB/s | ETA: 00:00:00 | 100%
C:\Users\syphilp\Downloads>
```

Crear túnel ssh

En el Putty vamos a SSH > Tunnels. En source port marcamos nuestro puerto local y en Destination ponemos <ip_remota>:<Puerto_remoto>. Una vez realicemos la conexión ejecutamos daytime desde cmd usando el puerto local.

