Linux Administration

Connecting to a Linux server using SSH

Xavier Belanger

What are SSH and OpenSSH?

- SSH stands for “Secure Shell”; it’s a network protocol created in 1995 by Tatu Ylönen, that has been standardized later on.
- OpenSSH is a free and open source implementation of that protocol. It originates from the OpenBSD project, and is now ported to various systems, and usually available by default on various Linux distributions.

How to connect to a server?

- Launch the SSH client
- Authenticate with a username and password, or a key.
- At the very first connection, validate the server fingerprint.
- Pay attention to any unexpected fingerprint change in the future.

Connecting with a password

- Using a password usually don’t require any special configuration, as long as the account is valid.
- The password will be required at each connection.

Connecting with a key

- Using a key requires an initial setup (generating and copying the key) and will simplify future connections.
- The benefit of using a key is that no password will be sent over the network.

Generating a key

- It is recommended to use different keys for different systems.
- You must protect your key with a passphase. Anyone getting a copy could use it if it is not protected.

Generating a key - Example

```
xavier@linux:~$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/xavier/.ssh/id_rsa): /home/xavier/.ssh/test-key
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/xavier/.ssh/test-key.
Your public key has been saved in /home/xavier/.ssh/test-key.pub.
The key fingerprint is:
SHA256:QEhUwmD93qWUCBUJTWQCMj8c3HP6NaEoW55uAlihBow xavier@linux.home.arpa
The key's randomart image is:
+---[RSA 2048]----+
|=.=O*OOo         |
|EBo.B=+ .        |
|..+. B.o o       |
|..o.+ +.= .      |
|o. = + +S+       |
|o . o o o        |
| . .             |
|  . o            |
|   o             |
+----[SHA256]-----+
xavier@linux:~$
```

Copying a key

- By default, SSH key are located in the *.ssh/authorized_keys* file in the user's home directory.
- Multiple keys can be stored, one per line.
- Permissions on the file and the directory are critical.

Copying a key - Example

```
xavier@linux:~$ ssh-copy-id -i /home/xavier/.ssh/test-key server.home.arpa
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/xavier/.ssh/test-key.pub"
/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


Number of key(s) added: 1


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


xavier@linux:~$
```

Copying files

- Files can be copied between two systems by using *scp* or *sftp*.
- sftp is similar to the old ftp command, and designed to be used in an interactive way.
- Both commands can benefit of using SSH keys.

Running commands remotely

- It is possible to run a command and get the result remotely over SSH.
- This will no work for commands requiring interaction or that would require their own terminal.

Advanced usages

- OpenSSH tunnels can be used to carry other network traffic, usually insecure ones.
- You can set one SSH server as a central point to connect to other systems (still via SSH) in an automatic fashion (sometimes knowns as “jump server” or “bastion host”).