
ATTACK
DEFENSE
by PentesterAcademy

Name	Recon: MSSQL: Nmap Scripts
URL	https://attackdefense.com/challengedetails?cid=2316
Type	Windows Recon: MSSQL

Important Note: This document illustrates all the important steps required to complete this lab. This is by no means a comprehensive step-by-step solution for this exercise. This is only provided as a reference to various commands needed to complete this exercise and for your further research on this topic. Also, note that the IP addresses and domain names might be different in your lab.

Step 1: Checking the IP address.

Command: ipconfig

```
Administrator: Windows PowerShell
PS C:\Users\Administrator> ipconfig

Windows IP Configuration


Ethernet adapter Ethernet:

   Connection-specific DNS Suffix  . : ap-southeast-1.compute.internal
   Link-local IPv6 Address . . . . . : fe80::e5fc:db45:ed9e:4e89%4
   IPv4 Address. . . . . . . . . . . : 10.0.26.149
   Subnet Mask . . . . . . . . . . . : 255.255.240.0
   Default Gateway . . . . . . . . . : 10.0.16.1
PS C:\Users\Administrator>
```

Step 2: Run Nmap scan against the subnet to discover the target machine's IP address.

Command: nmap 10.0.26.0/20 --open

Note: Nmap '**--open**' option would show only exposed ports of the live hosts.

```
PS C:\Users\Administrator> nmap 10.0.26.0/20 --open
Starting Nmap 7.91 ( https://nmap.org ) at 2021-04-03 13:08 Coordinated Universal Time
Nmap scan report for ip-10-0-18-201.ap-southeast-1.compute.internal (10.0.18.201)
Host is up (0.00s latency).
Not shown: 995 closed ports
PORT     STATE SERVICE
135/tcp  open  msrpc
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
1433/tcp open  ms-sql-s
3389/tcp open  ms-wbt-server
MAC Address: 06:8D:EF:39:4D:A6 (Unknown)

Nmap scan report for ip-10-0-26-149.ap-southeast-1.compute.internal (10.0.26.149)
Host is up (0.00s latency).
Not shown: 996 closed ports
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: 4096 IP addresses (8 hosts up) scanned in 33.31 seconds
PS C:\Users\Administrator>
```

We have discovered the target machine's IP address (**10.0.18.201**) and the target machine exposed to multiple ports. We can notice MSSQL port 1433 is also exposed.

Step 3: Checking MSSQL target server information.

Running ms-sql-info Nmap script to discover MSSQL server information.

Command: nmap --script ms-sql-info -p 1433 10.0.18.201

```
PS C:\Users\Administrator> nmap --script ms-sql-info -p 1433 10.0.18.201
Starting Nmap 7.91 ( https://nmap.org ) at 2021-04-03 13:10 Coordinated Universal Time
Nmap scan report for ip-10-0-18-201.ap-southeast-1.compute.internal (10.0.18.201)
Host is up (0.00s latency).

PORT     STATE SERVICE
1433/tcp open  ms-sql-s
MAC Address: 06:8D:EF:39:4D:A6 (Unknown)

Host script results:
| ms-sql-info:
|   10.0.18.201:1433:
|     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

Nmap done: 1 IP address (1 host up) scanned in 0.44 seconds
PS C:\Users\Administrator>
```

We have found that the target is running "**Microsoft SQL Server 2019**".

Step 4: Connect to the MSSQL using sqlcmd utility with provided credentials i.e **admin:anamaria**

About SQLCMD:

The sqlcmd utility lets you enter Transact-SQL statements, system procedures, and script files through a variety of available modes:

- At the command prompt.
- In Query Editor in SQLCMD mode.
- In a Windows script file.
- In an operating system (Cmd.exe) job step of a SQL Server Agent job.
-

The utility uses ODBC to execute Transact-SQL batches.

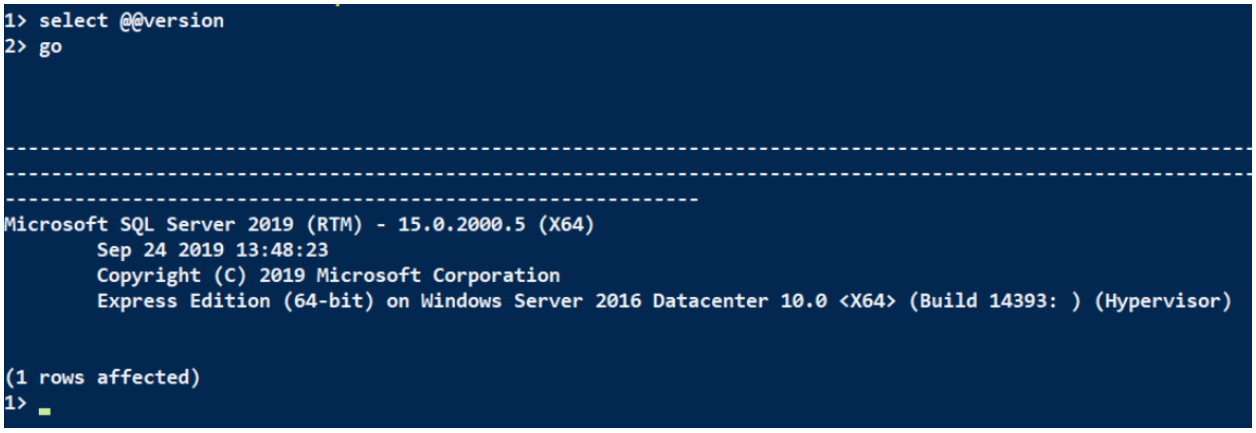
Source: https://docs.microsoft.com/en-us/sql/tools/sqlcmd-utility?view=sql-server-ver15

Command: sqlcmd -S 10.0.18.201 -U admin -P anamaria

We are connected with the target MSSQL server using provided credentials.

Step 5: Checking the version

Command: select @@version
go



Step 6: Determine current database

Command: select db_name();
go

```
1> select db_name();
2> go


------------------------------------
--------
master


(1 rows affected)
1> _
```

The current database is master.

Step 7: Discover the target machine hostname.

Command: SELECT HOST_NAME();
go

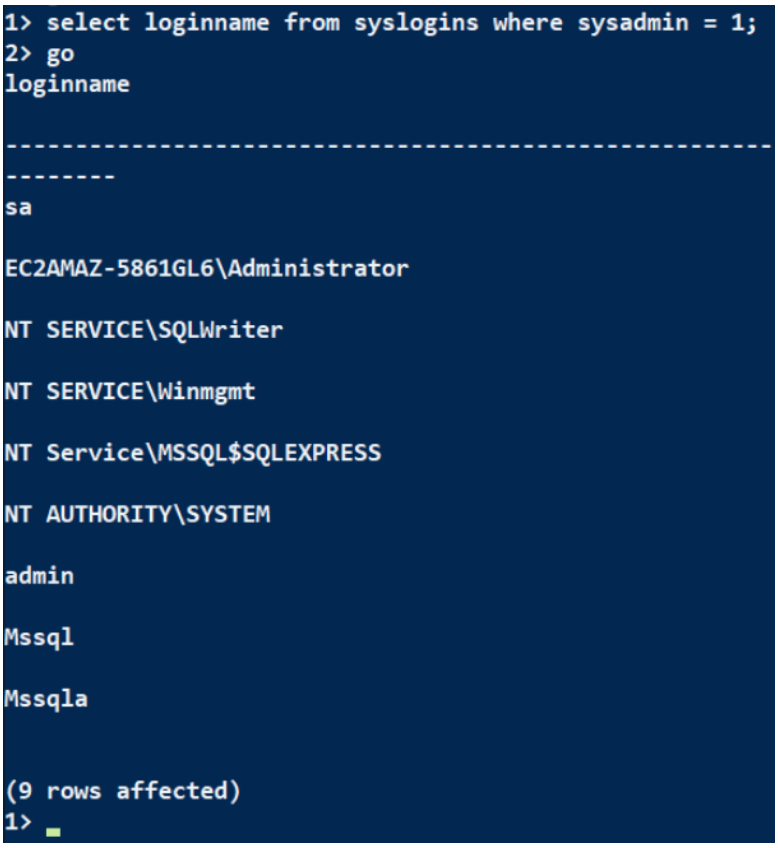
```
1> SELECT HOST_NAME();
2> go


----------------------
--------
MSSQL-CLIENT


(1 rows affected)
1>
```

Step 8: Determine users with sysadmin rights

Command: select loginname from syslogins where sysadmin = 1;
go



The users dbadmin, admin, and Mssql have sysadmin privileges.

Step 9: Discover all the present databases.

Command: select name from sys.databases;
go

```
1> select name from sys.databases;
2> go
name

-----------------------------------
--------
master

tempdb

model

msdb


(4 rows affected)
1>
```

There are a total of four databases i.e master, tempdb, model, msdb.

Step 10: Discover all the users hashes

Command: select name, password_hash FROM master.sys.sql_logins;
go

```
1> select name, password_hash FROM master.sys.sql_logins;
2> go
name
        password_hash
------------------------------------------------------------------------------------------------------------------------
-------- ---------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------------------
----------------------------------------
sa
        0x020011DBFAF35BA0D5E61A769E3604230FDE23E5D3E01E7FF0BA3875CF75554803E2F1E1977B78DE8F4489C95DF9BE979C02F1DEC5513
00C109C408C427934815755B600C7E0



##MS_PolicyEventProcessingLogin##
        0x0200191CF079F310FB475527AC320ABA7A4E8D5C3567BEF2462B96CE8A8629B7F986ED344AA0963AC3A096DA77056DAD77A4576444312
82E2AA2C2243BC635ABC6BB5F52552C



##MS_PolicyTsqlExecutionLogin##
        0x0200677385ACFE08BB1119246CF20F9D17C3A0D86BBB1D48874725F2C2E0E021260B885D0BA067427E09AFAD9079E6759AD6497EE7F1E
F3CD497D500585D7727EEBA64426083
```

```
admin
        0x02003814EDD67DCAB815B733D877A0FE7EC3470185864BD673C7273BA76C31E000C15E9FAE25A826F6BA03892E37D6A1ACAE17F171D21
DAD7B20D874CCC259BBF9FA2230B9C0



Mssql
        0x020001786154BB350AC708B5A4C3FC6B90DC68418A13BA5FCB76B155F8EEE14D72988EDB559D9A2D0D6FD5DD25B1FAB8431C0CA424D747
A5743624C30AA772B40C8F23C66E6A4



Mssqla
        0x0200987F06858112A7FA0C70FE3F53C64061B35AE864782FC9CFCDA3954ED60CA7E47E8497A571D177EDB596F125CB529D7B2753E4D8E
913C2B127A12207E3BCB75F70E29CB5



auditor
        0x020061CBE8509DFEA47FBC20BE854C4AC517BF6AA67F9F7C12D7D1EFB1F500BE279643C6CD19D370F9EFF4F2D9B981A16F6916BC4534E
8BA42D718F8B908FBFFFB40D5CC1A5E



dbadmin
        0x02000D6C6A0D55F536F9DBFF2D8CC1E0965C550E1A1A1E7C6DF8B7E6534AB817408F86DD9592B206862C4B7A3D1F6CA85F439360171D7
C5143D6FBA8606675DBAF5BEA40D15B



(8 rows affected)
1>
```

We can crack these hashes using the john the ripper tool and get plain-text credentials.

Step 11: Identify that the xp_cmdshell is enabled or not.

Command: SELECT name, CONVERT(INT, ISNULL(value, value_in_use)) AS IsConfigured
FROM sys.configurations WHERE name = 'xp_cmdshell';
go

```
1> SELECT name, CONVERT(INT, ISNULL(value, value_in_use)) AS IsConfigured FROM sys.configurations WHERE name = 'xp_cmdsh
ell';
2> go
name                                IsConfigured
----------------------------------- ------------
xp_cmdshell                                    1

(1 rows affected)
1>
```

The xp_cmdshell is enabled on the target machine.

Flag: enabled

References:

1. MSSQL (https://www.microsoft.com/en-in/sql-server/sql-server-2019)
2. SQLCMD (https://docs.microsoft.com/en-us/sql/tools/sqlcmd-utility?view=sql-server-ver15)