Writer's Notebook: SQL Injection 24 March 2013

SQL injection is an interesting sort of software vulnerability. It is relatively common in websites, and I've actually found a few vulnerabilities myself that were based around SQL injection.

Let me start by describing SQL. SQL, or Structured Query Language, is a language used to store and retrieve data in databases. Each database can have multiple named tables. Tables have multiple named columns. Tables can them have entries (rows) inserted that use the columns as their definition.

For a website, I might have a table called "users" with columns "username", "email", and "password". "password" would likely not be stored in plain text, but rather in a hash. for more information about hashes, refer to MP2 page 13 [in this portfolio, that's page 1B].

To insert a new user into the "users" table, I'll write:

```
1     INSERT INTO 'users' VALUES('george', 'george@gmail.com', 'a5de69f4');
```

When a website gets a hit for the "create user" page, it might run some code like this:

```
1     userName = userInput("username")
2     passWord = userInput("password").hash()
3     email = userInput("email")
4
5     query = "INSERT INTO 'users' VALUES('" + userName + "', '" + email + "',
'" + password + "');"
6     mysql_execute(query)
```

This would take user input and shove it into the database. Remember this code example, because it is important.

To obtain data from a database, I write:

```
1     SELECT * FROM 'users' WHERE username='george';
```

Which will return with a row that contains George's username, email, and hashed password. I can use similar code as before to dynamically form any SQL query.

Another thing I should mention now is the concept of SQL 'comments'. When the combination of characters "--" is located somewhere in an SQL statement, everything after the "--" is not executed. This is useful for writing comments in SQL without throwing an error. This:

```
1     SELECT * FROM 'users'; hi I like tacos
```

would give an error, because "hi I like tacos" is not valid SQL. This:

```
1     SELECT * FROM 'users'; -- hi I like tacos
```

would not, because everything after -- is ignored.

An attacker can use these concepts to his advantage.

Imagine a "log-in" procedure in your code that goes like this: