Sn0wF0x Doks

SQL [Structured Query Language] Injection

Websites Nowadays often Uses Database Connections. The Most common Way to Do this is to use SQL.

SQL is one kind of Language to use whenever you work with Databases. This Fact makes it as Usable, as well as Problematic. What I refer to, is useing it for bad Purposes. Thats where **SQL Injection** comes in.

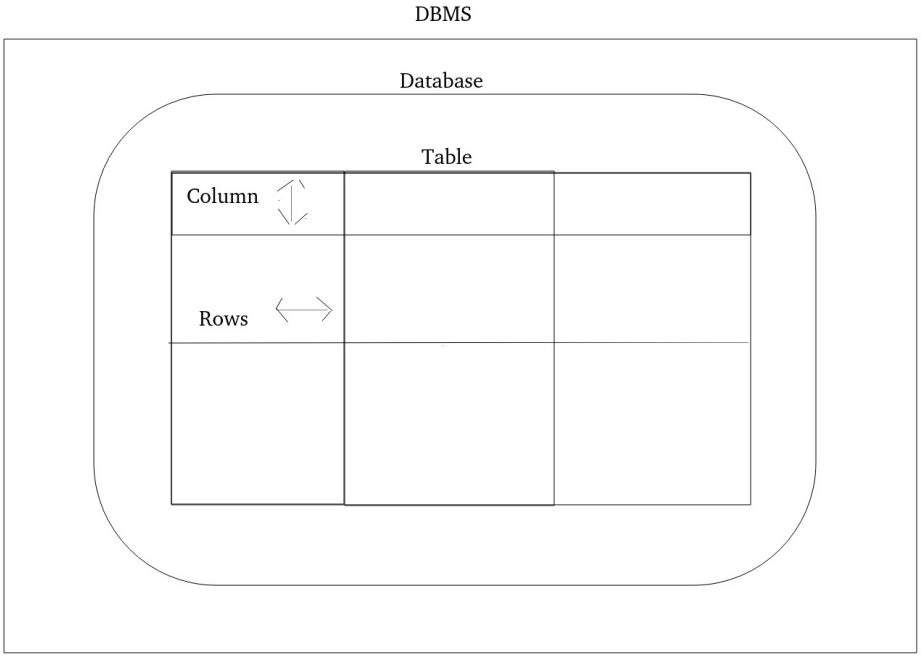
To put it in Short: **SQL Injection is the Abuse of Using Weak-Secured SQL Querys to inject malcioius Data or Make the SQL Server do Stuff, that he is NOT Intendet to do**.

So, what exactly is a Database that we will try to Hack via SQL Injection?

A Database is A Collection of Data with an Easy to read Layout.

Instead of Reading Plain Cryptic Data, A Database has Tables with columns and Rows to be Adressed.

Fairly Easy using SQL Language. This „Easy to Read Layout“ is what called an **DBMS** (or Database Management System) like **MSQL**.



You can have tons of Databases in on DBMS and tons of Tables in each Database.

So, now that we roughly know what we are Dealing with, Lets Talk about SQL.

SQL [Structured Query Language] is as I just said the „Language to Go“ to Communicate with a Databse.

To understand SQL Injection, we first need to understand the SQL Syntax.

Theory Part - BASIC SQL-Course

Lets say you want to get all data from a table called „users“. Then you would type:

```
SELECT * FROM users;
```

In short, you are telling SQL What to do, then what exactly is included in the order, followed by an instruction to beginn to declare a Destination of your command, followed by the table you desire to work with . Pretty Easy huh? This little Command uses a „*“ to tell mysql that you want to select ALL columns of the table users. Lets say you want only some very specific columns. What do we do?

```
SELECT column1,column2 FROM users;
```

Sounds like the same but with a bit more of Detail? Righto.
But can we only Define the Specific Columns? Not Really, we can also define which and how many results will get back to you. Lets say we return to this little Command here:

```
SELECT * FROM users;
```

And now we only want the very first enty of this.. how do we do this?

```
SELECT * FROM users LIMIT 1;
```

Yep its as easy as it looks like. This will Limit your Results to the very first one and nothing else. Now you only want the second one huh?

```
SELECT * FROM users LIMIT 1,1;
```

ez-pz

When you want to do SQL Injection, you want some juicy Admin Crediantls Right? Well Lets see how we would adress this issue..

```
SELECT * FROM users where username=‘admin’;
```

I guess you can see a Pattern in here.

Of Course you can also check if 2 Parameters are Positive before you want to get an result

```
SELECT * FROM users where username=‘admin’ and password=‘password’;
```

One Thing you can (and will learn Later on) is „Blind-SQL Injection“ where you basically see no result at all while trying to find the Database/Table Detals. For this Kind of Injection you will need to learn to guess some Database/Table details like the name… So Lets say you want a specific Name out of an Table usernames that starts with an „a“ (like admin for Example)

```
SELECT * FROM users where username like ‘a%’;
```

You see what changed in here? The = turned into an „like“ and the search Term turned to „a%“ which means, started with an „a“. There are some variants of this. Let’s say you want an entry where the opposite is the Case (So where you would like to see only Entries which ENDS with an „a“

```
SELECT * FROM users where username like ‘%a’;
```

Or prefer to find out which has an „a“ in them?

```
SELECT * FROM users where username like ‘%a%’;
```

So, this is the SELECT function for ONE Table (dont worry it wont get much harder, just a liiiiiitlle bit. Lets say you have 2 Tables and want to select from both them Them at the Same time:

```
SELECT column1,column2 FROM table1 UNION SELECT column1, column2
FROM table2;
```

(This is the Hardest Part from This, dont be worry haha)

So, up to now we can Read a Table(SELECT) and combine(UNION) huh? What comes next? Well, we are hackers arent we? We sometimes manipulate Data in Databases to give us a chance for a Good Privilege Escalation. So, how do we … Update the Entries?

```
update users SET username='root',password='pass123' where
username='admin';
```

So, what do we have here? I guess its Pretty Obvious but I want to show it to you :) We can relatively Precise tell the Query What, where and to what to change stuff.

Do we have some Black Hats here? (I wont hope so lol) Then the next Thing is for you

```
delete from users where username='martin';
```

The Most Important Thing about every of the Examples above is that the „;“ needs to be put to the end of the Query.

Thats it! Basic SQL For Basic SQL Injections!

Pratice Part – SQL Injections in a Nutshell

But How do we use THIS for Hacking when most of the websites URL Looks like this?

https://testwebsite.com/cataloge/article?id=1

Well, that is how YOU see it. The Webserver sees it a BIT different

```
SELECT * FROM articles where id=1 and Private =0 LIMIT 1;
```

Here we have 2 interesting Parameters:

=> id=1
=> Pivate = 0

The First one Obviously represents the ID of the Article in the Catalogue.
The Second one indicates that the Article is made for PUBLIC but not for private.. So in case that the Second one also is 1 which means Private…. How do we mmake it „public“?

https://website.thm/blog?id=2;--

What happend here?

Well, we have 2 new elements:

=> ;
=> –

The First one Says „End the String right now“
The Second one Says „Ignore everything that comes after me

Can you imageine what that means?

Just for you to remember, this is the original query

```
SELECT * FROM articles where id=1 and Private =0 LIMIT 1;
```

now with the new elemts it looks like this

```
SELECT * FROM articles where id=2;-- and Private =0 LIMIT 1;
```

or

```
SELECT * FROM articles where id=2;--;
```

The Private Part gets completley ignored due to the 2 Elements!

This technique is one of three Different Versions.

> This one is called „In-Band“ SQL Injection
> The Second one wwould be „Blind“ SQL Injection
> And the Thrird One „Out of Band“ SQL Injection.

So, while we learned a bit about In-Band SQLinjection, lets cover it a bit more… We have Some Things to discuss:

„In-Band SQL Injection“

Lets Talk About Feedback. We all need Feedback to know if something we did was right or wrong right? SQL Injection needs also some kind of feedback to know how we did our attack. This Leans us to the Following Kinds of Injection:

„Error-Based SQL Injection“

By Far one of the most usefull. The Information will directly be visible in your Webbrowser and can be used to enumerate the DB/Table even more

„Union-based SQL Injection“

If you know your target has tons of data this one here comes handy. You already leart why (If not, go back to the UNION Part of this Doc)

So enough for In-band, Lets Move on to the Next one:

„Blind-SQL-Injection“

You know what I said about Optical Feedback being Handy? Well Blind-SQL Says „We dont talk about error no-no-no-no“ and I mean this in the way I say it.

This is Basically Try and Error and hoping to guess right, because you only get some juicy text if you actually do the Injection Right.

One Little Help here might be „' OR 1=1;--“

This Basically means:

=> Force a True to the Webserver because 1=1 is ALWAYS true in SQL

This might or might now help us to determine how the Database will react to our Attempts.

In Normally Case The Way to go is the Following:

=> Find Database
=> Find Tables
=> Find Columns
=> Generate Final Query

This can take LOTS of Time and I cant Describe all of that here because I dont know all of this Method.

But one last „Little help“ I wanna give you.

There is something called „Time-based Blnd SQL Injection“.

What That Means? Well you wont get any help from The Server while Doing Blind-SQL Injection BUT you can add the SLEEP(5);-- Parameter to your Query so that the Query takes 5 Seconds to Execute in Case all works as expected.

This is just a Small Help but its a help.

I havent show you much here from this method because it can and WILL be confusing, so please if you wanna know more, read fruther much more Detailled Material!

```
http://vulnsite.com/referrer=admin123' UNION SELECT SLEEP(5),2
where database() like 'u%';--
```

„Out of Band SQLi“

This is Something very Uncommon. It Needs 2 differenct communication Channels, one for the launch of the Attack and one for the Results. We dont discuss this here because its quite rare

Well Thats it! We are done!

The Absolute basics about SQL-Injection!

Just In case you want something more … programmish’, I can give you 2 Good Ideas

==> SQLMap
==>DSSS [Damn small SQLi Scanner]

Credits:

I Hope this might Help Someone on their Journey through Infosec :)

If so, feel free to contact me on Twitter : https://twitter.com/MarcusChachuls1

Created by Sn0wF0x