

## Points In PHP And SQL Security Topics Explained

The number one security topic present in applications that use PHP is the SQL injection. This is because PHP allows for web developers to make unfortunate mistakes when it comes to creating their SQL queries. But thankfully, fixing the problem is easy: all that is necessary is a few tips in security.

SQL injections are defined by the vulnerability in the SQL query that PHP developers make use of. When the developer in question puts forth an SQL query, he or she needs to make an effort to validate any input that could come from any web form or entry field. A simple input statement such as "a' OR 'a'='a'" could compromise the security of one's database with ease.

Magic quotes have long helped web developers secure their SQL query statements. But as it stands today, this function is deprecated and no longer in use. Magic quotes have received a bad reputation since they do escape quotes- but they do so on the entire input, and not necessarily just a certain field we need to escape. Magic quotes are a hassle, and can even lead to performance issues. Thus, developers tend to ignore them.

There is but one simple solution when it comes to getting rid of the threat of an SQL injection. This simple solution comes via the function `mysql_real_escape_string()`. This function was created specifically for safeguarding against SQL injections, so it's well worth the time to use. Just pass any values being inserted through this function, and the result is a perfectly escaped string.

Another good way to prevent SQL injections is to simply restrict authority in SQL users where possible. For instance: it would be a good idea to create individual users that do specific things: such as create a table or update rows in the said table. This can help make the task of ruining one's hard work much harder for malicious web users, although it's a lot more work for webmasters (Although well worth it).

It should be noted that programs and web applications that stop SQL injections should not be obtained- since they commonly cost quite a bit of money. As long as webmasters take precautions with what they create, there should be no reason to spend hundreds of dollars on software that only makes use of escape characters and formatting data correctly. This type of application is created to con webmasters into buying something they don't need- so don't fall victim to them!

### In Conclusion

There isn't much effort that needs to be exerted in order to declare a database safe from harm. All that is needed is a little prevention- which comes from avid usage of the function and design principles previously stated. It may also be a good idea to use SQL injection scanners on large web applications to cover holes that might not have been covered over the course of the development period.

### About the Author

Learn more on [SQL Injection Explanation](#) and [SQL Injection Explain](#).

Source: <http://www.zogol.com>