# Internet Security Holes and How to Avoid Them

Open Web Application Security Project (OWASP): <https://www.owasp.org/index.php/Main_Page>

## Browser Attacks:

The attacker will insert HTML script into a text boxl

### Script Injection Attack:

The attacker use text input box to send a script that will be executed by the browser:

**How to prevent this attack:**

Use HTML sanitization library to verify all user inputs.

For example: **Bleach** is such a library in python.

**How to test for this attack:**

Instead of regular text, insert some html code:  
<script>

setTimeout(function() {

var tt = document.getElementById('content');

tt.value = "<h2 style='color: #FF6699; font-family: Comic Sans MS'>Spam, spam, spam, spam,<br>Wonderful spam, glorious spam!</h2>";

tt.form.submit();

}, 2500);

</script>

## SQL Attacks:

A guide to preventing SQL injection: <http://bobby-tables.com/>

### SQL Injection Attack

The attacker will insert SQL actions instead of text to gain access and/or manipulate the server-side database

**How to prevent this attack:**

Never, never, NEVER use Python string concatenation (+) or string parameters interpolation (% or {0}) to pass variables to a SQL query string. Not even at gunpoint.

Instead, use the cursor.execute command parameters to pass the argument.

For example:

Instead of:

cursor.execute("INSERT INTO test VALUES ({0})".format(value)

Use the following format:

cursor.execute("INSERT INTO test VALUES (%s)"**,**(value,))

**How to test for this attack:**

Send the following text in the text box:

* '); delete from posts; --
* WARNING: this will erase all the records in the database if it’s not protected.