**Example:**

**uname = request.POST['username']** ;

**passwd = request.POST['password']** ;

# SQL query vulnerable to SQLi

sql = “**SELECT** id **FROM** users **WHERE** username=’” + **uname** + “’ **AND** **password**=’” + **passwd** + “’”

# **Execute** the **SQL** statement

**database**.**execute**(**sql**)

These input fields are vulnerable to SQL Injection.

For example, attacker could use a trick involving a single quote and set the passwd field to:

password' OR 1=1

As a result, the database server runs the following SQL query:

**SELECT** id **FROM** users **WHERE** username='username' **AND** **password**=**'password' OR 1=1**'

Because of the OR 1=1 statement, the WHERE clause returns the first id from the users table no matter what the username and password are. The first user id in a database is very often the administrator. In this way, the attacker not only bypasses authentication but also gains administrator privileges.