

# TP INJECTION SQL SECURITY : LOW



The screenshot shows the DVWA (Damn Vulnerable Web Application) interface. At the top, there is a navigation menu with options: Home, Instructions, Setup / Reset DB, Brute Force, Command Injection, CSRF, File Inclusion, File Upload, Insecure CAPTCHA, SQL Injection (highlighted), and SQL Injection (Blind). The main content area is titled "Vulnerability: SQL Injection" and contains a form with a text input field containing the text "' or 1=1 --" and a "Submit" button. Below the form, there is a section titled "More Information" with a list of links to external resources:

- <http://www.securiteam.com/securityreviews/5DP0N1P76E.html>
- [https://en.wikipedia.org/wiki/SQL\\_injection](https://en.wikipedia.org/wiki/SQL_injection)
- <http://ferruh.mavituna.com/sql-injection-cheatsheet-oku/>
- <http://pentestmonkey.net/cheat-sheet/sql-injection/mysql-sql-injection-cheat-sheet>
- [https://www.owasp.org/index.php/SQL\\_injection](https://www.owasp.org/index.php/SQL_injection)
- <http://bobby-tables.com/>

**Pour le 4 premières questions, il est demandé d'utiliser des injections SQL manuelles :**

1. Déterminer le nombre et le nom des bases de données:
2. Déterminer le nombre et le nom des tables de la base de donnée utilisée.
3. Déterminer le nombre et le nom des colonnes de la table 'users'.
4. Déterminer le contenu des tables user et password.
5. Retrouver les résultats précédents en utilisant sqlmap