Reference

Definition **web app(**[**https://www.britannica.com/topic/Web-application**](https://www.britannica.com/topic/Web-application)**)**

**Ou (**[**https://www.sciencedirect.com/topics/computer-science/web-application**](https://www.sciencedirect.com/topics/computer-science/web-application)**)**

**1.4.1 Tautology-Based SQL Injection**

**This attacks depend upon taking advantage of the employment of conditional SQL always True, therefore circumventing authentication and access control.**

**Tautology refers to a statement, no matter parts, which is always true.**

**Attackers sow such statements within SQL queries for injecting into databases, therefore, the database continues to process unauthorized requests.**

**•\tOR '1'='1' is employed here such that the condition will always be true, finding entry.**

**•\tReal-World Scenario: A hacker compromises a login form of a website and bypasses user authentication to utilize an administrator's account.**

**•\tConsequence:**

**o\textbf{Unintended access of accounts.}**

**o\textbf{Possible breaches and tampering of data.}**

**o\textbf{Damage to user confidence and legal recourse.}**

**•\tPreventive measures:**

**o\textbf{Use parameterized queries.}**

**o\textbf{Sanitize the user input string thoroughly.}**

**1.4.2 Error-Based SQL Injection**

**This technique utilizes provoking and retrieving database error messages to learn information about the database schema.**

**By deliberately generating SQL errors, attackers can extract information such as table structure, column types, and even data.**

**•\tReal-World Scenario: An ill-configured e-commerce website is hacked by a hacker, where table names containing customer payment details are revealed.**

**•\tImpact:**

**o\tdisclosure of database schema.**

**o\tenabling more precise attacks.**

**o\tdisclosure of sensitive data.**

**•\tCountermeasures:**

**o\tdisable detailed error reporting in production.**

**o\tenforce strict error handling.**

**1.4.3 Blind SQL Injection**

**This technique is used when the application does not return database error messages and attackers have to make an indirect inference about database behavior.**

**1.4.3.1 Content-Based Blind SQL Injection**

**•\\tDetailed Description: Attackers make use of conditional statements to find minute differences in the application content or behavior.**

**•\\tImpact:**

**o\\tStealing data bit by bit through logical inference.**

**o\\tExtremely hard to find.**

**•\\tCountermeasures:**

**o\\tUse of web application firewalls (WAFs).**

**o Regular code reviews.**

**-**

**-**

**--**

**How sql works (**[**https://brightsec.com/blog/sql-injection-attack/**](https://brightsec.com/blog/sql-injection-attack/)**)**

**Les impacts (**[**https://brightsec.com/blog/sql-injection-attack/**](https://brightsec.com/blog/sql-injection-attack/)**)]**