Abdullrahman Alghanim

Lab 8 Database Hacking

CIS 4204

03/15/2025

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a web application

AI-generated content may be incorrect.

A screenshot of a computer security level

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

The ports that are configured are a great amount and you can see the difference between the two photos and how much of a difference in the ports there is

A screenshot of a computer

AI-generated content may be incorrect.

The reason 'OR '1'='1 is effective in an SQL injection attack lies in how it manipulates the logic of a database query. When user input is not properly sanitized, attackers can insert specially crafted strings that alter the intended behavior of a SQL statement. The phrase 'OR '1'='1 is a classic example that forces a conditional statement to always evaluate as true. By doing this, an attacker can potentially bypass login credentials or access data they are not authorized to see. The expression '1'='1 is always true, so when it is injected into a query, it tricks the database into thinking the condition has been met, even if the user provides incorrect information.

This form of attack is known as SQL injection and is considered one of the most dangerous vulnerabilities in web applications. According to the Open Web Application Security Project (OWASP), SQL injection is consistently ranked among the top security threats due to its simplicity and severe impact. The MITRE Corporation also classifies this issue under CWE-89, which refers to improper neutralization of special elements used in SQL commands. The attack works because the application fails to distinguish between user data and command syntax, allowing input to interfere with how the query operates. To prevent this, developers are encouraged to use secure coding practices such as parameterized queries and input validation.

**Sources:**

* OWASP. (2023). *SQL Injection*. Retrieved from https://owasp.org/Top10/A01\_2021-Broken\_Access\_Control/
* MITRE. (2023). *CWE-89: Improper Neutralization of Special Elements used in an SQL Command*. Retrieved from https://cwe.mitre.org/data/definitions/89.html