SCENARIO

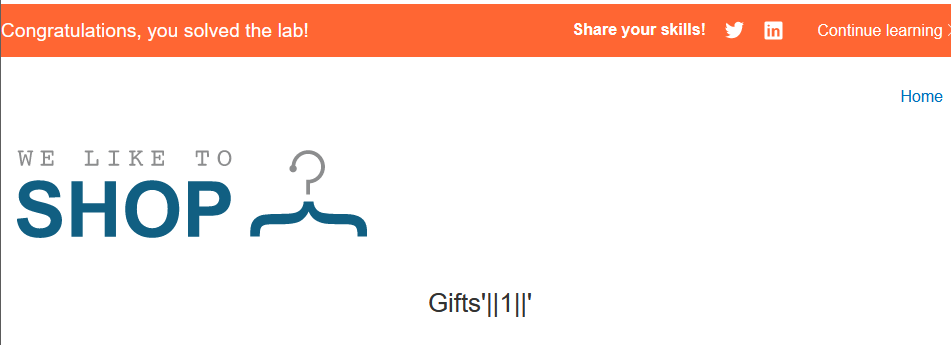
The application possesses a NoSQL Injection Vulnerability in its product category filter functionality as it is powered by a MongoDB NoSQL Database. We will try to exploit the vulnerability in order to gain administrator privileges using MongoDB operators.

**PROCEDURE**

1. Open the web application and select any category filter by clicking it.
2. Then inject the payload give below to make the condition true in order to display all the products.

**PAYLOAD**

# Gifts'||'1'=='1

**PROOF OF CONCEPT**

**REMEDIATION**

1. **Avoid Constructing Dynamic Queries:** If possible, use static, predefined queries or database methods. For MongoDB, this might mean using built-in functions to retrieve data rather than constructing dynamic JSON queries.
2. **Use Parameterized Queries:** Parameterized queries or prepared statements ensure that user input is always treated as data and not executable code. This is key in preventing injection attacks.
3. **Data Validation:** Always validate and sanitize user inputs. Use a known library or framework that has been specifically designed for this.
4. **Use Database User Roles:** Limit the rights of the application's database user. For instance, if an application only requires read access to the database, then it shouldn't have write access. By practicing the principle of least privilege, you can reduce the potential impact of an attack.
5. **Escape User Input:** Always escape user input, even if you're using prepared statements. This provides an extra layer of security against injection attacks.