Apply filters to SQL queries

Project description

As a security professional in a large organization, I investigated possible security threats by analyzing login attempt data and employee records. The objective was to identify suspicious behavior patterns, such as failed login attempts outside regular hours, logins from unexpected locations, or anomalies related to employee departments. To conduct this investigation, I queried the `log_in_attempts` and `employees` tables using SQL filters.

Retrieve after hours failed login attempts

SELECT * FROM log_in_attempts WHERE login_time > '18:00:00' AND success = 0

Retrieve login attempts on specific dates

SELECT * FROM log_in_attempts WHERE login_date BETWEEN '2022-05-08' AND '2022-05-09'

Retrieve login attempts outside of Mexico

SELECT * FROM log_in_attempts WHERE NOT country LIKE 'MEX%'

Retrieve employees in Marketing

SELECT device_id, department, office FROM employees WHERE department = 'Marketing' AND office LIKE 'East%'

Retrieve employees in Finance or Sales

SELECT employee_id, username, department FROM employees WHERE department = 'Sales' OR department = 'Finances'

Retrieve all employees not in IT

SELECT * FROM employees WHERE NOT department = '**Information Technology'**

Summary

This activity demonstrated the use of SQL filtering techniques to isolate relevant security-related data across multiple datasets. I used `WHERE` clauses with logical operators such as `AND`, `OR`, and `NOT` to create precise conditions that helped narrow down the results. For instance, I retrieved failed login attempts after working hours using:

```
SELECT * FROM log_in_attempts WHERE login_time > '18:00:00' AND success = 0;
```

To identify logins within a specific date range, I applied the `BETWEEN` operator on date values. Additionally, the `LIKE` operator with wildcards was used to match pattern-based conditions, such as offices beginning with "East" or countries starting with "MEX".

The queries also filtered employees by department, combining `OR` and `AND` logic to extract precise sets of records (e.g., employees in Finance or Sales, or those not in IT). This structured approach using SQL filters is critical in security investigations, where isolating the right data quickly enables timely threat assessment and mitigation.