Apply filters to SQL queries

Project description

Used SQL to investigate potential security issues with login attempts and employee machines. Focused on filtering data from 'employees' and 'log_in_attempts' tables.

Retrieve after hours failed login attempts

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

Use code with caution.

- Find failed logins (success = 0) after 6 PM.

Retrieve login attempts on specific dates

SQL
SELECT *
FROM log_in_attempts
WHERE login_date = '2022-05-08' OR login_date = '2022-05-09'
Get all logins on May 8th and 9th, 2022 (suspicious event timeframe).

Retrieve login attempts outside of Mexico

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

Use code with caution.

- Filter out any logins from Mexico (both 'MEX' and 'MEXICO').

Retrieve employees in Marketing

SQL
SELECT *
FROM employees
WHERE department = 'Marketing' AND office LIKE 'East-%';

Use code with caution.

- Find Marketing employees in offices located in the East building.

Retrieve employees in Finance or Sales

SQL
SELECT *
FROM employees
WHERE department = 'Finance' OR department = 'Sales';

Use code with caution.

- Get all employees from either Finance or Sales departments

Retrieve all employees not in IT

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

Use code with caution.

- Exclude employees from the IT department.

Summary

Used SQL filters to narrow down data for security investigation.
Applied `AND`, `OR`, `NOT` to combine conditions.
Used `LIKE` with `%` for pattern matching (e.g., 'East-%').
Filtered dates, times, and other data types.
These queries helped identify potential security risks and gather info for further analysis.