Apply filters to SQL queries

Project description

I'm responsible for enhancing system security within my organization. This involves investigating and resolving potential security issues, as well as updating employee computers as required. The steps outlined below illustrate how I utilize SQL with filters for security tasks.

Retrieve after hours failed login attempts

After a potential security incident post-business hours (after 18:00), I initiated an investigation into failed login attempts. Utilizing SQL, I used a query to filter for unsuccessful login attempts occurring after 18:00. The query selects data from the log_in_attempts table, employing a WHERE clause with an AND operator to narrow down results to failed attempts post-18:00. The conditions "login_time > '18:00'" and "success = FALSE" were used to isolate relevant data points.

Retrieve login attempts on specific dates

To investigate a suspicious event on 2022-05-09, I developed a SQL query to filter login attempts on that specific date and the preceding day, 2022-05-08. The query selects all data from the log_in_attempts table and utilizes a WHERE clause with an OR operator to filter for login attempts on either date. Conditions "login_date = '2022-05-09'" and "login_date = '2022-05-08'" were applied to isolate relevant login activities.

Retrieve login attempts outside of Mexico

Upon reviewing the organization's login attempt data, I identified potential issues with attempts originating from outside Mexico. To investigate further, I constructed a SQL query to filter for login attempts not originating from Mexico.
The query selects all data from the log_in_attempts table and employs a WHERE clause with NOT to exclude Mexico. Using LIKE with the pattern MEX%, I matched entries representing Mexico as MEX or MEXICO. The percentage sign (%) in LIKE represents any unspecified characters, allowing for flexibility in matching.

Retrieve employees in Marketing

To streamline computer updates for specific employees in the Marketing department, particularly those situated in the East building, I devised a SQL query to filter for their machines. The query starts by selecting all data from the employees table and employs a WHERE clause with AND to pinpoint employees in the Marketing department within the East building. I utilized LIKE with the pattern East% to match office locations in the East building, as indicated in the office column. The first condition, department = 'Marketing', targets Marketing department employees, while the second condition, office LIKE 'East%', narrows down employees in the East building.

Retrieve employees in Finance or Sales

To facilitate the update of machines for employees in the Finance and Sales departments, I crafted a SQL query to filter for their respective machines.
Beginning with the selection of all data from the employees table, I utilized a WHERE clause with OR to target employees in either the Finance or Sales departments. I opted for the OR operator instead of AND to capture all employees belonging to either department. The first condition, department = 'Finance', filters for employees from the Finance department, while the second condition, department = 'Sales', narrows down employees from the Sales department.

Retrieve all employees not in IT

To proceed with the final security update for employees outside the Information Technology department, I devised a SQL query to filter for their machines. Commencing with the selection of all data from the employees table, I employed a WHERE clause with NOT to identify employees not belonging to the Information Technology department. This approach ensures that only employees outside this department are included in the query output.

Summary

I utilized SQL queries to extract targeted information from two distinct tables: log_in_attempts and employees. Employing the AND, OR, and NOT operators allowed for precise filtering tailored to each task's requirements. Additionally, I utilized the LIKE operator alongside the percentage sign (%) wildcard to filter for patterns within the data.