Apply filters to SQL queries

By Mudapaka Sailaxman
Project Type: Cybersecurity Data Filtering with SQL
Tools Used: SQL (Structured Query Language)

Project description

Through the use of SQL, I was able to extract, filter, and analyze data from `log_in_attempts` and `employees` tables to support cybersecurity investigations and system maintenance tasks. The queries helped identify suspicious login activities, filter login attempts by date, time, location, and determine specific employee groups for targeted system updates. This project demonstrates how SQL can be used effectively in real-world scenarios to support data-driven decision-making in cybersecurity.

Retrieve after hours failed login attempts

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

This query selects all columns (`*`) from the `log_in_attempts` table. It filters the results using the `WHERE` clause with two conditions:

- `login_time > '18:00:00'`: This returns only the login attempts that happened **after 6:00 PM**.
- `success = 0`: This returns only the **failed login attempts**.

Together, the query helps identify **unsuccessful login attempts that took place after business hours**, which may be useful for spotting **suspicious or unauthorized access attempts**.

Retrieve login attempts on specific dates

```
SELECT * FROM log_in_attempts
WHERE login_date = '2022-05-09' OR login_date = '2022-05-08';
```

This SQL query is used to retrieve **all login attempts** that occurred on **May 9, 2022**, and **May 8, 2022** from the `log_in_attempts` table. These dates are important for investigating a suspicious event that happened on May 9.

- `SELECT * FROM log_in_attempts`:
  This selects **all columns** for each login attempt (such as username, time, country, success status, etc.).

- `WHERE login_date = '2022-05-09' OR login_date = '2022-05-08'`:
  This filter ensures that only the login attempts from **the day of the event** and **the day before** are returned.

This query helps in examining the **login activity leading up to and during the suspicious event**, which can be useful for identifying unusual behavior, failed attempts, or unauthorized access

Retrieve login attempts outside of Mexico

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

This query retrieves all login attempt records from the `log_in_attempts` table where the `country` value **does not start with** `'MEX'`.

- `SELECT * FROM log_in_attempts`:
  Selects **all columns** of each login attempt, including event ID, username, login time, country, IP address, and success status.

- `WHERE country NOT LIKE 'MEX%'`:
  Filters out records where the `country` field **starts with** `'MEX'`, such as `'MEX'`, `'MEXICO'`, or `'MEXICO CITY'`.

This is useful when investigating **suspicious login activity that did not originate in Mexico**.

Retrieve employees in Marketing

```
SELECT * FROM employees
WHERE department LIKE '%Marketing%'
AND office LIKE 'East%';
```

This query retrieves all employees from the `employees` table who work in the **Marketing department** and are located in the **East building**, regardless of the specific room or floor.

- `SELECT * FROM employees`:
  This selects all available information about each employee, such as their ID, device ID, username, department, and office.

- `WHERE department LIKE '%Marketing%'`:
  Filters the results to include only those employees whose department contains the word `"Marketing"`. The `%` symbols allow for other text before or after, such as `"Digital Marketing"` or `"Marketing and Sales"`.

- `AND office LIKE 'East%'`:
  Further narrows the results to employees located in offices that begin with `"East"` (e.g., `"East-170"`, `"East-320"`). The `%` symbol matches any additional characters after `"East"`.

This query helps identify the **exact machines and users** in the Marketing department located in the **East building**, so the security team can perform updates on the correct systems.

Retrieve employees in Finance or Sales

```
SELECT * FROM employees
WHERE department LIKE '%Sales%'
```

```
  OR department LIKE '%Finance%';
```

- `SELECT * FROM employees`:
  Selects all columns for each employee, including their employee ID, device ID, username, department, and office location.

- `WHERE department LIKE '%Sales%' OR department LIKE '%Finance%'`:
  Filters the results to include only those employees whose department contains the word **"Sales"** or **"Finance"**.

    - The `%` wildcard before and after allows matching values like `"Inside Sales"`, `"Sales & Marketing"`, `"Corporate Finance"`, or `"Finance Dept"`

This query is used to retrieve information about all employees who work in the **Sales** or **Finance** departments from the `employees` table. This helps the security team identify machines that require updates for these specific groups.

Retrieve all employees not in IT

This SQL query is used to retrieve all employees **who are not part of the Information Technology (IT) department** from the `employees` table. These are the employees whose machines **still need a security update**.

- `SELECT * FROM employees`:
  This part of the query selects **all columns** from the `employees` table, such as employee ID, username, department, device ID, and office.

- `WHERE department NOT LIKE '%Information Technology%'`:
  This condition filters out employees whose department name **contains** `"Information Technology"`.

    - The `%` wildcard allows for flexibility in case the department name appears with other words, like `"Corporate Information Technology"`
    -

Summary

In this project, I used SQL to analyze login data and employee records to support security investigations and updates. I worked with two main tables: `log_in_attempts` and `employees`.

Key Queries and Their Purpose:

- **After-hours failed logins:** Identified failed login attempts after 6:00 PM to detect suspicious behavior.
- **Login attempts on specific dates:** Retrieved logins from 2022-05-08 and 2022-05-09 for incident review.
- **Logins outside Mexico:** Excluded attempts from “MEX” and “MEXICO” using `NOT LIKE`.
- **Marketing department (East building):** Found employees in Marketing with offices starting with “East”.
- **Sales or Finance employees:** Filtered employees from either department for updates.
- **Exclude IT department:** Listed all other employees needing system updates.

Skills Used:

- SQL filtering (`WHERE`, `AND`, `OR`, `LIKE`, `NOT LIKE`)
- Pattern matching
- Date/time conditions
- Querying based on business/security needs