Apply filters to SQL queries

Project description

It presented a scenario where I was working as a security professional at a large organization. Part of the job was to investigate security issues to help keep the system secure. Was discovered some potential security issues that involve login attempts and employee machines. The goal is to examine the organization's data in their `employees` and `log_in_attempts` tables. I used SQL filters to retrieve records from different datasets and investigate the potential security issues.

Retrieve after hours failed login attempts

The team found failed login attempts made after business hours. I used the following query to view the login attempts made after hours:

```
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 41
Server version: 10.3.39-MariaDB-0+deb10u2 Debian 10

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [organization]> clear
MariaDB [organization]> select * from log_in_attempts where login_time > '18:00' and success = 0;
+----------+----------+------------+------------+---------+-----------------+---------+
| event_id | username | login_date | login_time | country | ip_address      | success |
+----------+----------+------------+------------+---------+-----------------+---------+
|        2 | apatel   | 2022-05-10 | 20:27:27   | CAN     | 192.168.205.12  |       0 |
|       18 | pwashing | 2022-05-11 | 19:28:50   | US      | 192.168.66.142  |       0 |
|       20 | tshah    | 2022-05-12 | 18:56:36   | MEXICO  | 192.168.109.50  |       0 |
|       28 | aestrada | 2022-05-09 | 19:28:12   | MEXICO  | 192.168.27.57   |       0 |
|       34 | drosas   | 2022-05-11 | 21:02:04   | US      | 192.168.45.93   |       0 |
|       42 | cgriffin | 2022-05-09 | 23:04:05   | US      | 192.168.4.157   |       0 |
|       52 | cjackson | 2022-05-10 | 22:07:07   | CAN     | 192.168.58.57   |       0 |
|       69 | wjaffrey | 2022-05-11 | 19:55:15   | USA     | 192.168.100.17  |       0 |
|       82 | abernard | 2022-05-12 | 23:38:46   | MEX     | 192.168.234.49  |       0 |
|       87 | apatel   | 2022-05-08 | 22:38:31   | CANADA  | 192.168.132.153 |       0 |
|       96 | ivelasco | 2022-05-09 | 22:36:36   | CAN     | 192.168.84.194  |       0 |
|      104 | asundara | 2022-05-11 | 18:38:07   | US      | 192.168.96.200  |       0 |
|      107 | bisles   | 2022-05-12 | 20:25:57   | USA     | 192.168.116.187 |       0 |
|      111 | aestrada | 2022-05-10 | 22:00:26   | MEXICO  | 192.168.76.27   |       0 |
|      127 | abellmas | 2022-05-09 | 21:20:51   | CANADA  | 192.168.70.122  |       0 |
|      131 | bisles   | 2022-05-09 | 20:03:55   | US      | 192.168.113.171 |       0 |
|      155 | cgriffin | 2022-05-12 | 22:18:42   | USA     | 192.168.236.176 |       0 |
|      160 | jclark   | 2022-05-10 | 20:49:00   | CANADA  | 192.168.214.49  |       0 |
|      199 | yappiah  | 2022-05-11 | 19:34:48   | MEXICO  | 192.168.44.232  |       0 |
+----------+----------+------------+------------+---------+-----------------+---------+
19 rows in set (0.003 sec)
```

Retrieve login attempts on specific dates

The team investigated a suspicious event that occurred on 2022-05-08 and 2022-05-09. My task was to retrieve all login attempts made during these two dates. I used the following query:
`SELECT * FROM log_in_attempts WHERE login_date = '2022-05-08' OR login_date = '2022-05-09';`
It printed out a table with 75 rows which meant there were 75 login attempts during these dates.

Retrieve login attempts outside of Mexico

The goal was to see all logins that did not originate in Mexico. The country field includes entries with `'Mex'` or `'Mexico'`. I used the next query:
`SELECT * FROM log_in_attempts WHERE NOT country LIKE 'Mex%';`
It printed a table with 144 rows.

Retrieve employees in Marketing

My team was updating employee machines and it was necessary to obtain information about the employees in the marketing department who are located in offices in the East building.

```
MariaDB [organization]> select * from employees where department = 'Marketing' and office like 'East%';
+-------------+--------------+----------+------------+----------+
| employee_id | device_id    | username | department | office   |
+-------------+--------------+----------+------------+----------+
|        1000 | a320b137c219 | elarson  | Marketing  | East-170 |
|        1052 | a192b174c940 | jdarosa  | Marketing  | East-195 |
|        1075 | x573y883z772 | fbautist | Marketing  | East-267 |
|        1088 | k8651965m233 | rgosh    | Marketing  | East-157 |
|        1103 | NULL         | randerss | Marketing  | East-460 |
|        1156 | a184b775c707 | dellery  | Marketing  | East-417 |
|        1163 | h679i515j339 | cwilliam | Marketing  | East-216 |
+-------------+--------------+----------+------------+----------+
7 rows in set (0.001 sec)
```

Retrieve employees in Finance or Sales

The team needed to perform updates to the computers of the employees in the finance and sales departments. I provided the information on the employees with the query:
`SELECT * FROM employees WHERE department = 'Finance' OR department = 'Sales';`

Retrieve all employees not in IT

The team was required to make a final update to the employees computers. This update was already running in the employees in the IT department. To get the information on all the others employees I run the follow query:

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

Summary

I applied filters to SQL queries to get specific information on login attempts and employee machines. I used two different tables, `log_in_attempts` and `employees`. I used the `AND`, `OR`, and `NOT` operators to filter for the specific information needed for each task. I also used `LIKE` and the percentage sign (`%`) wildcard to filter for patterns.