Aplicación de filtros a consultas SQL

Descripción del proyecto

Mi organización está trabajando para hacer que su sistema sea más seguro. Es mi trabajo asegurarme de que el sistema sea seguro, investigar todos los posibles problemas de seguridad y actualizar las computadoras de los empleados según sea necesario. En los pasos siguientes se proporcionan ejemplos de cómo utilicé SQL con filtros para realizar tareas relacionadas con la seguridad.

Recuperar intentos de inicio de sesión fallidos después de horas

Hubo un posible incidente de seguridad que ocurrió después del horario comercial (después de las 18:00). Todos los intentos de inicio de sesión fuera del horario laboral que fallaron deben investigarse.

En el código siguiente se muestra cómo creé una consulta SQL para filtrar los intentos de inicio de sesión fallidos que se produjeron después del horario comercial.

```
MariaDB [organization]> SELECT *
    -> FROM log_in_attempts
    -> WHERE login_time > '18:00' AND success = FALSE;
+----------+----------+------------+------------+---------+-----------------+---------+
| 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 |
```

La primera parte de la captura de pantalla es mi consulta y la segunda parte es una parte de la salida. Esta consulta filtra los intentos de inicio de sesión fallidos que se produjeron después de las 18:00. Primero, comencé seleccionando todos los datos de la `tabla log_in_attempts`. A continuación, utilicé una `cláusula WHERE` con un operador `AND` para filtrar los resultados y generar solo los intentos de inicio de sesión que se produjeron después de las 18:00 y que no se realizaron correctamente. La primera condición es `login_time > '18:00'`, que filtra los intentos de inicio de sesión que se produjeron después de las 18:00. La segunda condición es `success = FALSE,` que filtra los intentos de inicio de sesión fallidos.

Recuperar intentos de inicio de sesión en fechas específicas

Un evento sospechoso ocurrió el 2022-05-09. Cualquier actividad de inicio de sesión que haya ocurrido el 09/05/2022 o el día anterior debe ser investigada.

En el código siguiente se muestra cómo creé una consulta SQL para filtrar los intentos de inicio de sesión que se produjeron en fechas específicas.

```
MariaDB [organization]> SELECT *
    -> FROM log_in_attempts
    -> WHERE login_date = '2022-05-09' OR login_date = '2022-05-08';
+----------+----------+------------+------------+---------+----------------+---------+
| event_id | username | login_date | login_time | country | ip_address     | success |
+----------+----------+------------+------------+---------+----------------+---------+
|        1 | jrafael  | 2022-05-09 | 04:56:27   | CAN     | 192.168.243.140 |       0 |
|        3 | dkot     | 2022-05-09 | 06:47:41   | USA     | 192.168.151.162 |       0 |
|        4 | dkot     | 2022-05-08 | 02:00:39   | USA     | 192.168.178.71  |       0 |
```

La primera parte de la captura de pantalla es mi consulta y la segunda parte es una parte de la salida. Esta consulta devuelve todos los intentos de inicio de sesión que se produjeron el 09/05/2022 o el 08/05/2022. Primero, comencé seleccionando todos los datos de la `tabla log_in_attempts`. A continuación, utilicé una `cláusula WHERE` con un operador `OR` para filtrar mis resultados y generar solo los intentos de inicio de sesión que se produjeron el 09/05/2022 o el 08/05/2022. La primera condición es `login_date = '2022-05-09'`, que filtra los inicios de sesión en 2022-05-09. La segunda condición es `login_date = '2022-05-08'`, que filtra los inicios de sesión en 2022-05-08.

Recuperar intentos de inicio de sesión fuera de México

Después de investigar los datos de la organización sobre los intentos de inicio de sesión, creo que hay un problema con los intentos de inicio de sesión que ocurrieron fuera de México. Estos intentos de inicio de sesión deben investigarse.

En el código siguiente se muestra cómo creé una consulta SQL para filtrar los intentos de inicio de sesión que se produjeron fuera de México.

```
MariaDB [organization]> SELECT *
    -> FROM log_in_attempts
    -> WHERE NOT country LIKE 'MEX%';
+----------+----------+------------+------------+---------+----------------+---------+
| event_id | username | login_date | login_time | country | ip_address     | success |
+----------+----------+------------+------------+---------+----------------+---------+
|        1 | jrafael  | 2022-05-09 | 04:56:27   | CAN     | 192.168.243.140 |       0 |
|        2 | apatel   | 2022-05-10 | 20:27:27   | CAN     | 192.168.205.12  |       0 |
|        3 | dkot     | 2022-05-09 | 06:47:41   | USA     | 192.168.151.162 |       0 |
```

La primera parte de la captura de pantalla es mi consulta y la segunda parte es una parte de la salida. Esta consulta devuelve todos los intentos de inicio de sesión que se produjeron en países distintos de México. Primero, comencé seleccionando todos los datos de la `tabla log_in_attempts`. Luego, utilicé una cláusula `WHERE` con `NOT` para filtrar por países que no sean México. Utilicé `LIKE` con `MEX%` como patrón para coincidir porque el conjunto de datos representa a México como `MEX` y `MEXICO`. El

signo de porcentaje (`%`) representa cualquier número de caracteres no especificados cuando se utiliza con `LIKE.`

Recuperar empleados en Marketing

Mi equipo quiere actualizar los ordenadores de determinados empleados del departamento de marketing. Para hacer esto, tengo que obtener información sobre qué máquinas de los empleados actualizar.

En el código siguiente se muestra cómo he creado una consulta SQL para filtrar los equipos de los empleados del departamento de marketing del edificio Este.

```
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 |
```

La primera parte de la captura de pantalla es mi consulta y la segunda parte es una parte de la salida. Esta consulta devuelve todos los empleados del departamento de marketing en el edificio Este. Primero, comencé seleccionando todos los datos de la tabla de `empleados`. A continuación, utilicé una `cláusula WHERE` con `AND` para filtrar por los empleados que trabajan en el departamento de marketing y en el edificio Este. Utilicé `LIKE` con `East%` como patrón para hacer coincidir porque los datos de la columna de oficina representan el edificio East con el número de oficina específico. La primera condición es la `parte department = 'Marketing'`, que filtra por empleados en el departamento de Marketing. La segunda condición es la parte `de oficina LIKE 'East%'`, que filtra para los empleados en el edificio East.

Recuperar empleados en Finanzas o Ventas

También es necesario actualizar las máquinas para los empleados de los departamentos de Finanzas y Ventas. Dado que se necesita una actualización de seguridad diferente, tengo que obtener información sobre los empleados solo de estos dos departamentos.

En el código siguiente se muestra cómo creé una consulta SQL para filtrar los equipos de los empleados de los departamentos de finanzas o ventas.

```
MariaDB [organization]> SELECT *
    -> FROM employees
    -> WHERE department = 'Finance' OR department = 'Sales';
+-------------+-------------+----------+------------+-------------+
| employee_id | device_id   | username | department | office      |
+-------------+-------------+----------+------------+-------------+
|        1003 | d394e816f943 | sgilmore | Finance    | South-153   |
|        1007 | h174i497j413 | wjaffrey | Finance    | North-406   |
|        1008 | i858j583k571 | abernard | Finance    | South-170   |
```

La primera parte de la captura de pantalla es mi consulta y la segunda parte es una parte de la salida. Esta consulta devuelve todos los empleados de los departamentos de Finanzas y Ventas. Primero, comencé seleccionando todos los datos de la tabla de `empleados`. A continuación, utilicé una `cláusula WHERE` con `OR` para filtrar por los empleados que están en los departamentos de Finanzas y Ventas. Usé el `operador OR` en lugar de `AND` porque quiero que todos los empleados que estén en cualquiera de los departamentos. La primera condición es `department = 'Finance'`, que filtra por empleados del departamento de Finanzas. La segunda condición es `department = 'Sales'`, que filtra por empleados del departamento de ventas.

Recuperar a todos los empleados que no están en TI

Mi equipo necesita realizar una actualización de seguridad más sobre los empleados que no están en el departamento de Tecnología de la Información. Para realizar la actualización, primero tengo que obtener información sobre estos empleados.

A continuación se muestra cómo creé una consulta SQL para filtrar los equipos de los empleados que no pertenecen al departamento de tecnología de la información.

```
MariaDB [organization]> SELECT *
    -> FROM employees
    -> WHERE NOT department = 'Information Technology';
+-------------+-------------+----------+-----------------+-------------+
| employee_id | device_id   | username | department      | office      |
+-------------+-------------+----------+-----------------+-------------+
|        1000 | a320b137c219 | elarson  | Marketing       | East-170    |
|        1001 | b239c825d303 | bmoreno  | Marketing       | Central-276 |
|        1002 | c116d593e558 | tshah    | Human Resources | North-434   |
```

La primera parte de la captura de pantalla es mi consulta y la segunda parte es una parte de la salida. La consulta devuelve todos los empleados que no están en el departamento de tecnología de la información. Primero, comencé seleccionando todos los datos de la tabla de `empleados`. A continuación, utilicé una `cláusula WHERE` con `NOT` para filtrar por empleados que no están en este departamento.

Resumen

Apliqué filtros a las consultas SQL para obtener información específica sobre los intentos de inicio de sesión y las máquinas de los empleados. Utilicé dos mesas diferentes, `log_in_attempts` y `empleados`. Utilicé los operadores `AND`, `OR` y `NOT` para filtrar la información específica necesaria para cada tarea. También utilicé `LIKE` y el comodín del signo de porcentaje (`%`) para filtrar los patrones.