Actividad de cartera: Aplica filtros a consultas SQL

Escenario

Analiza el siguiente caso. Luego, completa las instrucciones paso a paso.
Eres un/a profesional de la seguridad en una gran organización. Parte de tu trabajo consiste en investigar los problemas de seguridad para ayudar a mantener el sistema seguro.
Recientemente descubriste algunos potenciales problemas de seguridad relacionados con los intentos de inicio de sesión y las máquinas de los/las empleados/as.
Tu tarea consiste en examinar los datos de la organización en sus tablas **`employees`** y **`log_in_attempts`**. Tendrás que usar filtros SQL para recuperar registros de diferentes conjuntos de datos e investigar los posibles problemas de seguridad.
Nota: Este escenario involucra las mismas consultas que las del lab Filtra con AND, OR y NOT. Puedes repasar el lab para obtener capturas de pantalla e incluirlas en el documento de tu cartera. Si lo deseas, también puedes completar esta actividad sin volver a consultar el lab escribiendo tus consultas en la plantilla.

Descripción del proyecto

En este proyecto, se utilizarán consultas SQL para investigar posibles problemas de seguridad en una organización, analizando las tablas `log_in_attempts` y `employees`. El objetivo es aplicar filtros para identificar intentos de inicio de sesión sospechosos y examinar información relevante de los empleados. Se emplearán operadores SQL como `LIKE`, `AND`, `OR`, `NOT`, así como filtros basados en fechas y horas.

---

1. Recuperar intentos fallidos de inicio de sesión fuera del horario laboral

Se requiere identificar intentos de inicio de sesión fallidos que ocurrieron fuera del horario laboral (9:00 a.m. a 6:00 p.m.). La consulta SQL es la siguiente:

```
SELECT event_id, username, login_date, login_time, country, ip_address
FROM log_in_attempts
WHERE success = FALSE
AND (login_time < '09:00:00' OR login_time > '18:00:00');
```

---

2. Recuperar intentos de inicio de sesión en fechas específicas

Para obtener intentos de inicio de sesión en fechas específicas (por ejemplo, el 15 de octubre de 2023 y el 1 de noviembre de 2023), se utilizará la siguiente consulta SQL:

```
SELECT event_id, username, login_date, login_time, country,
ip_address, success
FROM log_in_attempts
WHERE login_date IN ('2023-10-15', '2023-11-01');
```

---

3. Recuperar intentos de inicio de sesión fuera de México

Para identificar intentos de inicio de sesión realizados fuera de México, se empleará la siguiente consulta:

```
SELECT event_id, username, login_date, login_time, country,
ip_address, success
FROM log_in_attempts
WHERE country != 'Mexico';
```

---

4. Recuperar empleados del departamento de Marketing

Para recuperar los empleados del departamento de Marketing, se puede utilizar la siguiente consulta SQL, asumiendo que la tabla `employees` incluye una columna `department`:

```
SELECT employee_id, name, department
FROM employees
WHERE department = 'Marketing';
```

---

5. Recuperar empleados de Finanzas o Ventas

Para obtener empleados que pertenezcan a los departamentos de Finanzas o Ventas, se puede usar la siguiente consulta:

```
SELECT employee_id, name, department
FROM employees
WHERE department IN ('Finanzas', 'Ventas');
```

---

6. Recuperar todos los empleados que no pertenecen al departamento de IT

Para identificar empleados que no forman parte del departamento de IT, se utilizará la siguiente consulta SQL:

```
SELECT employee_id, name, department
FROM employees
WHERE department != 'IT';
```

Resumen

Este proyecto consiste en el análisis de intentos de inicio de sesión y de la información de empleados mediante consultas SQL. Las consultas permiten identificar intentos de inicio de sesión fuera del horario laboral, accesos desde países distintos a México, y empleados que pertenecen o no a determinados departamentos. Se utilizan operadores SQL como `AND`, `OR`, `NOT`, y filtros basados en fechas y horas para lograr una clasificación efectiva de los datos. Estos filtros ayudan a detectar comportamientos inusuales y posibles vulnerabilidades de seguridad.

Detalles adicionales sobre las consultas SQL:

Uso de LIKE para buscar un patrón:
El operador `LIKE` se usa para buscar coincidencias parciales dentro de una columna. Por ejemplo, para buscar empleados cuyo nombre comienza con la letra "A", se puede emplear la siguiente consulta:

```
SELECT * FROM employees WHERE name LIKE 'A%';
```

Filtrar por fechas y horas:
Para filtrar resultados por rangos de fechas o por horas, se pueden utilizar operadores como `>=` y `<=`:

```
SELECT * FROM log_in_attempts WHERE login_date BETWEEN '2023-10-01'
AND '2023-10-31';
```

Uso de AND y OR para múltiples condiciones:
Para aplicar múltiples condiciones en una consulta SQL, se pueden combinar los operadores `AND` y `OR`:

```
SELECT * FROM employees WHERE department = 'Ventas' AND country =
'USA';
```

Uso de NOT para exclusiones:
El operador `NOT` permite excluir ciertos resultados en las consultas. Un ejemplo sería el siguiente:

```
SELECT * FROM log_in_attempts WHERE NOT success = TRUE;
```