SQL Query Analysis for Security Investigation

Project Overview

In my role as a security analyst, I conducted an investigation into potential security incidents related to login attempts and employee machine access. This report outlines the SQL queries I utilized to filter and analyze data from the employees and log_in_attempts tables within the organization's database.

Query Breakdown

Retrieve After-Hours Failed Login Attempts
SQL

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

1. Use code with caution.
   This query identifies all unsuccessful login attempts that occurred after 6:00 PM, potentially indicating unauthorized access attempts outside of normal business hours.

Retrieve Login Attempts on Specific Dates
SQL

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

2. Use code with caution.
   This query retrieves all login attempts on May 8th and 9th, 2022, which were flagged as a period of suspicious activity.

Retrieve Login Attempts Outside of Mexico
SQL

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

3. Use code with caution.
   This query filters out any login attempts originating from Mexico (both 'MEX' and 'MEXICO'), aiding in the identification of potentially unauthorized access from other locations.

Retrieve Employees in Marketing

SQL

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

4. Use code with caution.
   This query identifies employees in the Marketing department who are located in offices within the East building. This information is crucial for targeting specific machines for security updates.

Retrieve Employees in Finance or Sales

SQL

```
SELECT *
FROM employees
WHERE department = 'Finance' OR department = 'Sales';
```

5. Use code with caution.
   This query retrieves all employees belonging to either the Finance or Sales departments, allowing for targeted security updates on their machines.

Retrieve all employees not in IT

SQL

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

6. Use code with caution.
   This query excludes employees from the IT department, streamlining the process of applying security updates to all other relevant personnel.

Conclusion

The SQL queries outlined in this report showcase my ability to effectively leverage filtering techniques to extract targeted data for security investigations. By using WHERE clauses,

comparison operators, logical operators (AND, OR, NOT), and pattern matching (LIKE), I was able to identify potential security risks and gather critical information for further analysis and remediation. These findings will inform future security enhancements and contribute to maintaining a secure environment for the organization.

Submitted by: Junaid Shakoor

Date of Submission: August 1, 2024