SQL Joins for Security Incident Investigation

Overview

In this project, I used SQL joins to retrieve and correlate security-relevant data across multiple tables within a relational database. As part of a simulated security incident investigation, I connected employee, machine, and login activity data to identify affected systems and user activity.

This lab demonstrates practical SQL skills used by security analysts to investigate compromised assets and user behavior.

Scenario

A security incident compromised several machines within the organization. To support the investigation, I was responsible for retrieving related data from multiple database tables using SQL joins.

The database included:

- Employee records
- Assigned machines
- Login attempt logs

Skills Demonstrated

- Relational database analysis
- SQL `INNER JOIN`, `LEFT JOIN`, and `RIGHT JOIN`

- Identifying shared keys between tables
- Correlating users, devices, and login activity
- Security incident investigation fundamentals

Tasks & Approach

1. Match Employees to Machines (INNER JOIN)

Used an **INNER JOIN** to identify which employees were assigned to which machines by matching records on the shared `device_id` column.

Purpose:
Determine which users were associated with potentially compromised machines.

---

2. Identify Unassigned Machines & Users (LEFT & RIGHT JOIN)

- **LEFT JOIN** returned all machines, including those not assigned to any employee.
- **RIGHT JOIN** returned all employees, including those without an assigned machine.

Purpose:
Identify unmanaged assets and users lacking assigned devices—both potential security risks.

---

3. Retrieve Employee Login Activity (INNER JOIN)

Used an **INNER JOIN** to connect employee records with login attempt data using the shared `username` column.

Purpose:
Determine which employees had login activity during the incident window.

Tools & Technologies

- SQL
- MariaDB
- Command-line database environment