Justin Ciocoi

Feb. 15, 2024

CSCI 400 Textbook Notes

Chapter 5: Database and Data Center Security

5.1: The Need for Database Security

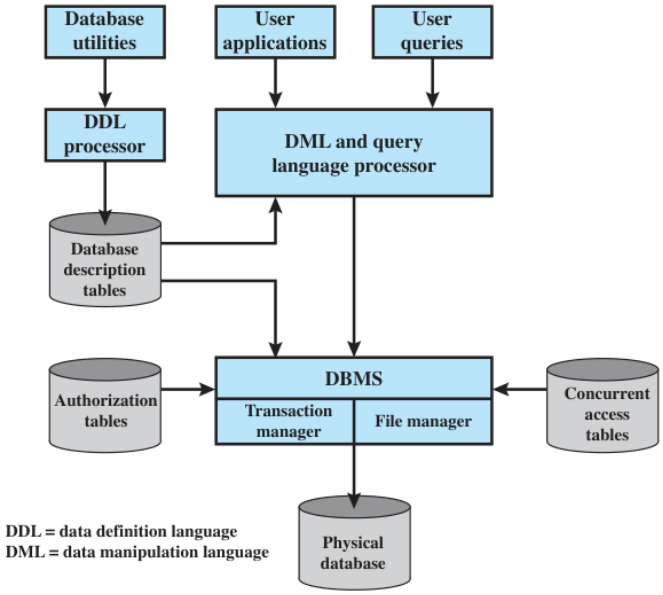
- As reliance on databases has grown, database security has been unable to keep pace with this reliance
- There are several reasons why this is the case
    - There is a dramatic imbalance between the complexity of modern database management systems and the security techniques used to protect these critical systems
    - Databases have a sophisticated interaction protocol, SQL, which is complex in nature
    - Effective database security requires a strategy based on a full understanding of the security vulnerabilities of SQL
    - The typical organization lacks full-time database security personnel
    - Most enterprise environments are made up of a mix of database platforms, enterprise platforms, and OS platforms, which creates a layer of complexity that acts as a hurdle for security personnel
    - There is an increasing reliance on cloud technology to host part or all of the corporate database

5.2: Database Management Systems

- By definition, a **database** is a structured collection of data stored for use by one or more applications
- Accompanying a database is a **database management system (DBMS),** which is a suite of programs used for constructing and maintaining a database
    - The DBMS will be used to offer ad-hoc query facilities to multiple users and applications

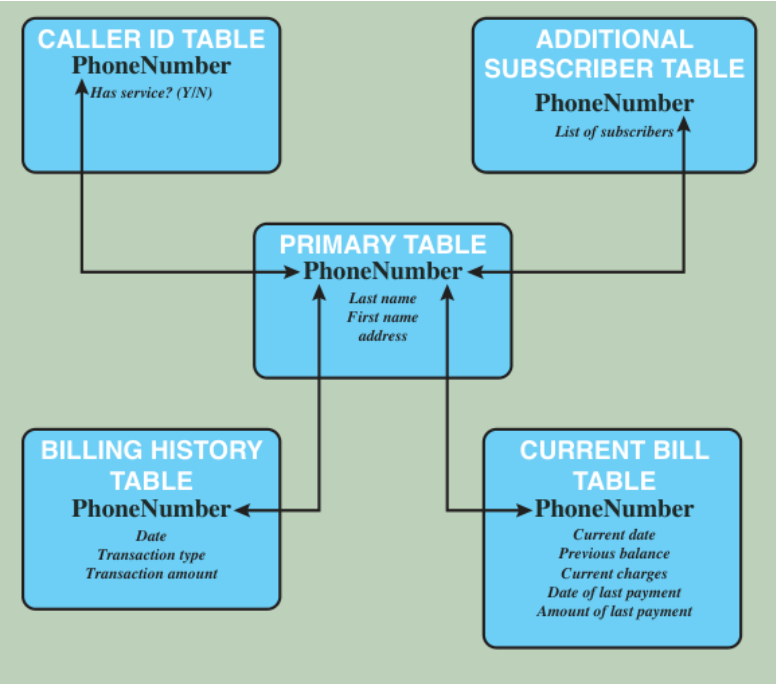
    - A **query language** provides a standard interface with which users and applications are able to interact with the database

- The below diagram shows the typical architecture of an enterprise database and DBMS



5.3: Relational Databases

- A **relational database** is, in its most basic form, a table of data that consists of rows and columns
    - Each column holds a particular type of data
    - Each row contains a specific value for each column
    - Ideally there is one column where every value is unique, providing an identifier or key for that row or entry
- If there is a unique identifier present in all tables, entries from multiple tables can be linked together - this is demonstrated below



- A relational database uses a relational query language to access the database, which allows the user to request data that fir a given set of criteria
    - SQL is an example of such a language
- **Elements of a Relational Database**
    - *Relation*, which refers to a table or file
    - *Tuple*, which refers to a row in the table or an entry
    - *Attribute*, which refers to a column in the table or a field
    - *Primary Key*
        - Uniquely identifies a row
        - Consists of one or more column names
    - *Foreign Key*, which links one table to attributes in another table
    - *View/Virtual Table*
        - The result of a query that returns selected rows and columns from one or more tables
        - Views are often used for security purposes

- Below are a few examples of tables present in a relational database, as well as a view derived from those tables

Department Table

Did	Dname	Dacctno
4	human resources	528221
8	education	202035
9	accounts	709257
13	public relations	755827
15	services	223945

primary key

Employee Table

Ename	Did	Salarycode	Eid	Ephone
Robin	15	23	2345	6127092485
Neil	13	12	5088	6127092246
Jasmine	4	26	7712	6127099348
Cody	15	22	9664	6127093148
Holly	8	23	3054	6127092729
Robin	8	24	2976	6127091945
Smith	9	21	4490	6127099380

foreign key — primary key

(a) Two tables in a relational database

Dname	Ename	Eid	Ephone
human resources	Jasmine	7712	6127099348
education	Holly	3054	6127092729
education	Robin	2976	6127091945
accounts	Smith	4490	6127099380
public relations	Neil	5088	6127092246
services	Robin	2345	6127092485
services	Cody	9664	6127093148

(b) A view derived from the database

5.4: SQL Injection Attacks

- **Structured Query Language (SQL)** is a standardized language to define schema, manipulate, and query data in a relational database
- There are several similar versions of ANSI/ISO standard, but all follow the same basic syntax and semantics
- SQL statements can be used to
    - Create tables
    - Insert, delete, and modify data in tables
    - Create views
    - Retrieve data with query statements
- SQL Injection attacks are one of the most prevalent and dangerous network based security threats in the modern day
- Thee attacks are designed to exploit the nature of web application pages and their underlying SQL databases

- SQL Injection is used to send malicious SQL commands to the database server
- The most common goal of this type of attack is bulk data extraction, but other goals are also possible, including the following
    - Modification or deletion of data
    - Execution of arbitrary operating system commands
    - Launching denial-of-service (DoS) attacks
- Typically, a SQL attack works by prematurely ending a text string in a SQL statement and appending a new command
    - The input is then terminated with the comment symbol such that the SQL server will ignore subsequent text in the command
- The following are different avenues for SQL Injection
    - *User Input*, where SQL is attacked using user-crafted input
    - *Server Variables*, where attackers exploit HTTP to place data directly into HTTP and network headers
    - *Second-Order Injection*, where a user injects a logic bomb into a SQL server that will eventually launch an injection attack from within the system itself
    - *Cookies*, where cookies are altered such that SQL queries are modified
    - *Physical User Input*
- There are generally three types of countermeasure against SQL Injection attacks
    - **Defensive Coding**
        - This includes manual defensive coding practices
        - Parameterized query insertion
        - SQL DOM
    - **Detection**
        - Signature and Anomaly based detection
        - Code analysis

- **Run-Time Prevention**

5.5: Database Access Control

- A database access control determines what portions of the database a user has access to as well as what access rights each user has
- There a range of different administrative policies that can be employed in database access control
  - *Centralized Administration*, where there are a small number of privileged users who may grant and revoke access rights
  - *Ownership-Based Administration*, where the creator of a table may grant and revoke access rights to that table
  - *Decentralized Administration*, which is ownership-based administration where any user who has been granted access rights may also grant and revoke access rights
- **Fixed Roles in Microsoft SQL Server**
  - Server Roles

Role	Permissions
sysadmin	Can perform any activity in SQL Server and have complete control over all database functions
serveradmin	Can set server-wide configuration options, shut down the server
setupadmin	Can manage linked servers and startup procedures
securityadmin	Can manage logins and CREATE DATABASE permissions, also read error logs and change passwords
processadmin	Can manage processes running in SQL Server
dbcreator	Can create, alter, and drop databases
diskadmin	Can manage disk files
bulkadmin	Can execute BULK INSERT statements

  - Database Roles

Role	Permissions
db_owner	Has all permissions in the database
db_accessadmin	Can add or remove user IDs
db_datareader	Can select all data from any user table in the database
db_datawriter	Can modify any data in any user table in the database
db_ddladmin	Can issue all Data Definition Language (DDL) statements
db_securityadmin	Can manage all permissions, object ownerships, roles and role memberships
db_backupoperator	Can issue DBCC, CHECKPOINT, and BACKUP statements
db_denydatareader	Can deny permission to select data in the database
db_denydatawriter	Can deny permission to change data in the database

5.6: Inference

- Inference refers to the process of performing authorized queries and deducing unauthorized information from the legitimate responses received
- This problem arises when the combination of a number of data items is more sensitive than the individual items, or when a combination of data items can be used to infer data of higher sensitivity
- Here we can see a basic example of inference looking at a table, 2 views, and a derived view serving as inferred information

Item	Availability	Cost ($)	Department
Shelf support	in-store/online	7.99	hardware
Lid support	online only	5.49	hardware
Decorative chain	in-store/online	104.99	hardware
Cake pan	online only	12.99	housewares
Shower/tub cleaner	in-store/online	11.99	housewares
Rolling pin	in-store/online	10.99	housewares

(a) Inventory table

Availability	Cost ($)
in-store/online	7.99
online only	5.49
in-store/online	104.99

Item	Department
Shelf support	hardware
Lid support	hardware
Decorative chain	hardware

(b) Two views

Item	Availability	Cost ($)	Department
Shelf support	in-store/online	7.99	hardware
Lid support	online only	5.49	hardware
Decorative chain	in-store/online	104.99	hardware

(c) Table derived from combining query answers

- **Inference Detection**
  - There are two general approaches to inference detection
    - **Inference Detection During Database Design**
      - This removes an inference channel by altering the database structure or changing the access control policy
      - This approach often results in strict access controls that reduce overall availability
    - **Inference Detection at Query Time**
      - This approach aims to eliminate an inference channel during a query or sequence of queries
      - If a channel is detected, then the query will be denied or altered
  - Some inference detection algorithm is needed for wither of these approaches

5.7: Database Encryption

- Typically, the database is the most valuable information resource for any organization
- Therefore, there are usually multiple layers of protection for a database
  - Firewalls, access control systems, DB access control systems

    - Finally, encryption of the data itself is the last line of defense
- Encryption can be implemented at the database level, the record level, or even at the level of an individual field
- There are some disadvantages to the encryption of data in a database, including
    - **Key Management**, since authorized users must have access to the decryption key for the data for which they are authorized
    - **Inflexibility**, such that it becomes more difficult to do record searching if a database is encrypted

5.8: Data Center Security

- A **Data Center** is an enterprise facility which houses a large number of servers and other network and computing equipment
- A single center can have tens of thousands of individual devices
- Generally, redundant power supplies and network connections are in use at data centers
- TIA-492 specifies the minimum requirements for telecommunications infrastructure of data centers
- The following is a simple breakdown of the data center security model

Category	Elements
Data Security	Encryption, Password policy, secure IDs, Data Protection (ISO 27002), Data masking, Data retention, etc.
Network Security	Firewalls, Anti-virus, Intrusion detection/prevention, authentication, etc.
Physical Security	Surveillance, Mantraps, Two/three factor authentication, Security zones, ISO 27001/27002, etc.
Site Security	Setbacks, Redundant utilities Landscaping, Buffer zones, Crash barriers, Entry points, etc.