MANAGING DATA SECURITY IN AN APPLICATION CONTEXT

The goal of database security is to protect data from accidental or intentional threats to their integrity and access.

Data administration is often responsible for developing overall policies and procedures to protect databases. Database administration is typically responsible for administering database security on a daily basis.

Threats to Data Security

- The following threats must be addressed in a com prehensive data security plan:
  - ***Accidental losses, including human error, software, and hardware-caused breaches*** Creating operating procedures such as user authorization, uniform software installation procedures, and hardware maintenance schedules are examples of actions that may be taken to address threats from accidental losses. As in any effort that involves human beings, some losses are inevitable, but well-thought-out policies and procedures should reduce the amount and severity of losses. Of potentially more serious consequence are the threats that are not accidental.
  - ***Theft and fraud*** These activities are going to be perpetrated by people, quite possibly through electronic means, and may or may not alter data. Attention here should focus on each possible location shown in Figure 7-25. For example, physical security must be established so that unauthorized persons are unable to gain access to rooms where computers, servers, data communications facilities, or computer files are located. Physical security should also be provided for employee offices and any other locations where sensitive data are stored or easily accessed. Establishment of a firewall to protect unauthorized access to inappropriate parts of the database through outside communication links is another example of a security procedure that will hamper people who are intent on theft or fraud.
  - ***Loss of privacy or confidentiality*** Loss of privacy is usually taken to mean loss of protection of data about individuals, whereas loss of confidentiality is usually taken to mean loss of protection of critical organizational data that may have strategic value to the organization. Failure to control privacy of information may lead to blackmail, bribery, public embarrassment, or stealing of user passwords. Failure to control confidentiality may lead to loss of competitiveness. State and federal laws now exist to require some types of organizations to create and communicate policies to ensure privacy of customer and client data. Security mechanisms must enforce these policies, and failure to do so can mean significant financial and reputation loss.
  - ***Loss of data integrity*** When data integrity is compromised, data will be invalid or corrupted. Unless data integrity can be restored through established backup and recovery procedures, an organization may suffer serious losses or make incorrect and expensive decisions based on the invalid data.
  - ***Loss of availability*** Sabotage of hardware, networks, or applications may cause the data to become unavailable to users, which again may lead to severe operational difficulties. This category of threat includes the introduction of viruses intended to corrupt data or software or to render the system unusable. It is important to counter this threat by always installing the most current antivirus software as well as educating employees on the sources of viruses. We will discuss data availability in Chapter 8.
- Two critical areas that strongly support data security are client/ server security and Web application security. We address these two topics next before outlining approaches aimed more directly at data security.]

Establishing Client/Server Security

- Database security is only as good as the security of the whole computing environment. Physical security, logical security, and change control security must be established across all components of the client/server environment, including the servers, the client workstations, the network and its related components, and the users.

- Server security

SERVER SECURITY In a modern client/server environment, multiple servers, including database servers, need to be protected. Each should be located in a secure area, accessible only to authorized administrators and supervisors. Logical access controls, including server and administrator passwords, provide layers of protection against intrusion.

Most modern DBMSs have database-level password security that is similar to system-level password security. Database management systems, such as Oracle and SQL Server, provide database administrators with considerable capabilities that can provide aid in establishing data security, including the capability to limit each user's access and activity permissions (e.g., *select*, *update*, *insert*, or *delete*) to tables within the database. Although it is also possible to pass authentication information through from the operating system's authentication capability, this reduces the number of password security layers. Thus, in a database server, sole reliance on operating system authentication should discouraged.

- Network security

NETWORK SECURITY Securing client/server systems includes securing the network between client and server. Networks are susceptible to breaches of security through eavesdropping, unauthorized connections, or unauthorized retrieval of packets of information that are traversing the network. Thus, encryption of data so that attackers cannot read a data packet that is being transmitted is obviously an important part of network security. In addition, authentication of the client workstation that is attempting to access the server also helps enforce network security, and application authentication gives the user confidence that the server being contacted is the real server needed by the user. Audit trails of attempted accesses can help administrators identify unauthorized attempts to use the system. Other system components, such as routers, can also be configured to restrict access to authorized users, IP addresses, and so forth.

- Application Security Issues in Three-Tier Client/Server Environments
  - In a three-tier environment, the dynamic creation of a Web page from a data base requires access to the database, and if the database is not properly protected, it is vulnerable to inappropriate access by any user.
  - Also of interest is privacy. Companies are able to collect information about those who access their Web sites. If they are conducting e-commerce activities, selling products over the Web, they can collect information about their customers that has value to other businesses.
    - If a company sells customer information without those customers' knowledge or if a customer believes that may happen, the company might be violating ethical and privacy standards and, depending on the context, also be acting in a way that is in violation of the law.
  - Protecting and securing static web pages or alike info

Figure 7-26 illustrates a typical environment for Web-enabled databases. The Web farm includes Web servers and database servers supporting Web-based applications. If an organization wishes to make only static HTML pages available, protection must be established for the HTML files stored on a Web server. Creation of a static Web page with extracts from a database uses traditional application development languages, such as Visual Basic.NET or Java, and thus their creation can be controlled by using standard methods of database access control. If some of the HTML files loaded on the Web server are sensitive, they can be placed in directories that are protected using operating system security, or they may be readable but not published in the directory. Thus, the user must know the exact file name to access the sensitive HTML page. It is also common to segregate the Web server and limit its contents to publicly browsable Web pages. Sensitive files may be kept on another server accessible through an organization's intranet.

  - Protecting and securing dynamic web pages

Security measures for dynamic Web page generation are different. Dynamic Web pages are stored as a template into which the appropriate and current data are inserted from the database or user input once any queries associated with the page are run. This means that the Web server must be able to access the database. To function appropriately, the connection usually requires full access to the database. Thus, establishing adequate server security is critical to protecting the data. The server that owns the database connection should be physically secure, and the execution of programs on the server should be controlled. User input, which could embed SQL commands, also needs to be filtered so that unauthorized scripts are not executed.

Note that dynamic pages have direct access to database thus more need for security

  - User authentication security

Access to data can also be controlled through another layer of security: user-authentication security. Use of an HTML log-in form will allow the database administrator to define each user's privileges. Each session may be tracked by storing a piece of data, or cookie, on the client machine. This information can be returned to the server and provide information about the log-in session. Session security must also be established to ensure that private data are not compromised during a session because information is broadcast across a network for reception by a particular machine and is thus susceptible to being intercepted. TCP/IP is not a very secure protocol, and encryption systems, such as the ones discussed later in this chapter, are essential. A standard encryption method, Secure Sockets Layer (SSL), is used by many developers to encrypt all data traveling between client and server during a session. URLs that begin with https:// use SSL for transmission.

Additional methods of Web security include ways to restrict access to Web servers:

- Restrict the number of users on the Web server as much as possible. Of those users, give as few as possible superuser or administrator rights. Only those given these privileges should also be allowed to load software or edit or add files.
- Restrict access to the Web server, keeping a minimum number of ports open. Try to open a minimum number of ports, preferably only http and https ports.
- Remove any unneeded programs that load automatically when setting up the server. Demo programs are sometimes included that can provide a hacker with the access desired. Compilers and interpreters such as Perl should not be on a path that is directly accessible from the Internet.

- Data Privacy
  - Goes into things about data privacy but all common knowledge stuff
  - Page 362