DATABASE SECURITY MECHANISM

Dorin Iordache
Lecturer eng., Romanian Naval Academy "Mircea cel Bătrân"
Fulgerului nr.1, Constanta, 8700, Romania
email: diordache@seanet.ro

Abstract
Database security was and still is an important objective for information security. The potential security threats posed by databases and computers networks is significant and ever increasing. Threats for operational safety can be caused by both unauthorized and authorized activities of some database users. Institutions have to implement some security mechanisms in their environment. Nowadays, database file ought to be protected. For this reason, it is important to track access to all data that is processed via database management systems and to classify and code that such information. Also, it is necessary to implement a security mechanism at the database files level. The recent expansion in communication and data distribution networks has resulted in a range of new security threats.

Key words: database security, authentication, security, information security.

INTRODUCTION

Database security is a branch of information security management, but has it's own specific problems. Now, more than ever before, information is flowing in all directions, with varying degrees of importance and in a number of different forms. These include public information and news as well as economic, military and financial information. Information can be transmitted in several different ways, from magnetic strips to smart cards. As a result, it is important to adopt basic security measures in order to protect computer operating systems and the data stored within them.

Databases introduce a number of unique security requirements for their users and administrators. On one hand, databases are designed to promote open and flexible access to data. On the other hand, it's this same open access that makes databases vulnerable to many kinds of malicious activity.

The security of data collections, which are contained and manipulated using a specialized system, otherwise known as "databases" (name given by H.Ullman), has become increasingly complicated. [ULLMAN1976]

Security information in database environment contents the following issues[TIPT1997]:

- database files security;
- information secrecy;
- users authentication;
- audit users database actions;

The complexity of the design and implementation of a database security system depends on several factors, including[SANHU, WOOD1976]:

- System users;
- Data structure;
- How widely the database will be circulated, in the case of distributed databases;
- What the consequences would be of loosing information;
- Specification, modelling and verification of data security level;
- Level of integration with computer systems, etc

The problematic of user's access control it is demonstrate in figure 1[CAST1994].

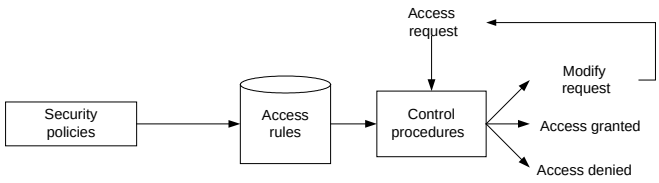

Fig. 1. User's access control

The most commonly used database systems are Microsoft Access, FoxPro and Visual Basic database. These systems are not as secure as programmes such as SQL Server, Oracle, Sybase do. However, Microsoft Access and other similar database system are cheap and therefore affordable and popular.

It is known that the Microsoft Access database files are protected with primitive procedures. The procedure is based on user name and password. This authentication mechanism was hacked. That is the reason to implement own security mechanism.[IORD2001]

In such systems, the security problems are:

- There is no control mechanism to ensure the quality of the information generated by users other than DBMS
- There is no powerful mechanism to protect database file
- Data is not protected against inside or outside DBMS or the application
- It is impossible to control user access to the database
- There has been no complete audit of database functions
- There is no control mechanism to ensure the quality of the information generated by users other than DBMS
- Data is permanently available, other than being controlled by OS;
- There is no automatic backup mechanism

Therefore, it is necessary to develop a security mechanism for the access database in order to improve its security level.

DATABASE SECURITY OBJECTIVES

Therefore, it is compulsory to implement some security mechanism for database system that I previously mentioned. The mechanism has to solve the lack of security partial or complete.[SCOTT2003, DENN1982]

I suggest making the following design changes to improve database security and address the problems I have previously outlined:

a.The user authentication mechanism should include the following aspects:

- The user authentication is based on user name and password;
- The password is coded with a specific procedure;
- Users ought to have a specific level of security;
- The users should be grouped together to share system tasks

Every group has specific options that are selected through the application menu. The system's users and their jobs are shown in figure 2.

Every time the users account is accessed, details should be automatically stored on file. The possible users are:

- User – common users;
- Auditor – audit administrator, responsible with audit and log files;
- DBA – database administrator;
- SA – security administrator;
- Programmer – applications programmer;
- Application manager – with full rights on application program.

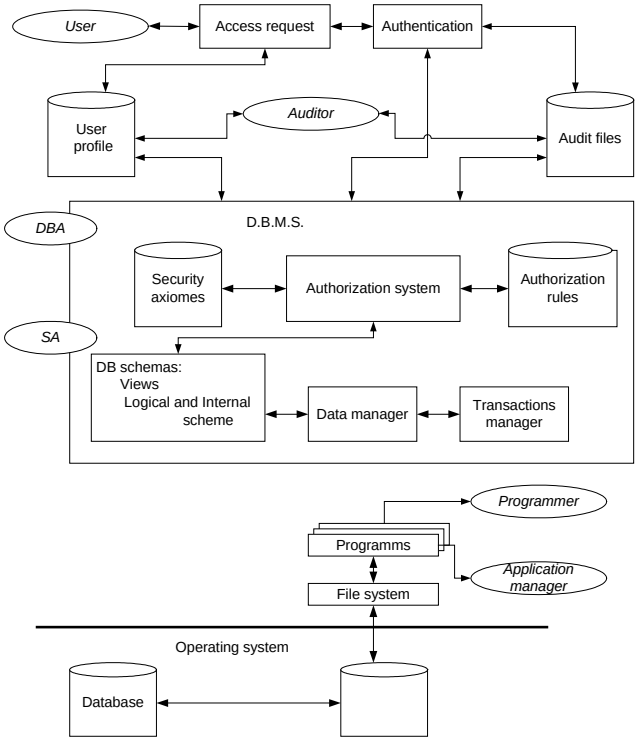

Fig. 2. Database security mechanism and their users

b.Coded procedures at the file or tuple level

- Data should be given a security level. For example: Top Secret, Secret, Classified, Unclassified;
- The following information should be coded: information about the user: username, password, security level; details of database use activity by user; details about the level of security ie. high etc; users login and logout activity.

c.The most important database files should be code protected. If it is possible the entire database file will be coded.

SECURITY MECHANISM

In order to meet the objective outlined above, the database should have the following structure:

- Specific mechanism to code the database files;
- Security mechanism to solve users request;
- Tables for: users, groups and their security level;
- A 'code and decode' procedure for each field;
- Specific menus which are generated according to users or user group security levels;
- An audit mechanism

Example: Security mechanism database structure is:

Users : UserID, Username, Password, User's group and user's security level;

Group: GroupID, GroupName and GroupDescription;

SecurityLevel: Sec_LevelID, Sec_LevelName and Sec_leveldescription.

In that situation the security database mechanism contains:

a.User authentication mechanism with the following data:

Users, Groups, Security level. Level of information secrecy. Users should be given a secrecy level, like I previously mentioned. The secrecy levels should be defined in relation to the content of the information that can be accessed

b. The menu components

Each user should have specific group and security level and is similar with the figure 2 users.

c. Audit mechanism

Every database access request should be recorded in a single table. For security reasons, I suggest this file is coded with another algorithm. It's important to record the login and logout date and time and whether access was granted or rejected.

d. Implementation of a database security models

Many security models have been proposed in the literature. Some of them operate for the protection of information in operating systems and in database systems, such as: Access matrix, Take-Grant, Action-Entity, Wood et.al., Sea View, Jajodia-Sandhu, Smith-Winslett, etc.

e. Security files mechanism

Implementation of a security files mechanism in accordance with user's function shown in figure 2.

The security mechanism structure and functions are described in figure 3.

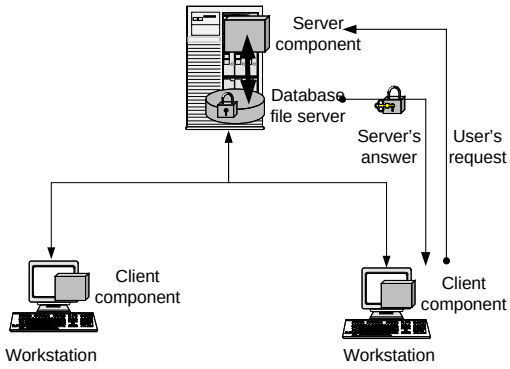

Figure 3. Security mechanism structure and functionality

The mechanism shown in figure 3 solve the has the following important jobs:

- user authentication;
- code and decode the database files.

We can use visual cryptography mechanism, in order to improve the user's authentication mechanism.

After the mechanism decides if the user requests are the valid one, it starts the decode process for database files in accordance with the scheme describe in figure 3.

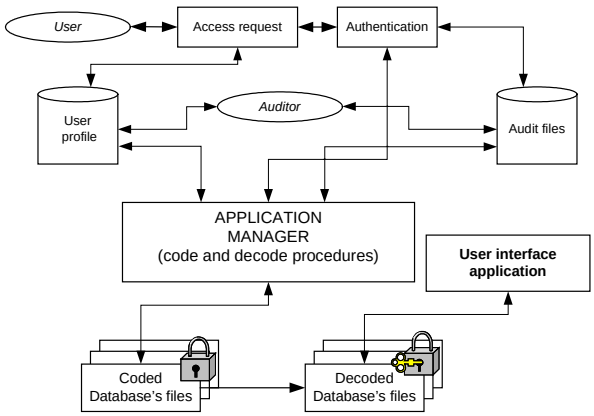

Figure 4. Code and decode database files mechanism

After the user gain the right to access the database, the application manager decodes the database files. The decoded database files are stored in other place than the coded files, because of security reasons.

The security of database will be increase in that way. But, in the same time, the mechanism has one major disadvantage: to decode and code the database files the user will wait a period of time. During that process, coding and decoding the files, the database system is not available.

The decode operation will take place once the first user want to access the database file. The mechanism will keep the database file uncoded during user working time. If other user requests to work on the same database, the application manager gives the

access after user authentication without decode operation. The database files are already decoded.

But, the security administrator have to specify the period of working time in order to eliminate the possibility the database file to stay in system uncoded.

When the period of working time is off the user's access is blocked and the database files are going to be coded.

I tested and I measured the time for code and decode using files with different dimensions.

For testing, I use the following dates:

-hardware: Pentium 600 Mhz, 256 Mo Ram;

-software: C program , with Polling-Hellman algorithm for coding and decoding files; the gmp library, for operating with huge numbers (it was used in order to implement the Polling-Hellman algorithm).

-operating system: LINUX SuSe 8.0;

For test, I use six files with different dimension.

The used files dimension is shown in Table 1.

Table 1. The used files dimension

File name	Dimension [byte]
File1	1,031,956
File2	3,617,953
File3	5,983,789
File4	57,805,051
File5	1,011,100,114
File6	543,722,827,124

The time obtained for decode operation is shown in figure 4. I dignify the time was determined in previously conditions.

Figure 4. Experiment results (time in seconds)

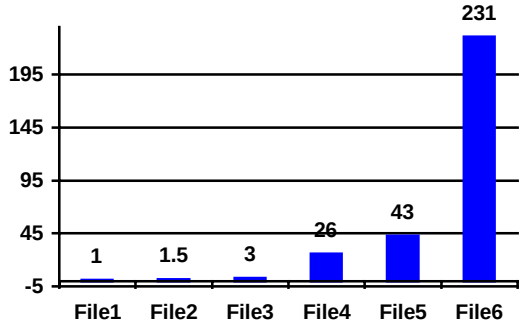

Figure 4. Experiment results

I consider the almost 4 minutes is reasonable for one user to wait to work with database.

CONCLUSIONS

The security of DBMS can be improved by implementing:

- An application client-server to code and decode database files;
- A user authentication mechanism;
- An application menu for user options, according to their functions;
- The coding of key internal and external procedures for secure information;
- Tracking user access through an audit mechanism;
- A non-referential database structure;

Securing an information system can be a costly process as it involves investing both time and money in research and training. However, by adopting these procedures, the security of the database can be improved without incurring any significant expense.

This security mechanism solves the following problems:

- activate/deactivate databases;
- code/decode database files in accordance with users' rights;
- grant/revoke users login;
- collect data about users' activity;
- restrict time period when the databases are available;
- possibility to manage more than one database.

The problems that I previously mentioned are advantages but, there is also disadvantages, at the same time. The most important disadvantages consist of long time action for coding and decoding database files. Therefore, the proposal database security mechanism it is very useful in case of small databases. The database security increases in these systems.

It is compulsory to program the application with other software than database software. For example, you it can be use C++, JAVA or something like that. Hereby, the users can't access the database through Access DBMS, VBA or other software because the database files will be coded. This solution has great results with small database files. If the dimension of database file is huge time for code and decode is great. This is a disadvantage. But, the database security is improved and the database management is secure.

REFERENCES

AMO1994 Amoroso E. 1994, *Fundamentals of Computer Security Technology*, Prentice Hall International Editions

DENN1982 Denning D.E. 1982, *Cryptography and Data security*, Addison-Wesley,

DION1981 Dion L.C. 1981, *A Complete Protection Model*, in proceedings IEEE Symp. on Security and Privacy, Oakland, CA

HRU1976 Harrsion M., Ruzzo W.L., Ullman J.D. 1976, *Protection in Operating System*, Communications of the ACM, vol 19

IORD2001 Iordache D. 2001, *Ameninţări asupra securităţii sistemelor de calcul*, Buletinul ştiinţific al ANMB nr. 3-4

IORD2001 Iordache D. 2001, *Detecţia intruşilor într-o reţea UNIX (LINUX)*, Buletinul Ştiinţific al ANMB, nr. 1

IORD2001 Iordache D. 2001, *Modele de securitate pentru bazele de date*, Buletinul ştiinţific al ANMB nr. 2

TIPT1997 Krause M., Tipton H. 1997, *Handbook of Information Security Management,* CRC Press LLC

CAST1994 Castano S., Fugini M.G., Martella G., Samarati P., 1994, *Database Security*, Addison-Wesley Publishing Company

SANDHU S. Oh, R.Sandhu, *A Model for Role Administration Using Organization Structure,* http://www.list.gmu.edu/confrnc/sacmat/sacmat02-oh.pdf

SCOT2003 Scott N. 2003, *Database security: protecting sensitive and critical information,* http://www.infosyssec.org

TSICH1977 Tsichritzis D., Klug A. 1977, *DBMS framework report of the study group on database management systems,*AFIPS Press

ULLM1980 Ullman J.D. 1980, *Principles of Database Systems*, Computer Science Press

WOOD1979 Wood C., Summers R.C., Fernandez E.D. 1979, *Authorization in Multilevel Database Models*, Information Systems Pergamon Press, vol. 4