Journal of Applied Science and Engineering Innovation, Vol.2 No.6, 2015, pp.220-222
ISSN (Print): 2331-9062
ISSN (Online): 2331-9070

Implementation and Design of Structure of Multilevel Secure Database System

Yuyang Du[1],Haiyan Zhao[1] , Mingshun Xing[1], and Ran Li[1]
[1] *Department of Network service, Xi'an Communication Institute, Xi'an 710106, China*

Abstract: This paper introduces the design of multilevel secure database management system and offer relevant flow chart and core code, in order to provide technical supplies to the future database.

Keywords data security; Implementation and Design; Multilevel Secure Database System

INTRODUCTION

A database is an integrated and organized collection of logically related records or files or data that are stored in a computer system which consolidates records previously stored in a separate files into a common pool of data records that provides data for many applications. The data is typically organized to model aspects of reality in a way that supports processes requiring information, such as modelling the availability of rooms in hotels in a way that supports finding a hotel with vacancies.

DESIGN OF SYSTEM

The system adopts three structure layers. The top layer is the user service layer and it provides a friendly interface for users to browse information. The middle layer is the application layer services, which provides standard database access for database applications and controls the access to database, being completed by the database proxy server. The lowest layer is the data service layer, using the function of data definition, storage, backup and retrieval and it is completed by the database server.

In order to meet the needs of secure access and introduce database security proxy, the system can be divided into client and database security proxy. The client proxy includes application program interface module and communication surface module. Database security agent comprises a communication interface module, and requests analysis module, accessing and inferring control module control module, audit module and proxy access module. The system architecture is shown in Figure1.

The application program sends user's requests and the user's identity information through the client interface. Firstly, the database agent analyzes the query request when receiving the news, through the SQL analytic and transfer to the discretionary access control module. When discretionary access control check passes, the user access request is transmitted to the mandatory access control module for mandatory access check. After compulsory access control checks, user access request will be transmitted to the inference control module, checking whether the user can reason out unauthorized information according to information obtained by this visit and prior knowledge. If the reasoning control is checked by inspection, the user is allowed to target the database access module through a proxy, or he/she will be informed that the client that access is denied and disconnected. Whether the user can make it, the audit module will still records.

Database security proxy is bridged between the applications and the database, which plays a role of database middle ware. Agent system provides a standard database access interface for the database applications and manages the requests for the access to the database, which completely overcomes the disadvantages of traditional client/server model and has the advantages of good reusing and managing, flexibility and easy maintenance etc.

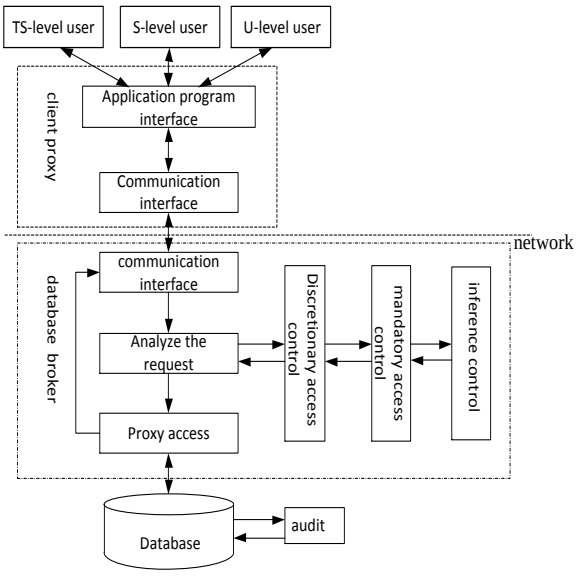

Figure 1 System architecture related modules schematic diagram

SYSTEM IMPLEMENTATION

System processes as shown in figure2.

Firstly the system will initializes, treatment process begins processing the information, which is sent by the client. If the information received is the end, the process is finished. If the operation is a user's request information database, the request analysis module is called to analyze the user request, then the analysis results and the user identity information will pass to control module, if the access control checks do not pass, then the user is returned to refuse query information and ends the process. If the access control checks pass, the information will be transmitted to inference channel control module, after the channel control module detects, decision results obtained. Results are divided to allow access and deny access, if it is allowed access, the database of user information will transmit to proxy access module. The module will be completed by proxy access module. If it is denied access, client sends denied information to users and ends the treatment process. Whatever the outcome, the audit module should be recorded, when audit module discovered an illegal operation, the audit module will terminate users access and alarm.

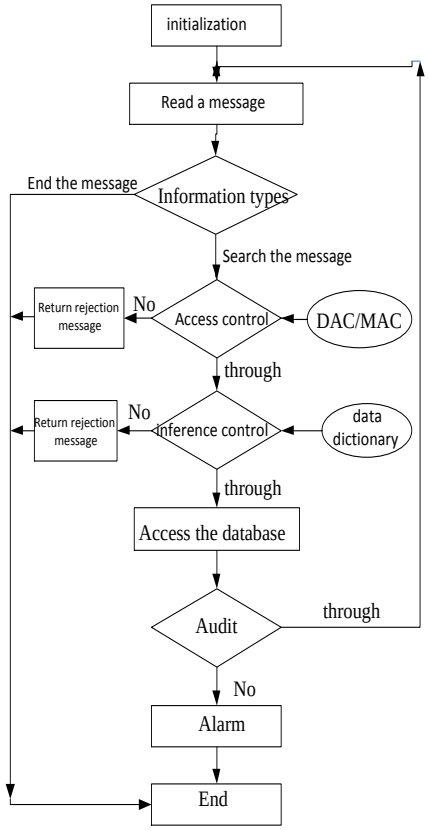

Figure 2 The flow chart of the system

The flow of the core code is as follows:

```
#define DB_QUIT                 1
#define DB_CLOSE    2
#define DB_QUERY    3
#define DB_FAILED   4
#define DB_PASS                 5
#define DB_DENY                 6
#define DB_ALERT    7

extern void initDatabase();
externint Receive(Message * msg);
externintReadReceive(char * info);
extern void userQuit(char * user);
externintAccessCheck(char * user,char * pass);
extern void SendToUser(char * user,char * msg);
externintinferenceCheck(char * user);
externintAccessToDataBase(char * user);
externintAuditUser(char * user);
externintAlertMessage();
extern Message msg;

voidDatabaseStart()
{
        initDatabase();//initialization data
        while (Receive(&msg)!=DB_ERROR)
        {
                int res;
                intinfo_type;

        info_type=ReadReceive(msg.info);//  read a
message
                if (info_type==DB_QUIT)// if the
type of information is the ending type
                {
                        userQuit(msg.user);// end
                }
                else if (info_type==DB_QUERY)//
if the type of information is the searching type
                {

        res=AccessCheck(msg.user,msg.pass);//
Check the access
                        if (res==DB_DENY)// if it
fails the checking
                        {

        SendToUser(msg.user,"access       deny!");//
send to user "access deny!"

        userQuit(msg.user); //  end
   }
                        else if (res==DB_PASS)//
if it passes the checking of visiting
                        {

        res=inferenceCheck(msg.user);//  infer  the
checking channel
                                if
(res==DB_DENY)// if the checking fails
                                {

        SendToUser(msg.user,"access       deny!");//
send to user "access deny!"
```

```
        userQuit(msg.user);//
        end
    }
                                    else if
(res==DB_PASS)// if it passes the checking channel
                                    {

        AccessToDataBase(msg.user);// access to
database

        res=AuditUser(msg.user);// audit
                                    if
(res==DB_ALERT)// if the auditing user has illegal
operation
                                    {

        AlertMessage();//warming

        userQuit(msg.user);// end
                                    }
                                }
                            }
                        }
        }
    }
```

CONCLUSION

Multilevel secure database management system has been researched by a large number of foreign researchers. Although the multilevel secure database prototype system in many high security has been achieved, there are many problems which have not been solved yet. This paper introduces the design of a multilevel secure database management prototype system, completing the system structure. We sincerely hope that this research can provide some technical supports for the database development in the future.

ACKNOWLEDGMENT

This work is supported by the National Science Foundation of China(61179002), Shaanxi Natural Sciences Foundation(2011JM8030).

REFERENCES

A Evfimievsk, J Gehrke, R Srikant. Limiting privacy breaches in privacy preserving datamining [J] . In Proceedings of the 22nd Symposium on Principles of Database Systems, ACM Press, 2003: 211- 222

Hinlce Thomas H.InIerence Aggregation Detection in Database Management Systems[C]. In:Pro IEEE Symp Research in Security and Privacy, Oakland, CA, New York. 1988:96～106

J Domingo- Ferrer. Advances in inference control in statistical databases: An overview in inference control in statistical databases: from theory to practice [J]. LNCS2316, Springer- Verlag, 2002:1-7

Jajodia S, Meadows C. Inference problems in multilevel secure database management systems[A]. Abrams M, Jajodia S, Podell H, eds. Information Security: An Integrated Collection of Essays[C]. Los Alamitos: IEEE Computer Society Press, 1995，570-584

L Wang, D Wijesekera, J Sushi.l Cardina lity-based inference control in sum - only data cubes[J]. In Proceedings of the 7th European Symposium on Research in Computer Security, 2002

Qu, G., Zhang, J., & Zhao, W. (2014). Primary Study on the Reformation of Production Engineering Practice Course System of the Petroleum Engineering. Journal of applied science and engineering innovation, 1(3), pp.212-216.

S. R. izv,. J.H. aritsa. Maintaining data privacy in association rule mining[J]. In Proceeding s o f the 28th International Conference on Very Large Data Bases, 2002: 682- 693