Send Orders for Reprints to reprints@benthamscience.ae

A Design of Multi-level Structure Security Architecture on Database Application System

Haibin Ma[1,*] and Chunyan Xie[2]

[1]*College of Vocational and Technology, Hebei Normal University, Shi Jiazhuang, China*

[2]*College of Information Technology, Hebei Normal University, Shi Jiazhuang, China*

Abstract: Database application system mainly exists three aspects of insecurity: insecurity of data storage, fragilities of database application, the risk of secret data in the database. Refer to these three factors, This paper presents a multi-level structure security system, which includes decision-making level, code level, and data encryption level. This design enhances security of database site, security of applications accessing to the database and confidentiality of the data.

Keywords: Database application system, fragilities of database application, multi-level structure.

1. INTRODUCTION

The design of security system of database application system on multi-level structure database application system (DBAS) is a kind of application software to solve some practical problems based on database on the support of database management system (DBMS). It includes database, application program accessing to database and client using DBAS [1-3].

Because of the open of internet and insecurity of communication protocol, Data transmitted on Internet is fragile to be damaged, stolen, distorted, displaced and lost. In the database, data is stored in huge amount and used by too many clients, so the security is more and more important. Nowadays, the invasion to Web database runs wild, such as SQL injection, cross-site scripting attacks and unauthorized access and so on. All invasions may attack the database by passing around the foreground security system [4].

SQL injection usually attacks the program to make the system abnormal, which is a common problem. Another problem is accessing unauthorized to database by running around the security system, which makes some secret data divulged [5].

2. THE ANALYSIS OF INSECURITY FACTORS

There are mainly 3 factors: insecurity of database site, fragilities of database application, and dangers of secret data in the database. As shown in Fig. (**1**).

Refer to the problems of Peripheral security of DBAS, this paper proposed a new security system. It is presented that the security system should include 3 levels, that is, one level to protect the security of database site, one level to protect the security application program for access and the last level to protect the confidentiality of data in database. The position of security system is as shown in Fig. (**2**).

3. THE FUNCTION OF SECURITY SYSTEM

According to the need of the security, the structure could be divided into 3: Decision level, code level, and data encryption level. These levels could protect the weak section of DBAS, their function is as follows:

1) Decision level

At First, this level decodes the data packets reaching to the underlying of the database site. Then it tests abnormal packets as rules, which are just the description of existing abnormal packet. The abnormal packets would be analyzed by data mining method, and sent to system administrators [6-9]. Then the rules would be renewed to intercept this kind of attack. Intelligent mining module would diagnose and analyze how to resist similar Invasion in the future, and keep step with the evolving fragilities and attacks.

2) Code level

This level controls the security of accessing to database by code. If the code is not safe enough, the invaders will enter the background by the way of client machines to distort the data arbitrarily. Most DBAS has the dangers of insecurity.

3) Data encryption level

Many invaders could steal even distort the data by using operating system or the username with password of the database site, that means they run around the security monitor of decision level and code level. Beside that, the administrator of database could access to the database by his authority. All the relational databases are the type of port, so many clients could access to the database by connecting the analysis tools with the port, and run around the security

1874-4443/14

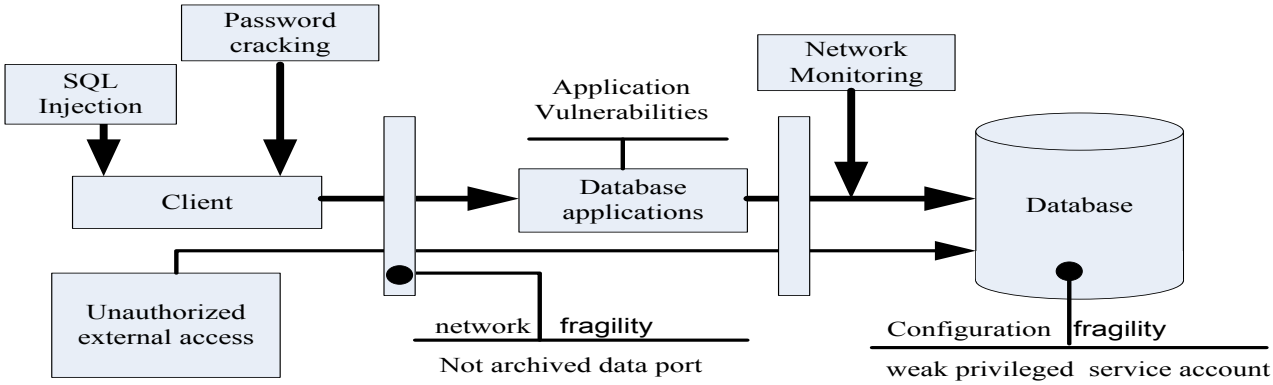

Fig. (1). Database application system security threats.

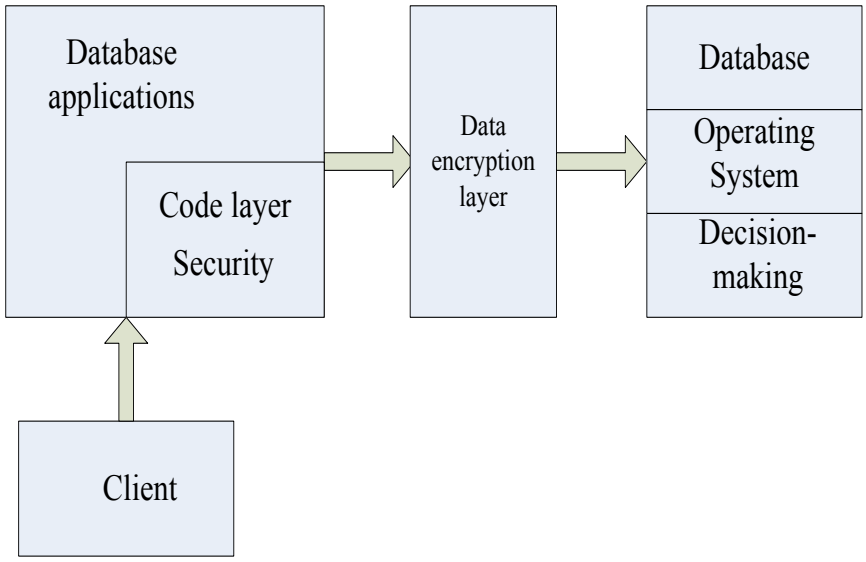

Fig. (2). Database application system security structure.

system of operating program. That is dangerous, and even worse, all the database system has a public account and password to access. Therefore, encryption for database is an effective method to avoid the loss and distortion.

3.1. Decision Level

Decision level decodes the original data packet reaching the site according to protocol solution, and stores all the data. Then it is analyzed by the data mining module and abnormal analysis module. If the result is reasonable, the data will be sent to the operating system, otherwise, the data will be sent to the system administrator to make new rules to prevent similar attacks [10, 11].

Based on the analysis of network packets, original network packets are used as data source for attack analysis. From that, some useful information could be derived, and the similar attack could be prevented after matching it with known attack signatures or with normal network behavior to judge its danger. With this protection, security of database site would be better. The structure of decision level is as shown in Fig. (**3**).

Decision level is in charge of monitoring the data packets on internet. Decision level should catch the data packets on the physical line road, and then send them to processing program. The decoding part is consisted of packet capture, packet decoder, preprocessor, and retrieval engine. Packet capturer catches the data packets, and then sends them to packet decoder for decoding. After making sure the protocol and feature, Packet decoder stores data in database. Preprocessor program used to check or modify the packet, so as to correctly interpret for the behavioral of detection engine.There are many attacks that can not be detected by the characteristic pattern matching method. So this section is used to detect non-feature-based attacks and standardization of traffic.

The function of retrieval engine is a rule analysis and a feature test. Retrieval engine analyzes the rules in the rule bank to build attacking feature. After analyzed by data

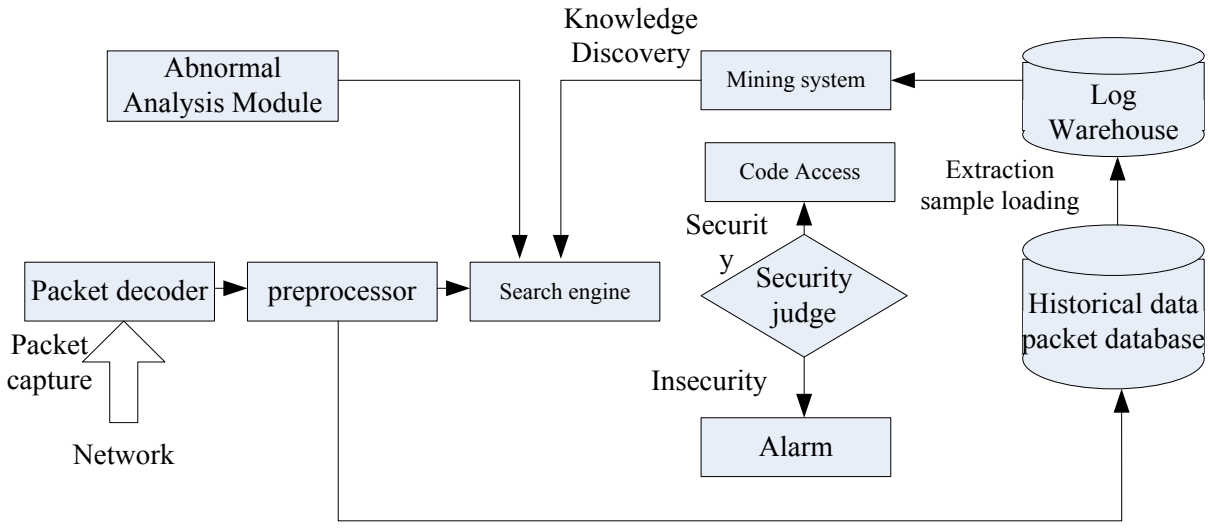

Fig. (3). Decision-making structure of database security system.

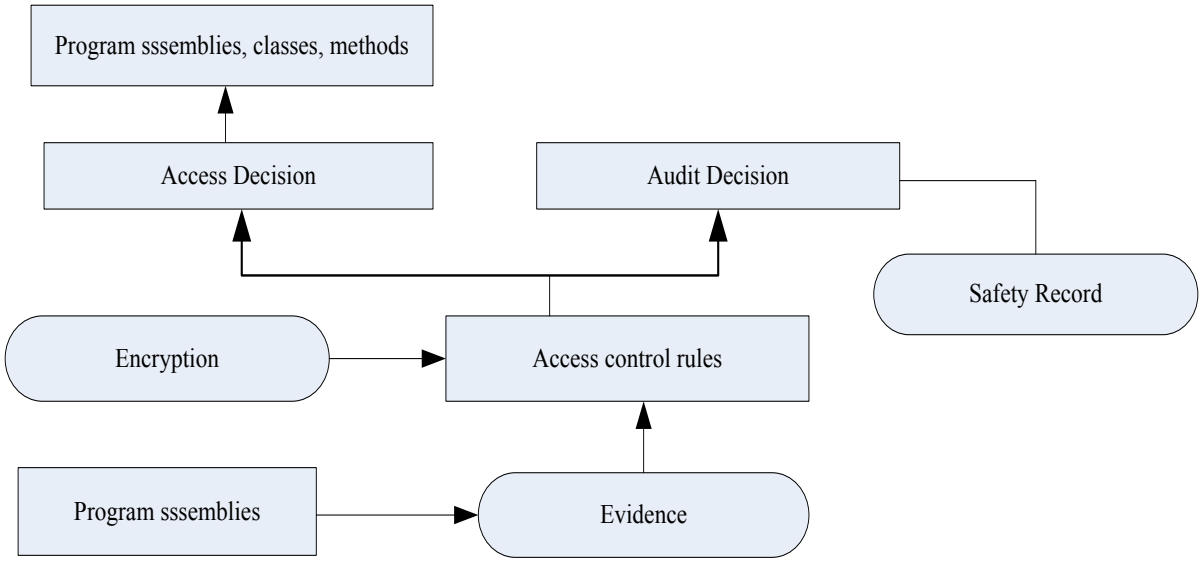

Fig. (4). Authorized access control model of code.

mining module, the abnormal flow and structure would be sent to abnormal analysis module for calculation its rationality.

3.2. Code Level

In the database application system, the security of code access to the database by is an important issue. According to the servicing security system, the clients could login in by using authorized code, while there come some problems. Many clients could get code from multiple sources, and some of these sources may be unreliable. Sometimes some code may contain errors or malicious code to operate some programs unknown by client. To solve this problem, controlling module based on CAS (code access system) is in charge of access controlling in code level of object in different sizes. This paper uses authorized service layer defined free to provide security service for code series.

The security of the entire. NET flat is based on two aspects: Type security of Managed Code and common language runtime (CLR). CLR is the virtualized execution environment, and one of its advantages is that a new security model can be developed transcending the underlying operating system. CLR achieves Safe execution model independently from the host platform, and is particularly suitable for dynamic security Systems. In this model, the authority is granted to the code rather than the users. Before loading an assembly, CLR will collect evidence of the source code and contact the assemblies Memory with evidence together. With its help, the security system will grant correspondent permissions to the new code. CLR runs the evidence by a security strategy. The Security strategy accepts the evidence as input and produces a permission set as output. This security model is called Code Access Security (CAS). The model is as shown in Fig. (**4**).

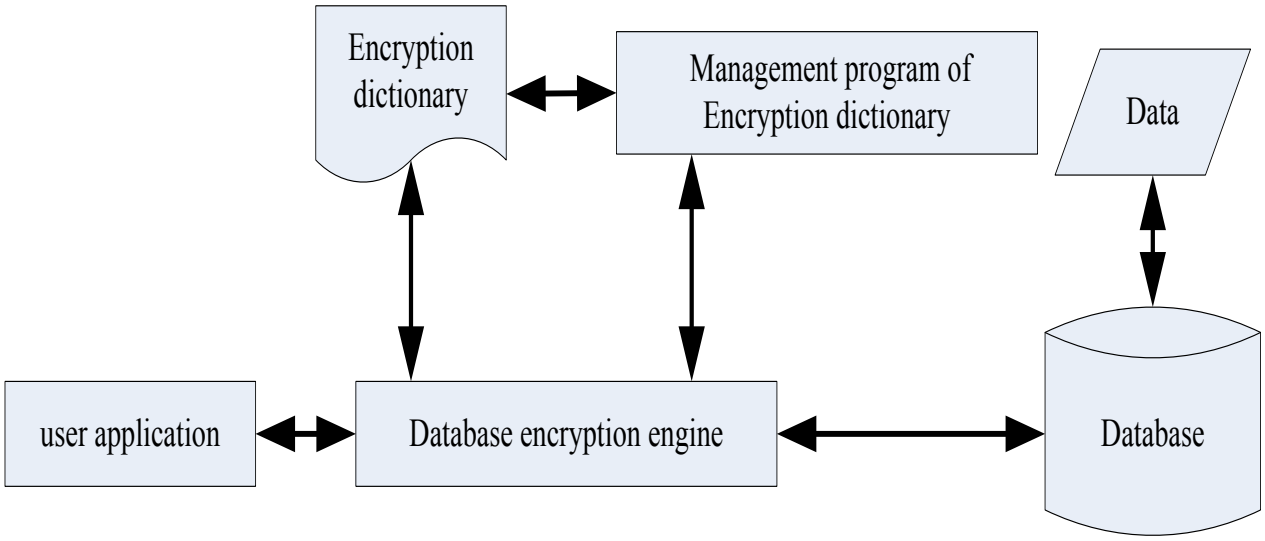

Fig. (5). Encryption system structure.

First it is necessary to collect of credential information of program assembly. Specified code is from the evidence that depends on the source of code and itself data of program assembly. CLR offers seven types of evidence which has developed source of code and tell who write this code. Evidence information was given to the encrypted portion of the access control rules, access decision verdict in accordance with the access control rules and control to access to the assemblies, classes and methods. Decision section divides the security of events into different levels, such as ignoring, warning or alert. It audits record of decision and alerts managers to deal with illegal access in a timely.

3.3. Data Encryption Level

The design of decision level and code level could enhance the security of host computer placing database and access security to database by application program. While data in database is not safe enough, the hikers might steal even distort the data by using operation system, the administrators of database could use the database as please, and something like that. So Data encryption is an effective method to protect data information. The core of information security is the security of data, that is, database encryption is the core of security issue.

Typically, database application (client/server model) accesses to remote database system by calling the database programming interface. And general interface allows applications to use Structured Query Language (SQL) to query the data. A new middleware is designed in this paper. It is an interface program for security and confidentiality between the application and the database management systems. With its help, the encryption and decryption of data and the function for key management is Completed.

The design in this paper did some essential transformation to normal database access process. The interface program for security and confidentiality captures SQL submitted by application program, and encrypts data according to the encryption rules. The SQL statement would be changed into SQL before submitted. When the authorized query of data is required, the cipher text SQL would be decrypted according to Decryption rules to normal SQL before transferring to the client.

The encryption system has 2 parts: one is the management program of encryption dictionary, and another one is encryption and decryption engine. The structure is as shown in Fig. (**5**).

The encryption system stores the requirement of encryption and the basic information in the encryption dictionary, and uses the encryption and decryption engine to do the encryption, decryption and transformation. Different application systems have different encryption rules, and complete all the operation on the backstage, so the database server is transparent. Encryption and decryption engine is the core of encryption system, and placed between application and database server. It has 3 modules: encryption and decryption module, client interface module and database interface module.

Encryption dictionary is the rules for data encryption and defines all the requirement of every data sheet. It is a special database, and describes the characteristics of encrypted object. It includes field number, field names, field types, field size, field precision, scale, type of encryption, security attributes and so on. The administrator is the only one to have the right to build and manage the dictionary. Management program of Encryption dictionary is the tool of defining encryption data. Its user is just the administrators.

CONCLUSION

As to the three insecurity factors(insecurity of storing data, fragilities of database application, the risk of secret data in the database) of the database application system, DBAS security system is divided into three levels: Decision-support

level, code level, and data encryption level. The combination of security of database storage, security of applications accessing to the database and confidentiality of the data itself solves the insecurity problems of database application system. This system covers the entire process of database application system and help to achieve comprehensive and dynamic control of security of database application system.

CONFLICT OF INTEREST

The authors confirm that this article content has no conflict of interest.

ACKNOWLEDGEMENTS

Declared none.

REFERENCES

[1] X. Liu, and Z. Liu, "The analysis and discussion of computer database security management", *J. Shandong Polytech. Univ.*, vol. 27, no. 1, pp. 65-68, Feb 2013.

[2] Q. Ni, and Y. Mao, "Database security of metrology information management system", *Comput. Modernizat.*, vol. 21, pp. 182-185, Jan 2014.

[3] Y. He, "Based on SQL server database security mechanisms research and analysis", *Inform. Security Technol.*, vol. 20, pp. 48-50. Feb 2014.

[4] Y. Zhang, "Research on the database security management lssues", *Inform. Security Technol.*, vol. 18, pp. 49-47, Apr 2013.

[5] W. Liu, "Network database security technology", *Informat. Security Technol.*, vol. 22, pp. 58-60. Dec 2011.

[6] Y. Ge, and M. Zhang, "Research and application of database security", *Value Eng.*, vol. 21, pp. 246-248, Feb 2014.

[7] W. Zhao, X. Wei, and Y. Liu, "The implementation of prescription statistics preventing system based on database auditing", *J. Dongguan Instit. Technol.*, vol. 22, pp. 28-34, June 2014.

[8] H. Liu, "Research on the remote PC connects to database in security check for bottom system", *Comput. Program. Skills Mainten.*, vol. 16, pp. 101-103, June 2012.

[9] Q. Wang, "Research on the database security under the condition of cloudy computing", *Network Comput. Security*, vol. 22, pp. 65-67, Apr 2014.

[10] F. Gao, "The security and management strategy of Computer database", *Inform. Comm.*, vol. 21, pp.108-110, Jan 2013.

[11] R. Li, "Computer database security management research", *Manufact. Automat.*, vol. 34, no. 5, pp. 24-26, Oct 2012.

Received: September 16, 2014 Revised: December 23, 2014 Accepted: December 31, 2014