Leakage Prevention Method for Structured Data Based on Hierarchical Classification

Liangliang Tang, Zexin Lin, Ruizhong Chen
Information Center
Guangdong Power Grid Corporation
Guangzhou, China

*Hongxia Ma
State Key Laboratory of Information Security (SKLOIS)
Institute of Information Engineering, CAS
Beijing, China
mahongxia@iie.ac.cn

***Abstract*—As the phenomenon of users with low security classification to access to sensitive data of high-level is likely to occur during the access to sensitive information in structured data, resulting in leakage of corporate information assets, and causing serious losses to enterprises, in this paper, a new leakage prevention method is proposed for structured data in which an anti-leakage strategy with appropriate granularity is developed to achieve structured data leakage prevention based on hierarchical classification of data, effectively solving the problem of leakage of corporate information assets; meanwhile highly sensitive data, sensitive data and internal data containing sensitive information are encrypted to further protect the security of their storage and access.**

Keywords—data leakage prevention; security classification value; sensitivity level; hierarchical encryption

I. INTRODUCTION

Corporate information system stores a large amount of structured data, and its production, storage and applications are limited to relational databases (such as Oracle). Users can realize the operations (such as creation, query, add, deletion) of the structured data stored in the relational database through Structured Query Language (SQL). However, there is often a lot of sensitive information in these structured data, and if access to the sensitive information is not controlled, users with low security classification are likely to access to high-level sensitive data, resulting in leakage of corporate information assets and causing serious damage to enterprises.

This paper presents a leakage prevention method of structured data assets based on hierarchical classification in which a leakage prevention strategy with appropriate granularity is developed to achieve structured data leakage prevention based on hierarchical classification of the data so as to effectively solve the disclosure problem of corporate information assets; at the same time, highly sensitive data, sensitive data and internal data containing sensitive information are encrypted to further protect their security.

II. RELATED WORK

Currently, the use of large data has put forward higher requirements on information security. More and more enterprises store vast amounts of data into the cloud, so that data management is decentralized, the place where users process their data is hard to control, and it is difficult to distinguish between legal and illegal users, easily leading to the invasion of illegal users who steal or tamper with important data. How to prevent data leakage is currently an important issue to be addressed [1].

Anti-leakage protection for structured data is to restrict in a fine-grained way the behavior and permissions of usersô access to the database so as to prevent unauthorized access. In 2004, Radu et al protected the user access via digital watermarking method [2]. Role-based access control [3] and attribute-based access control [4] have been the main methods used for user access control in recent years. With the development of cloud computing technology, a new distributed access control method - encrypted access control [5] has become a new method for encrypted storage and access control. The method uses a cryptographic algorithm to encrypt data storage while enabling data access control via key distribution. It is possible to achieve a more fine-grained access control.

III. LEAKAGE PREVENTION STRATEGY FOR STRUCTURED DATA BASED ON HIERARCHICAL CLASSIFICATION

In this paper, the idea of encrypted access control is adopted, and data hierarchical classification is made by classifying structured data into four sensitivity levels: highly sensitive, sensitive, internal and public. Meanwhile, the database system user is also divided into four levels based on the time limit based key management scheme [6]: highly classified, classified, internal and external users. The data asset security management and control platform maintains a database table (or column) and an information table of the corresponding relation of security classification. By querying all the columns belonging to a security classification, the strategy will be issued. The basic requirement of the leakage prevention strategy is: only when the security classification of a personnel is higher than or equal to the sensitivity level of the data, can he access to the data of the sensitivity level. The specific strategy is as follows:

Strategy = <client IP> <database user ID> <user security classification> <whether the data creator> <data column sensitivity level> <time> <operation type> <response action> <severity level>

<Client IP> = {human resources department IP, authorized IP list}

<Database user ID> = {ID of all database users}

<User security classification> = {Highly classified, classified, internal, external}

This work was sponsored by the Information Center of Guangdong Power Grid Corporationôs project of Study on Data Security in Big Data Environments (No.K-GD2014-1019) and Xinjiang Uygur Autonomous Region science and technology plan (No.201230121), Strategic Priority Research Program of Chinese Academy of Sciences (No. XDA06040601).

<Whether the data creator> = {Yes, No}
<Data column sensitivity level> = {highly sensitive, sensitive, internal, public}
<Time> = {authorized period}
<Operation type> = {query, modify, add, delete}
<Response action> = {block, record}
<Severity level> = {high, medium, low, no}

The leakage prevention strategy includes:

1) Leakage prevention strategy of highly sensitive data

Strategy 1 = <client IP = human resources department IP or authorized IP list> <database user ID = any> <user security classification= highly classified> <whether the data creator = No> <data column sensitivity level = highly sensitive> <time = authorized period> <action type = modify or add or delete> <response action = block> <Severity level = high>

Strategy 1 shows that when users are highly classified, the column data is highly sensitive, but if the user is not the creator of the data, the column data will be blocked from being modified, added and deleted. The severity level is high.

Strategy 2 = <client IP = human resources department IP or authorized IP list> <database user ID = any> <user security classification < highly classified> <whether the data creator = No> <data column sensitivity level = highly sensitive> <time = authorized period> <action type = any> <response action = block> <Severity level = high>

Strategy 2 indicates that when the security classification of the user is lower than highly classified, the column data is highly sensitive, and if the user is not the creator of the data, any action on the column data will be blocked. The severity level is high.

2) Leakage prevention strategy of sensitive data

Strategy 3 = <client IP = human resources department IP or authorized IP list> <database user ID = any> <user security classification > classified> <whether the data creator = No> <data column sensitive level = sensitive> <time = authorized period> <action type = modify or add or delete> <response action = block> <severity level = medium>

Strategy 3 shows that when the security classification of the user is higher than classified, the sensitive level of the column data is sensitive, but when the user is not the creator of the data, the column data will be blocked from being modified, added and deleted. The severity level is medium.

Strategy 4 = <client IP = human resources department IP or authorized IP list> <database user ID = any> <user security classification<classified> <whether the data creator = No> <sensitive level of data column = sensitive> <time = authorized period> <action type = any> <response action = block> <severity level = medium>

Strategy 4 shows that when the security classification of the user is lower than classified, the sensitive level of the column data is sensitive, and when the user is not the creator of the data, any action on the column data will be blocked. The severity level is medium.

3) Leakage prevention strategy of internal data

Strategy 5 = <client IP = human resources department IP or authorized IP list> <database user ID = any> <user security classification = internal> <whether the data creator = No> <sensitive level of data column = internal> <time = authorized period> <action type = modify or add or delete> <response action = block> <severity level = low>

Strategy 5 shows that when the user wants to modify, add, or delete the internal data, but the user is not the creator of the data, the operation of the column data will be blocked. The severity level is low.

4) Leakage prevention strategy of public data

Strategy 6 = <client IP = human resources department IP or authorized IP list> <database user ID = any> <user security classification = all> <whether the data creator = No> <sensitivity level of data column = public> <time = authorized period> <action type = modify or add or delete> <response action = block> <severity level = low>

Strategy 6 shows that when the user modifies, adds or deletes a public data, but the user is not the creator of the data, the operation of the column data will be blocked. The severity level is low.

IV. Structured Data Encryption Method Based on Hierarchical Classification

A. Calculation of Security Classification Values

Security classification tree is a binary tree. Assume that the data is sequentially marked as 0,1,2,3 for the security classification, and map them to the leaf node of the tree.

The following example shows how security classification is mapped to a complete binary tree (CBT). As shown in Fig. 1, security classification values (s0 ~ s3) are represented by 00, 01, 10, 11 respectively in binary. For simplicity, B(s) is used as the binary representation of the security classification (s), and $V_{B(s)}$ is represented as the value of the security classification (s). In Fig. 1, the value marked with an asterisk represents the value of the midpoint node of the security classification tree. The value of each node in the complete binary tree can be calculated based on the path from the root node to the node. The value of the root node is set as $H(w)$, where w is a random integer. Therefore, the following equation is made to calculate the value of each node, where $\|$ represents concatenation.

$$V_{0^*} = H(H(w) \| 0),\ V_{1^*} = H(H(w) \| 1),$$
$$V_{00} = H(H(H(w) \| 0) \| 0)) = H(V_{0^*} \| 0), \cdots$$
$$V_{11} = H(H(H(w) \| 1) \| 1)) = H(V_{1^*} \| 1);$$

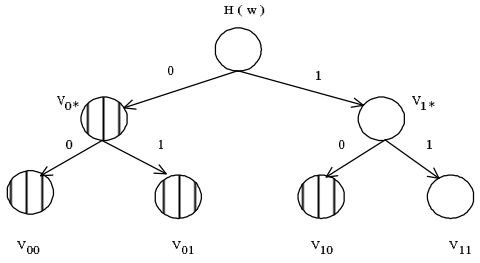

Fig. 1. Diagram of binary tree of security classification

Since the value of the leaf node of complete binary tree can be calculated from the value of child root node (for example, V_{00} , V_{01} , V_{10} and V_{11} can be calculated from the values of their root nodes, and accordingly, V_{10} and V_{11} can be calculated from the values of their root nodes, given the value of V_{0*} and V_{10} , the valued of security classification s_0-s_2 can be calculated according to the above equation. When the security classification of the user is higher than or equal to the accessible data of security classification s_2, the user may calculate V_{00} , V_{01} and V_{10} on his own only if V_{0*} and V_{10} are given.

B. Generation of Encryption Key and Key Management

The data in each column in the database table has different property and different sensitive value, so the data in each column use a unique key for encryption, which is known as an encryption key. In a database of hierarchical classification, the column data in a table can be accessed by internal, classified and highly classified personnel, assuming its sensitive value is internal (that is, the data column is internal data). That means highly classified personnel can access the data of all sensitive levels in the database. However, if the encryption key of each column of data is directly distributed to all the personnel who can access it, then a highly classified personnel has to save the encryption keys of all the data columns, and the encryption key of one data column will be distributed to more than one person, which is likely to cause the leakage of the keys.

This paper presents a key management method, that is, a data column encryption key $K_{x,s}$ consisting of the column key K_x and the security classification value $V_{B(s)}$. The specific encryption key of the column data is generated by the following formula:

$$K_{x,s} = H_K(K_x \| V_{B(s)}) \qquad (1)$$

Where $K_{x,s}$ is the decryption key of the data column, $H_k(\cdot)$ is a keyed HMAC, K is the system access root key, K_x is the column key, and $V_{B(s)}$ is the security classification value.

The root key K is generated during the initialization of the system, and the system root key K is the only constant value; the column key K_x is generated before the encryption of each column of data by the database protection server, and the column key of each column is not the same and unique used to generate the decryption key together with the security classification value $V_{B(s)}$ which is generated by the security classification tree.

When a user registers in a database system for the first time, the database protection server will send the medium value of the security classification computed to allow the user to access in the form of certificate to the user based on the user's security classification value. The user calculates each security classification value on their own on the client. For example, highly classified users can access highly sensitive data, sensitive data, internal data and public data, and therefore need to calculate the three security classification values corresponding to highly sensitive data, sensitive data and internal data (access to public data does not require security classification value). When the user requests access to sensitive data, as the security classification of the user matches the sensitivity level of the data to access while assuming that the user's identity and operation on the data to access are legitimate, the user needs to send the security classification value corresponding to the sensitive data to the database protection server which will generate the decryption key of the data column according to the column key of the accessed data column and its security classification value, and send it to the database server. If the key is correct, decrypt the data column and return it to the user; if the user wants to access to sensitive data, but he sends the security classification value corresponding to highly sensitive data or internal data, the final calculated key is wrong, and the corresponding content decoded is garbled.

Suppose a personnel with the security classification of internal steals the security classification value corresponding to sensitive data or highly sensitive data of a personnel with the security classification of highly classified, as the employee's own security classification of internal is below the sensitive level of the accessed data, he can not achieve access to sensitive data or highly sensitive data even if he has the security classification value corresponding to sensitive data or highly sensitive data.

V. Implementation of Database Leakage Prevention System Based on Hierarchical Classification

In this paper, since the highly sensitive data, sensitive data and internal data in the database are stored in ciphertext, access to the ciphertext data is achieved primarily through key distribution. That is, only a user with a key is able to access the appropriate data in order to more effectively prevent the leakage of sensitive information.

The database leakage prevention system based on hierarchical classification includes a database management server and a database protection server. Wherein the database management server is responsible for the centralized management of all software modules and is the main provider of policy management, event management, log summary and report analysis, working as the central management platform of the database information leakage prevention system. The database protection server needs to be deployed between the application server and the database server in a bypass operating mode, and is responsible for monitoring all traffic in the access to the database server through the application server. In addition, unlike the traditional plain text based database leakage prevention system, the database protection server is also responsible for generating column key, security classification value and encryption key to encrypt the data in the database. When a user registers, it will send the medium value used to calculate the security classification value in the form of certificate to the user. When a user requests access, the decryption key will be generated based on the user-generated value of security classification and the corresponding data in the database column will be decrypted and sent to the user. The overall architecture and workflow of the database information

leakage protection system based on hierarchical classification is as shown in Fig. 2 below:

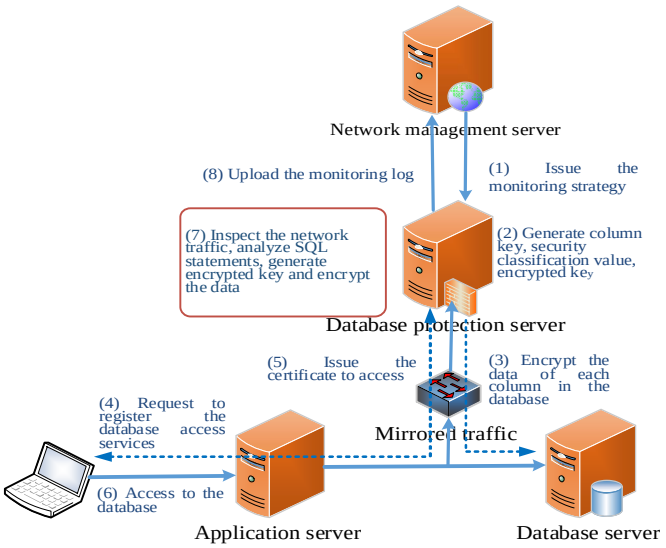

Fig. 2. Overall architecture and workflow of cipher-based database information leakage prevention system

The following steps are specifically included:

a0. The database protection server generates system root key, column key, security classification value and encryption key of highly sensitive data, sensitive data, internal data in each column, and encrypts highly sensitive data, sensitive data, internal data in each column in the database using encryption key;

a1. When a user requests a registration for database access services through the client, the database protection server will send the medium value used to calculate the security classification value of the user in the form of access certificate to the user based on the user's security classification;

a2. The user client calculates the value of security classification allowing the access based on the access certificate of the medium value;

a. The user client sends a request to access a data column in the structured data to the database server through the application server;

b. The database protection server analyzes SQL statements by using mirror to determine whether the access request contains illegal access or not - to determine whether the security classification of the user matches the sensitivity level of the data to be accessed or not, and at the same time to determine whether the user's identity and operations on the data to be accessed are legitimate or not; if not, illegal access is contained; specifically, to determine whether the security classification of a user matches the sensitivity level of the data to be accessed or not, the following steps are included: S1. The database protection server filters the userôs security classification table through the user's identity information, or obtains the user's security classification information through the userôs electronic security classification certificate; S2. The database protection server obtains the sensitivity level information about the data column based on the data column the user requests to access to; and matches the security classification of the user with the sensitivity level of the data column to be accessed to; determines whether the user's identity and operations on the data are legal or not, specifically including: determining whether the user is the creator of the data or not, while judging whether the user's access includes operations like modification, addition, or deletion; if the user is not the creator of the data, and his operations on the data include modifications, additions or deletions, the access is an illegal operation;

c. If illegal access is not included, and the data column being accessed to is public data, data requested to access is returned; if illegal access is not included and the data column being accessed is highly sensitive data, sensitive data or internal data, the user client sends the value of security classification corresponding to highly sensitive data, sensitive data or internal data, and the database protection server generates a decryption key of the data column according to the formula (1) in Section 4.2 based on the security classification value and the corresponding column key to decrypt the appropriate data column in the database server and return the data requested to access.

VI. CONCLUSION

Since the phenomenon of users with low security classification to access to sensitive data of high-level is likely to occur if access to sensitive information is not controlled when there is often a large amount of sensitive information in structured data, resulting in leakage of corporate information assets, and causing serious losses to the enterprise, in this paper, a new leakage prevention method is proposed for structured data in which an anti-leakage strategy with appropriate granularity is developed to achieve structured data leakage prevention based on hierarchical classification of data; meanwhile highly sensitive data, sensitive data and internal data are encrypted and only when the userôs security classification level is higher than or equal to the sensitivity level of the data, can he access to the appropriate data column, thus further protecting the security of highly sensitive data, sensitive data and internal data effectively.

REFERENCES

[1] X. Yan, D. Zhang. Big data research. Computer Technology and Development. Vol. 23 No. 4, Apr. 2013. 168-172.

[2] R. Sion, M. Atallah, S. Prabhakar. Rights protection for relational data. IEEE Transactions on Knowledge and Data Engineering. Vol. 16. No. 12, Decenber 2004. 1509-1525.

[3] Science Applications International Corporation (SAIC). Role-Based Access Control (RBAC) Role Engineering Process Version 3.0. 11 May 2004.

[4] A. Mohan, D. M. Blough, An Attribute-Based Authorization Policy Framework with Dynamic Conflict Resolution, Proceedings of the 9th Symposium on Identity and Trust on the Internet, 2010.

[5] A. Harrington, C. Jensen. Cryptographic access control in a distributed file system. Proceedings of the eighth ACM symposium on Access control models and technologies, 2003, 158-165.

[6] W.-G. Tzeng, A Time-Bound Cryptographic Key Assignment Scheme for Access Control in a Hierarchy, IEEE Trans. On Knowl. and Data Eng., 14(1), 182ï 188, 2002.