Survey on Database Security

Ramandeep Kaur
M.Tech CSE Dept
Jalandhar,(India)

Kiranpreet
M.Tech CSE Dept
Jalandhar,(India)

Prince Verma
Assist. Prof.(M.Tech CSE Dept
Jalandhar,(India)

ABSTRACT
A database-management system (DBMS) is a collection of correlated data and a set of programs to access correlated data. The collection of data, usually referred to as the database, contains information able to an enterprise. And hence the concept of security lies with-in. As the database contains all information relevant to particular enterprise, so it's very important to be in safe side to prevent all threats related to data. Various Security Mechanisms have been developed to prevent ill-legal and un-authorized access of data (enterprise related, web or network related etc.) by un-authorized users and hackers. Database security is a growing concern of the market these days as there has been found an un-expected growth in ill-legal access of data.

This paper is the brief analysis of all the threats and attacks against database security and integrity. Database security is the mechanisms that secure the database from data tampering, deliberate threats, unauthorized users and hackers. Security refers to protecting data from unauthorized users and Integrity refers to protecting the data against authorized users. Both concepts must be taken into consideration for making a complete check on overall security of database.

Keywords
Access Control, Attacks, Threats, Discretionary Control, Encryption and Decryption

1. INTRODUCTION
Data stored in a database is a very valuable resource for an organization and therefore should be kept secure and confidential. The term "security" refers to the protection of database against any unauthorized access that may be either intentional or accidental. Whereas Integrity refers to that any authorized access, updation, or deletion of data in the database doesn't change the validity of the data [1].

There are many aspects to security in database applications, including security at the application layer and security at the database layer. While Applications typically support a fairly complex set of access control policies, any one is having the direct access to the database can bypass the access control policies together. In addition to database administrators, anyone who discovers the database login/password used by the application has the ability to directly modify the database. Thus, even if all security measures have been taken to ensure security at the application logic level, we need to have the ability to detect any malicious actions into the database. To support this

Type of detection, database intrusion detection system is required wherein malicious transactions may be detected while system still compromising with the application level security measures. However there is a requirement of the approach based on database intrusion detection. Application based on the database systems is ubiquitous these days, often storing critical data that should not be compromised in any way. Such applications are built on multiple layers of software: at the top level is the application software, typically running on a web-enabled application server, at the next level is database system which stores the data, and below the database system are the operating system and storage system layers. Application security requires actions at each of these levels. In this work we consider the security at database layer where data are stored and protected from the malicious actions.

Applications typically have a complex security model built into the application, but when communicating to the database, an application typically connects as single database user. Anyone who gets access to the database login/password used by the application has the ability to frequently read or modify the database, bypassing all the security features built into the application. This problem is exacerbated since the database login and password are often stored in clear text in the application code or configuration files, accessible to system administrators. In addition, database administrators have full access to the data in the database. When dealing with mission critical data, preventing, or detecting and repairing, unauthorized updates to the database is absolutely critical, even more than preventing or detecting unauthorized attacks, since it may severely affects the ability of the organization to function. In this paper, we address the problem of detecting unauthorized updates/ malicious actions to the database[2].

In recent years,Web and database security technologies can ensure the confidentiality, integrity and usability of data information system, and can effectively protect the security and reliability of information system. Therefore, in order to better secure the information systems, we need to learn Web and database security-related knowledge. This chapter covers extensively practical and useful knowledge of web and database security [3].

In this paper we also proposed two mixed techniques to secure the database i.e. one is authentication followed by cryptography of database. In existing system the algorithm is less secure, less complex and much superior to implement in any system. The encryption algorithm is built on genetic algorithm, it is used to encrypt the database and validate the user's login id and password, it must verify the user and allow the user to access the database. The decryption also has a login and password based on decryption algorithm and genetic algorithm[4].

2. TYPE OF ATTACKS

There are mainly two types of attacks:-

Direct Attacks

A direct attack means raid the target directly. These are explicit attacks and are successful only if the database does not implement any protection law. If this attack fails, the attacker moves to the next.

Indirect Attacks

Indirect attacks are the attacks that are not directly executed on the target but information from or about the target can be received through other intermediate objects. Combinations of quaere used some of them having the target to cheat the security mechanisms. These attacks are difficult to track. The attacker executes the above attacks in different ways [5].

2.1 Attacks on Database can also be Classified into Passive and Active Attacks

2.1.1 Passive Attack

In passive attack, attacker only observes data present in the database. Passive attack can be done in following three ways:

1) *Static leakage:* In this type of attack, information about database plaintext values can be obtained by observing the snapshot of database at a particular time.

2) *Linkage leakage:* Here, information about plain text values can be obtained by linking the database values to

position of those values in index.

3) *Dynamic leakage:* In this, changes performed in database over a period of time can be observed and analyzed and information about plain text values can be obtained.

2.1.2 Active Attacks

In active attack, actual database values are modified. These are more problematic than passive attacks because they can mislead a user. For example a user getting wrong information in result of a query. There are some ways of performing such kind of attack which are mentioned below[5]:

1) *Spoofing* – In this type of attack, cipher text value is replaced by a generated value.

2) *Splicing* **–** Here, a cipher text value is replaced by different cipher text value.

3) *Replay* **–** replay is a kind of attack where cipher text value is replaced with old version previously updated ordeleted.

3. SECURITY AND INTEGRITY THREATS

A threat is any situation, event or personnel that will adversely affect the database security and smooth and efficient functioning of the organization. Threats to database can be intentional or accidental. Attempts have been made to protect the data through the use of firewalls and data encryption. Access control becomes more important in an environment, where data resources are shared and not all users are privileged to access and modify all data. Shareability demands some mechanism to control who does what to what data[1]. Some of the database security threats are as follows :

3.1 Basic Threats

3.1.1 Errors and Omissions

Errors and omissions are an important threat to data and system integrity. These errors are caused not only by data entry clerks processing hundreds of transactions per day, but also by all types of users who create and edit data. Many programs, especially those designed by users for personal computers, lack quality control measures. However, even the most sophisticated programs cannot detect all types of input errors or omissions. A sound awareness and training program can help an organization reduce the number and severity of errors and omissions.

Users, data entry clerks, system operators, and programmers frequently make errors that contribute directly or indirectly to security problems. In some cases, the error is the threat, such as a data entry error or a programming error that crashes a system. In other cases, the errors create vulnerabilities. Errors can occur during all phases of the systems life cycle.

3.1.2. Fraud and Theft

Computer systems can be exploited for both fraud and theft both by "automating" traditional methods of fraud and by using new methods. For example, individuals may use a computer to skim small amounts of money from a large number of financial accounts, assuming that small discrepancies may not be investigated. Financial systems are not the only ones at risk. Systems that control access to any resource are targets (e.g., time and attendance systems, inventory systems, school grading systems, and long-distance telephone systems). Computer fraud and theft can be committed by insiders or outsiders. Insiders (i.e., authorized users of a system) are responsible for the majority of fraud.

Since insiders have both access to and familiarity with the victim computer system (including what resources it controls and its flaws), authorized system users are in a better position to commit crimes. Insiders can be both general users (such as clerks) and technical staff members. An organization's former employees, with their knowledge of an organization's operations, may also pose a threat, particularly if their access is not terminated promptly.

3.1.3 Employee Sabotage

Employees are most familiar with their employer's computers and applications, including knowing what actions might cause the most damage, mischief, or sabotage. The downsizing of organizations in both the public and private sectors has created a group of individuals with organizational knowledge, who may retain potential system access (e.g., if system accounts are not deleted in a timely manner). The number of incidents of employee sabotage is believed to be much smaller than the instances of theft, but the cost of such incidents can be quite high.

3.1.4 Loss of Physical and Infrastructure Support

The loss of supporting infrastructure includes power failures (outages, spikes, and brownouts), loss of communications, water outages and leaks, sewer problems, lack of transportation services, fire, flood, civil unrest, and strikes.

3.1.5. Threats to Personal Privacy

The accumulation of vast amounts of electronic information about individuals by governments, credit bureaus, and private companies, combined with the ability of computers to monitor, process, and aggregate large amounts of information about individuals have created a threat to individual privacy. The possibility that all of this information and technology

may be able to be linked together has arisen as a specter of the modern information age[8].

4. SECURITY MEASURES

Database security refers to protection from malicious access. Absolute protection of the database from malicious abuse is not possible, but the cost to the perpetrator can be made high enough to deter most if not all attempts to access the database without proper authority. To protect the database, we must take security measures at several levels[6]:

• **Database System**: Some database-system users may be authorized to access only a limited portion of the database. Other users may be allowed to issue queries, but may be forbidden to modify the data. It is the responsibility ofthe database system to ensure that these authorization restrictions are not violated.

• **Operating System**: No matter how secure the database system is, weakness in operating-system security may serve as a means of unauthorized access to the database.

• **Network**. Since almost all database systems allow remote access through terminals or networks, software-level security within the network software is as important as physical security, both on the Internet and in private networks.

• **Physical**. Sites with computer systems must be physically secured against armed or surreptitious entry by intruders.

• **Human**. Users must be authorized carefully to reduce the chance of any user giving access to an intruder in exchange for a bribe or other favors.

5. TYPE OF SECURITY

Database security is a very broad area that addresses many issues, including the following [7]:

5.1 Legal and ethical issues regarding the right to access certain information. Some information may be deemed to be private and cannot be accessed legally by unauthorized persons. In the United States, there are numerous laws governing privacy of information.

5.2 Policy issues at the governmental, institutional, or corporate level as to what kinds of information should not be made publicly available—for example, credit ratings and personal medical records.

5.3 System-related issues such as the system levels at which various security functions should be enforced—for example, whether a security function should be handled at the physical hardware level, the operating system level, or the DBMS level.

• The need in some organizations to identify multiple security levels and to categorize the data and users based on these classifications—for example, top secret, secret, confidential, and unclassified. The security policy of the organization with respect to permitting access to various classifications of data must be enforced.

A first security problem associated with databases is that of controlling the access to a statistical database, which is used to provide statistical information or summaries of values based on various criteria. For example, a database for population statistics may provide statistics based on age groups, income levels, size of household, education levels, and other criteria. Statistical database users such as government statisticians or market research firms are allowed to access the database to retrieve statistical information about a population but not to access the detailed confidential information on specific individuals. Security for statistical databases must ensure that information on individuals cannot be accessed. It is sometimes possible to deduce certain facts concerning individuals from queries that involve only summary statistics on groups; consequently this must not be permitted either. This problem, called statistical database security.

A secondsecurity issue is data encryption, which is used to protect sensitive data—such as credit card numbers—that is being transmitted via some type of communications network. Encryption can be used to provide additional protection for sensitive portions of a database as well. The data is encoded by using some coding algorithm. An unauthorized user who accesses encoded data will have difficulty deciphering it, but authorized users are given decoding or decrypting algorithms (or keys) to decipher the data. Encrypting techniques that are very difficult to decode without a key have been developed for military applications.

6. AUTHORIZATION,PRIVILRGES AND VIEWS

Authorization is a process of permitting users (whose identity has been authenticated) to perform certain operations on certain data objects in a shared database. The person who is incharge of specifying the authorization is the Authorizer (one who owns the data). In most cases authorizer is DBA. A user may have several forms of authorization using various manipulation functions[1].

• Read authorization allows reading, but not modification, of data.

• Insert authorization allows insertion of new data, but not modification of existing data.

• Update authorization allows modification, but not deletion, of data.

• Delete authorization allows deletion of data.

We may assign the user all, none, or a combination of these types of authorization. In addition to these forms of authorization for access to data, we may grant a user authorization to modify the database schema:

• Index authorization allows the creation and deletion of indices.

• Resource authorization allows the creation of new relations.

• Alteration authorization allows the addition or deletion of attributes in a relation.

• Drop authorization allows the deletion of relations[6].

Privileges: A user who has been granted some form of authorization may be allowed to pass on this authorization to other users. However, we must be careful how authorization may be passed among users, to ensure that such authorization can be revoked at some future time.

Views: A view is a dynamic result of one or more relational operations with the base tablewhich produce another table. It is a virtual table[1]. A view can hide data that a user does not need to see. The ability of views to hide data serves both to simplify usage of the system and to enhance security. Views

simplify system usage because they restrict the user's attention to the data of interest. Although a user may be denied direct access to a relation, that user may be allowed to access part of that relation through a view. Thus, a combination of relational-level security and view-level security limits a user's access to precisely the data that the user needs.

In our banking example, consider a clerk who needs to know the names of all customers who have a loan at each branch. This clerk is not authorized to see information regarding specific loans that the customer may have. Thus, the clerk must be denied direct access to the loan relation. But, if she is to have access to the information needed, the clerk must be granted access to the view cust-loan, which consists of only the names of customers and the branches at which they have a loan. This view can be defined in SQL as follows:

Create view cust-loan as

(Selectbranch-name, customer-name

Fromborrower, loan

Whereborrower.loan-number = loan.loan-number)

Suppose that the clerk issues the following SQL query:

Select*from cust-loan

Clearly, the clerk is authorized to see the result of this query. However, when the query processor translates it into a query on the actual relations in the database, it produces a query on borrower and loan. Thus, the system must check authorization on the clerk's query before it begins query processing[6].

7. DATABASE SECURITY AND THE DBA

The database administrator (DBA) is the central authority for managing a database system. The DBA's responsibilities include granting privileges to users who need to use the system and classifying users and data in accordance with the policy of the organization. The DBA has aDBA account in the DBMS, sometimes called a system or super user account, which provides powerful capabilities that are not made available to regular database accounts and users (Note 1). DBA privileged commands include commands for granting and revoking privileges to individual accounts, users, or user groups and for performing the following types of actions:

7.1. Account creation: This action creates a new account and password for a user or a group of users to enable them to access the DBMS.

7.2. Privilege granting: This action permits the DBA to grant certain privileges to certain accounts.

7.3. Privilege revocation: This action permits the DBA to revoke (cancel) certain privileges that were previously given to certain accounts.

7.4. Security level assignment: This action consists of assigning user accounts to the appropriate security classification level.

7.5. Introduction to Statistical Database Security: Statistical databases are used mainly to produce statistics on various populations. The database may contain confidential data on individuals, which should be protected from user access. However, users are permitted to retrieve statistical information on the populations, such as averages, sums, counts, maximums, minimums, and standard deviations[7].

8. ENCRYPTION AND DECRYPTION

The method used for inserting, updating and viewing the data in the database is explained below:

8.1 Inserting Values into the Database: The method used for insertion is explained below:

8.1.1 Authentication: This is a basic security measure that is available in most of the systems. This Authentication feature has a valid user name and password. The feature has been designed in such a way that a particular userid cannot be used to login after three continuous unsuccessful attempts, i.e., that particular userid would be blocked and for further access the administrator has to reset the password.

8.1.2. Input Record: The record for the database is accepted. The type of encoding has to be specified for insertion the values of the fields in database are given. For Updating, initially the primary key value and the encoding type are given. If it is found to be valid then the other field values are obtained. The encoding type of a row cannot be changed while updating. In updating if the encoding type or the primary key value of the database is specified wrongly then the system automatically comes out of application.

8.1.3. Encryption Algorithm: All the data given in the above step are given to the encryption algorithm for encryption. These are the data that are to be encrypted before adding to the database. The encryption process has the following processes.

8.1.4. Encoding of Plain Text: The type of encoding depends on the problem. In this paper we have chosen octal encoding and binary encoding. Here each character has a specific octal/binary representation. Considering the plain text as a stream of characters, the characters of the plain text are converted to an octal/binary format which is

Used in the further process of encryption.

8.2 Updating and Viewing the Records in the Database

8.2.1. Encoding: The receiver obtains the cipher text which is in alphanumeric format. But since the encryption was

Done in an octal/binary format, the cipher text is again encoded to octal/binary format. The sender and receiver use the same type of octal encoding schemes so as to maintain integrity and this is decided before the encryption and decryption of data starts.

8.2.2. Interchange Characters: After converting the cipher text to octal format, the receiver proceeds to get back the octal encoded plaintext and the original best keys are obtained. The process is done by taking the alternate characters with first one for data and the next one for key. This process is repeated for all the data that he receives. Since our paper used multi point crossover during encryption, which both the users share before starting the communication.

8.3.3. Octal Decoding: The decryption is not complete after reverse crossover of the cipher text. The reverse crossover gives only the octal encoded plain text. But the plain text is not octal encoded according to sender. So the octal decoding is to be done with the octal encoded plain text.

Thus after this step the receiver gets the complete plain text[4].

9. CONCLUSION AND FUTURE SCOPE

The security in the DBMS is one of the main concerns of the researchers now-a-days and there is an interest to develop the possible database intrusion detection systems. We discuss the three approaches for database IDS and basic design of such architectures. We again emphasize that to the best of our knowledge, this is the one literature presenting the design of database IDS architectures. With the rapid development of network database, it facing major difficulties like accessed by unauthorized users, hackers and others. So overcome this problem we use cryptography and genetic algorithm. The encryption of the database into alpha-numeric form is achieved by performing octal encoding, crossovers and octal decoding and then sent to the receiver where the receiver decrypts the cipher text by performing octal encoding, reverse crossover and octal decoding the octal sequence. Databases are a favorite target for attackers because of their data. There are many ways in which a database can be compromised. There are various types of attacks and threats from which a database should be protected. Solutions to most of the threats mentioned above have been found, although some solutions are good while some are only temporary.

10. REFERENCES

[1] Anshuman Sharma, Anurag Gupta, Jagmohan Mago, "Fundamentals of DBMS", Fourth Ed.,Lakhanpal Publishers, 2011

[2] Elisa Bertino, Ravi Sandhu, "Database Security-Concepts, Approaches, and Challenges," IEEE Transactions on Dependable and Secure Computing, vol. 2, no. 1, pp. 2-19, Jan.-March 2005, doi:10.1109/TDSC.2005.9

[3] Jiping Xiong, Lifeng Xuan, Jian Zhao and Tao Huang, "Web and Database Security", Unpublished

[4] Sabareesan M, Gobinathan N, "Network Database Security Issues and Defense", International Journal of Engineering Research and Applications(IJERA), Vol. 3, Issue 1, January -February 2013, pp.1748-1752, ISSN: 2248-9622

[5] Shelly Rohilla, Pradeep Kumar Mittal, "Database Security: Threats and Challenges ", International Journal of Advanced Research in Computer Science and Software Engineering, Volume 3, Issue 5, May 2013 ISSN: 2277 128X

[6] Silberschatz, Korth & Sudarshan, "Database System Concepts", Third Ed., McGraw Hill International Editions, Computer Science Series-1997

[7] Ramez Elmasri and Shamkant B. Navathe, "Fundamentals of Database Systems", Third Ed., Pearson Education

[8] An Introduction to Computer Security: The NIST Handbook", National Institute of Standards and Technology, U.S. Department of Commerce, Special Publication 800-12