ABSTRACT

The development of computer technology, computer networks play an important role in human society, to the social development. It used a variety of database technology, which is giving hackers to provide a means of attack damage. To this, aiming at the computer network database security threats and hackers prevention of related problems are analyzed.

Keywords - Database, SQL, Database explosion, exception handling, security.

I. INTRODUCTION

This huge computer network platform makes the community more and more information. Database technology has played a leading position. Meanwhile, network intrusion aggravating, each country and big companies systems have hacker's patronage, and these large systems are supported by the database. Therefore, the database security has become the object of concern to everyone. And single Intrusion Detection System (IDS) can be installed on a segment or gateway of computer networks to promptly analyze each packet and to isolate normal activities from abnormal activities. IDS users may also install multiple Intrusion Detection Systems (IDSs) at different segments of the same network to safeguard their computer resources. Attacks that target the application level may be different from attacks that target operating system or Database Management Systems (DBMS) of targeted networks. Hence, there are different categories of IDSs. Researchers have shown that different IDSs have different formats for logging network intrusions. The patterns of packets that aim to illegally steal or hijack sensitive information from computer networks are different from flooding attacks that aim to disrupt availability of computer services. Thus, this paper critically describes and analyses emerging security challenges Computer network database. The review will be functionally useful to researchers, vendors, security professionals and IT end users in general.

II. DATABASE ATTACK MEANS

For the invasion of computer network database, the following database explosion and SQL injection on analysis.

SQL injection attacks

Some form of hacker attacks, of which 60% may be subject to SQL injection attacks. It security and control firms and Internet Crime Complaint Center have issued a report that this year the number of SQL injection attacks is rising, especially related to financial services and online retail website. SQL injection attack will be the top of six network security threats. Therefore, defense against SQL injection attacks is very important.

SQL injection is a way to insert or add the SQL code to the application (user) input parameters in the attack, then after passing these parameters to the backend SQL Server to parse and execute. From the normal port access, the surface is no different from ordinary access (firewalls are generally not aware of it), the use of procedural legitimacy of the user input data are not strictly testing or the characteristics of, from the client to submit SQL code to collect information in the database, and then obtain the site administrator's account and password.

SQL injection attack steps:

- The use of special SQL statements looking for injection vulnerability;
- Using injection vulnerability repeated attempts to obtain background information on the database;
- Analysis of the database information, laying the groundwork for further attacks. Common SQL injection attacks are generated by constructing a dynamic string, the following describes several situations:
  - Mishandling of the escape character

Escape character in the database which has a special meaning. For example: single quotes ('), space ( ), double vertical bar (||), comma (,), dots (.) and double quotes ("), etc. Now take the single quotes as an example. SQL database resolve single quotes between code and data separator. Therefore, only the URL or Web page (or application) of the field, enter
a single quotes, you can identify the Web site is subject to SQL injection attacks. Here is a user input is passed directly to the dynamically created SQL statement:

```php
$result = mysql_query ($SQL);
$rowcount = mysql_num_rows ($result);
$row=1;
While ($db_field=mysql_fetch_assoc ($result))
{
    If ($row<=$rowcount)
    {
        Print $db_field [$row] . "<BR>";
        $row++;
    }
}
```

**Figure 1.** Mishandling of the escape character

- **Type of improper handling**
  If a user supplied field no verifies the validity or implementation of compulsory type, there will be an SQL injection. For example:
  
  ```sql
  SELECT * FROM data WHERE id = " + variable + ";
  ```
  
  Here, “variable” is a hope that with the “id” field in the figures. If, when used a number field in a SQL statement, if the programmer did not check the legitimacy of user input, such attacks occur.

- **Query set not handled properly**
  Sometimes necessary to use dynamic SQL statements on the application of some complex code, because the program development stage may not know the table or query field. Here is a simple example; it will pass user input directly to the dynamically created SQL statement.

```php
$SQL="SELECT $_GET ["column1"], $_GET ["column2"],
$_GET ["column3"], FROM $_GET ["table"];  
$result=mysql_query ($SQL);
$rowcount=mysql_num_rows ($result);
$row=1;
While ($db_field=mysql_fetch_assoc ($result))
{
    If ($row<=$rowcount)
    {
        Print $db_field [$row] . "<BR>";
        $row++;
    }
}
```

**Figure 2.** Query set not handled properly

- **Improper Error Handling**
  Improper Error Handling will give the system a security problem. The most common problem is the detailed internal error message is displayed to the attacker. These details will be provided for the attack potential pitfalls associated with the system, an important clue. For example: An attacker could use this information to extract how to modify or construct injection to avoid the developers query, and get how to manipulate the database.

Here is an example. Select a user identifier from the drop-down list, the script generates a dynamic SQL statement:

```csharp
Private void SelectedIndexChanged (object sender, System.EventArgs e)
{
    String SQL;
    SQL="SELECT * FROM table ";
    SQL+="WHEREID=" + UserList.SelectedItem.                 Value+"";
    OleDbConnection con =new OleDbConnection(connectionString);
    OleDbCommand cmd =new OleDbCommand (SQL, con);
    try
```


```csharp
{ 
Con.Open ();
Reader=cmd.ExecuteReader ();
Reader.Read();
lblResults.Text="<b>"+reader ["LastName"];
lblResults.Text+=","+reader ["FirstName"]+"</b><br>
lblResults.Text+="ID:"+reader ["ID"] +"<br>
reader.Close();
}
catch (Exception err) 
{
  lblResults.Text="Error getting data. ";
  lblResults.Text+=err.Message;
}
finally 
{
  Con.Close ();
}
```

Figure 3. Improper error handling

**Mishandling of multiple submission**

White list [2] is a character in addition to the list of outside, the other characters against the technology used. There are any types of attacks which can be presented in various ways, to effectively maintain such a list is a very arduous task. Use the list of unacceptable characters is to define the potential risk is likely to ignore a list of unacceptable characters. Developers often used to think that users will follow the logical order of their designed flow operation. For example: When a user has reached a second form in the form, they will expect the user must have completed the first form. But in fact, out of order with a direct URL to request resources can be very easy to avoid the expected data flow.

**A. Database explosion attacks**

We are looking into the purpose of vulnerability, which is the desired database, such as: user name, password, etc., can also take this further to obtain permission. If we do not have to get into the database, not better? Thus, database explosion has become a much simpler means of invasion. Database explosion is also a favorite and effective effort invasion tactics for the invaders. Database explosion is through some technical means or bugs get the address of the database, and illegal downloading of data to the local.

Hackers are very happy in this work. Because when hackers get site database, site management accounts can be obtained on site destruction and management, but also through the database by website user private information, even by the server's highest authority. For database explosion, there are many ways to be downloaded on the network database. The main: %5c database explosion, conn.asp database explosion.

- **%5c database explosion**

  % 5c database explosion is open a Web page, the URL address in the "/" into "/% 5c", and then submitted to the server, you can get the path of the database.

  In fact, not all URLs are valid only with the behavior of alls to the database, the web site. For example: "/asp?id=". Just make sure that the page has call database behavior, to explosive library can be implemented% 5c. % 5c database explosion, is not the page itself vulnerability, but to use the IIS decoding a feature. If the IIS security settings are not considered comprehensive, and developers did not consider the case of IIS the error, it will be exploited by attackers. Why do we use% 5c? It is the "/" hex code, but the author "/" and "/% 5c" will produce different results. % 5c database explosion is the URL "/" into "/% 5c", and then need to consider a problem. What is the location in which"/" into "% 5c" it? This site is a few levels need to see the directory. General website directory sites are second or third more, there is% 5c libraries are more likely to burst. For example: http://hxhack.com/soft/view.asp?id=58. Here it one of the second "/" into "/% 5c". So, which is http://hxhack.com/soft% 5cview.asp? id = 58.

- **conn.asp database explosion**

  conn.asp database explosion is the first explosion occurs. Database calls conn.asp here represents documents, mostly the name. Even though some sites will be renamed, we also discussed as conn.asp.
After "% 5c" database explosion appears, conn.asp database explosion begins rarely mentioned. I think that, % 5c database explosion server security database with the strengthening of the role will be less and less. Instead the role of conn.asp database explosion greater can be man-made structure. If% 5c database explosion is to use an absolute blast library path error, then conn.asp database explosion is to use a relative path to database error. Generally speaking, as long as the system conn.asp not at the root, and call the file in the root directory, there will be problems. Accurate to say that, conn.asp and call it the relative position of the file if changed, would be an error, burst out of the database path.

B. Unsafe configuration database

These are just for the security code, but can not ignore the security of the database itself. We can use many methods that can be stolen or manipulated to reduce the data access level, and SQL injection attacks caused damage. Now the mainstream databases Oracle, MySQL, and SQL Server and so on. They are the default user with a lot of pre-installed content. For example:

SQL Server database system administrator account "sa"; MySQL's "root" user account; Oracle's "SYS", "SYSTEM", "DBSNMP" account and so on. These accounts by default pre-set passwords are often known. Different types of database servers also added its own access control model; a variety of permissions for the user account is assigned to different operations. But also default support exceeds the demand and the ability to modify the function of the attacked (xp_cmdshell, LOAD_FILE, OPENROWSET, ActiveX, etc.). Developers write programs, the general permission to use a built-in account to connect to the database, rather than the program needs to create a specific account. So that these powerful built-in account on the database needs to perform many independent operations and procedures. When the attacker to use SQL injection vulnerability and the authorized account connect to the database is to be used in the database privileges of the account code.

III. PREVENT SQL INJECTION ATTACKS AND DATABASE EXPLOSION ATTACKS

At present, computer network database for SQL injection attacks and database explosion attacks, there are security precautions following methods:

A. User input validity check

Review the use of the user, mainly through the URL submission check client existence of the parameters "and, or .", "\", "\", exec, select, from, insert, delete, update, count, user, xp_cmdshell , Add, net, drop, table, truncate, mid '', for SQL injection and other common characters, or strings, but also limit the length of user input. Examples of special characters check the function as follows:

```csharp
public bool inputCheck (string str)
{
    string str1="select|insert|delete
                        from | count | drop table | update
                        Truncate | asc | mid | char
                        xp_cmdshell | exec | master
                        netlocalgroup administrators
                        \| net user | | | or | and |
                        String str2=" [-.:|\()[^\@!\^\$\&]\] |\{[^%@\^!\$\&]\} "
                        bool result= !Regex. IsMatch(str,
                        str1, Regex-Options. IgnoreCase) | |
                        Regex. IsMatch (str, str2));
    return result;
}
```

Figure 4. User input validity check

A. Application of storing process

Application designed specifically to use stored procedures to access the database is a SQL injection can prevent or mitigate the impact of the design. A stored procedure is stored in the database program. Depending on the database, you can use many different languages and variations to write stored procedures. Storage process is very helpful to reduce the potential serious impact of SQL injection vulnerabilities, because in most of the database using stored
procedure can be configured at the database level access control. This is important, which means if we find the available SQL injection problems, you can configure through the proper licensing to ensure that the attacker can not access sensitive information in the database.

B. Encapsulate the client to submit information
This approach needs the support of RDBMS, Oracle currently only using the technology.

C. Database file extensions to ASP, ASA
Traditional preventive measures, most users prefer the suffix of the database into the ASP MDB before or ASA. Although in this way to prevent database explosion, with the continuous development of computer technology, similar to those of traditional methods can not meet the requirements of the latest prevention. Modified for ASP or suffix after the ASA database files, hackers can determine the storage location by looking, you can quickly download tools such as combination of Thunder download available.

D. Add "#" in front of the database name
Currently, many database administrators to add the # sign in front to avoid the database is downloaded; this is because IE can not download the file with a # sign. But the Web can be used in addition to regular access; it can also be combined with coding techniques to access IE. Each of the different characters in IE in all relevant codes, encoding binary%23 to replace the # sign, followed by treatment in this manner, even if the suffix with the # same database file can be download. Such as: # data.mdb file downloads, simply enter%23data.mdb browser is to download the database file through IE, which is for the # symbol can not play the role of defensive measures.

E. To the database user password encryption
User passwords and other sensitive information encryption, such as the use of MD5 encryption, that is, ciphertext = MD5(plain). MD5 can not be decrypted by the attacker even if they know there is encrypted in the database the same password as the garbled; he has no way of knowing the original password. It should be stressed that once the user is lost or forgotten password is difficult to retrieve.

F. Hidden back entrance
Taking into account the background database management often will take a different way of operation and maintenance, will not add fortified SQL injection code, nor will it set the appropriate admin page the link points. In order to avoid background landing page is scanned, programmers need to be adjusted back-catalog management, which best can not be set to: manage, admin and so easy to break words, usually set into a non-word letters, numbers, combinations of various Form more complex combinations as possible. The name of the login page, try not to set to: login.asp and other names, so as not to be cracked. H. Establish reasonable error return policy
SQL error message as often disclose certain details of database design, so the application is running SQL error occurs, do not put the database returns the error message displayed to the user without discrimination. Reasonable approach is to package SQL error pages, according to whether the user has debug privileges, decided to show all error messages to the user or only prompt SQL runtime error.

IV. INDUSTRIAL STANDARDS
The efficacy and validity of industrial best practices for safeguarding standalone computers, smart devices, information resources, computer networks and their peripherals are not frequently evaluated in the domain of computer network security and network forensics. The main reason is that industrial best practices are mostly propounded by globally recognized body of experts such as American National Standards Institute (ANSI), National Institute of Standards and Technology (NIST), British Standards Institute (BSI), Standard Organization of Nigeria (SON), Institute of Electrical and Electronics Engineers (IEEE), International Standards Organization (ISO) and Information Systems Audit and Control association (ISACA) [14, 19]. While industrial standards are constantly being updated and new disposable devices are being manufactured from time to time, we are unsure whether the new versions of each standard is sufficient enough to adequately provide the necessary guidelines that network forensics experts and IS auditors would use to effectively discharge their duties.

A. Outsourcing of IT operations
Outsourcing of IT functions is becoming the best IT practices in the industry. The rate of change in customer requirements, natural disasters and computer crimes across the globe are also on the increase [12-14]. IT auditors must continually evaluate third party applications, selection of vendors; profiles of access assigned to vendors and Business Contingency Planning (BCP) procedures of their organizations to ensure strict compliance with best practices and to
lessen downtime. There is a growing rate of complexity on how to actually conduct purposeful Business Impact Analysis (BIA) and penetration testing across multiple outsourced IT operations in order to anticipate potential vulnerabilities of computers and cloud resources that can be exploited by malicious users and hackers. There are numerous unidentified stealthy attacks and techniques to evade detection that may not be covered by the present industrial best practices known to the BIA team. Consequently, current best industrial practices and the results of most BIA are highly subjective.

B. Redundant intrusions and redundant alerts

IDSs fundamentally generate numerous alerts while in operation to detect potential attacks. From that fact, conceptual discussions of redundant intrusions and redundant alerts can generate controversies in some cases. Asically, redundant intrusions such as in Figure 4.1 below are similar intrusions that are reoccurring over time. Statistically, some alerts are related, some are partially related and others may not correlate altogether.

In other words, it is difficult to determine how closely two alerts triggered by IDS co-vary. It is also difficult to separate alerts that form perfect negative correlation, no correlation or those that form perfect positive correlation. Human factor in analyzing intrusion logs is relevant in the context of administration of IDS especially whenever attackers launch attacks that cause NIDS to trigger alerts with the aim of exhausting capabilities of the analysts. The issue here is that some analysts might not notify their management or their strategic managers on the need to deploy additional hands to cooperatively analyze swamped alerts for the fear of being fired. Accordingly, many attacks go unnoticed despite series of warnings from the IDS in use. Several factors can cause alerts swamping and intrusion redundancy.

Figures 4.2 to 4.4 illustrate that configurations of a standalone computer system can completely different from configurations of computers in a set of computer networks. The basic fact is that an attack on a standalone computer system has the same source and destination address unlike an attack that involves computers networks such as in Figure 3. Essentially, Figure 3 shows one IDS (Q) that mediates between two computer networks A and B and another IDS (M) that is installed on a computer in network B. Figure 4 illustrates the complexity in the installation of IDSs shown in Figure 3. Figure 4 therefore shows an attack that originates from a computer machine or a device (S) towards another computer system or a device (T). There are IDS in the target system (T) and another IDS (NIDS) that mediates between the source of the attack and its destination. Users often deploy multiple IDSs to maximally detect network intrusions. Homogeneous, heterogeneous or semi-heterogeneous IDSs can be deployed in this regards. Homogeneous IDSs are similar IDSs such as in the deployment of Snort in multiple segments in an organisation.
Heterogeneous IDSs comprise of different categories of IDSs. It may signify the deployment of NIDS (such as Snort for monitoring computer networks) and host-based IDS (such as OSSEC for monitoring the integrity of internal components of computing systems in an organisation).

However, in an attempt to rigorously safeguard computer systems and cloud resources from intruders, design flaws are indirectly built into them.

V. CONCLUSION
Security of the database for any system, are crucial. In response to these attacks on the database, we will have to be considered during the development phase, in the code and database configuration on the good aspects of a comprehensive prevention. This is not only the beginning for the system to improve safety and reduce the risk of much safety, security maintenance for the system post-human, material and financial resources to save significant costs.

REFERENCES