Database Security –Threats & Prevention

Simanta Shekhar Sarmah

Business Intelligence Architect, Alpha Clinical Systems Inc., NJ, USA

Abstract

Information security is vital. It conditions the economic activity of companies and the trust in public bodies voluntary or accidental disclosure of financial or private data can have unfortunate consequences economic, commercial and ... legal, around 25% of incidents are internal staff, 50% are due to loss or theft of miscellaneous equipment. What is the security of IT and methodology of security management of IT? What is the concept of penetration database? How to Secure databases and ensuring compliance? what are the strategic and technical security measures for good database security?

Keywords — Cybersecurity, Database protection, Threats to a database, Information assurance, Security measures, Information security

I. INTRODUCTION

Security in IT is the set of means implemented to reduce the vulnerability of computer systems against accidental or intentionalthreats that it may be confronted. In other words, it is the set of techniques that ensure that the resources of the information system (hardware or software) of an organization are used only in the context where they are planned. The basic requirements of IT security boils down to ensuring:

- Availability: System information must always be available to authorized persons.
- Confidentiality: Information about the system should only be distributed to authorized persons.
- Integrity: Information about the system should only be changed by authorizedpersons.

In general, it could be argued that the methodology of security management of IT is defined as follows:

A. Perform a Risk Analysis [1]

Risk Analysis is only possible way to protect oneself against risks that we do not know! That said, it is appropriate for each company to assess the risks, that is to say, measure them according tothe probability of their appearances and their possible effects. It is in companies' best interest to evaluate, albeit roughly, these risks and the means to be implemented, depending on their costs. The notion of risk can be understood as being the product of a harm by the probability of its occurrence. The notion of risk is defined by specialists according to the following equation:

Risk = Injury x Probability of occurrence

This formula implies that an event whose probability is quite high but whose injury can be prevented is an acceptable risk. The same goes for an event of unstoppable gravity (eg collapse of a building), but with a low probability of occurrence. It goes without saying that in the first case, the risk becomes acceptable only if the preventive measures against the harm are effective and efficient.

B. Establish a Security Policy

Once the risk analysis has been carried out, the security policy is put in place. This has the role of:

- Define the framework for using the resources of the information system
- Identify the security techniques to be implemented in the different departments of the organization while complying with the ISO 2000X standard [2]
- Educate users about IT security

C. Implement Security Techniques

These techniques are the answer to the basic requirements of IT security defined above. Their role is to ensure the availability, integrity, confidentiality and in some cases the sustainability of information in information systems. Security techniques include:

- Vulnerability audit and penetration tests (Pen-Test)
- Securing data: encryption, authentication, control access
- Network security: Firewall, IPS/IDS, etc
- Security information monitoring
- User education
- The business recovery plan.

II. THE CONCEPT OF PENETRATION DATABASE

The SGBD organizes data and gives users the means to retrieve information, this information is based on data statistical functions for example:

- If access is uncontrolled, sensitive data would not be entered.
- Confidentiality issue: Prevention of unautorized disclosure of data/information.
- Integrity issues: Prevention of unautorized modification of data.

- Broadly speaking, computer security also includes.
- Availability: Prevention of denied access to data.

A. Database Security Theats

In general, the major risk associated with any attack depend on three factors : threats, vulnerabilities and impacts.

This part represent Top 10 security threats in databases and howa it works:

1. SQL injection.
2. Excessive privilege abuse.
3. Abuse of legitimate privilege.
4. Privilege escalation.
5. Exploitation of Vulnerabilities in Vulnerable or Incorrectly Configured Databases.
6. Weakness of the native audit.
7. Denial of service.
8. Vulnerabilities of database communication protocols.
9. Unauthorized copying of sensitive data.
10. Exposure of backup data.

The infrastructure of a company's database is subject to a large number of threats, among the most critical threats ranked number 1 in the digital world is SQL injection [4]. By addressing this threat, organizations will meet global compliance requirements such as OWASP [5] and industry best practices for data protection and risk mitigation.

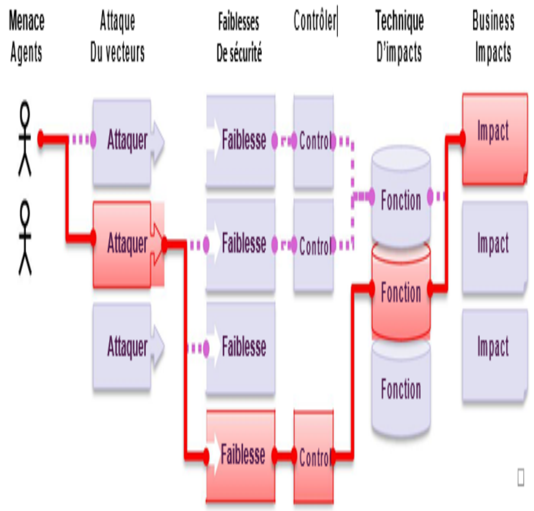

Figure 2: The risk of web and more specifically the databases [6]

Explaination of some security threats :

1) SQL Injection [18]:

In an SQL injection attack, the author typically inserts (or "injects") unauthorized database information into a vulnerable SQL data string. Typically, the affected data strings include stored procedures and input settings for web applications. This injected information is sent to the database where it is executed. Using SQL injection, attackers can gain unlimited access to an entire database.

2) Excessive Privilege Abuse [7]:

When users (or applications) have access privileges to a database that exceeds the requirements of their professional function, they may abuse these privileges for malicious purposes. For example, a university dean/principal whose function requires only the ability to change student contact information may benefit from excessive database update privileges to modify grades.

Database users may end up with excessive privileges for the simple reason that most of the time database administrators do not have the time to set or update control mechanisms/access for an individual user. As a result, all users or large user groups have default generic access privileges that far exceed the requirements of their specific function.

3) Abuse of legitimate Privilege [8]:

Users may also abuse legitimate access privileges to a database for unauthorized purposes. Imagine a potential malicious health official with privileges to view patients' medical records through a custom web application. The structure of the web application normally limits the privileges of users to viewing the medical record of a single patient. Multiple folders cannot be viewed simultaneously, and electronic copies are not allowed. However, the attacker can work around these limitations by connecting to the database using some other means such as MS-Excel. With the use of MS-Excel and its legitimate login credentials, the employee can retrieve and update the patient's medical records. It is unlikely that such personal copies of the patient record databases comply with the rules on the protection of patient data as defined by medical institutions. There are two risks to consider. The first is the malicious official who tries to resell patients' medical records. The second (and perhaps most common) is the careless employee who retrieves and backs up a large amount of data on his client computer for legitimate business purposes. Once this data is backed up on another computer, it becomes vulnerable to Trojan horses, laptop theft, and so on.

4) Privilege Escalation [9]:

Attackers can take advantage of database platform software vulnerabilities to turn the access privileges of an ordinary user into those of an administrator. All vulnerabilities can be found in most cases in stored procedures, built-in functions, protocol implementations, or even in SQL data. For example, a software developer working in a financial institution can take advantage of a vulnerable function to claim database access administrator privileges. With administrator privileges, the malicious developer can disable auditing mechanisms, create ghost accounts, transfer funds, and so on.

5) *Exploitation of Vulnerabilities in Vulnerable or Incorrectly Configured Databases [10]*:

Databases are often vulnerable, uncorrected, or have accounts and configurations always set by default.While vendors are developing patch packs to fix systems for a specific vulnerability, enterprise databases remain freely exploitable. When a hotfix is released, it is not available immediately. There are different aspects to consider when applying a hotfix to a database. First, the organization must first evaluate the system fix procedure with the fix in question, trying to understand how the fix would affect the system. Sometimes a fix may conflict with existing code, or it may involve other operations. Then, the system experiences a downtime when the database server fails to provide users with a service to fix it. Finally, large companies with dozens or even hundreds of databases must provide a correction plan, prioritizing the databases, which must be corrected first. Therefore, it is not surprising to see that for many companies, the correction process lasts several months, usually between 6 to 9 months (duration established on the basis of research conducted by the independent group of Oracle users or IOUG *). Access to databases, system administrators and IT administrators, developers, all play a role in the remediation process. While resources and time are limited, servers remain vulnerable for months after launching a patch.

Account and configuration settings that are always set by default on a production database can be exploited by an attacker. An attacker can attempt to access the database using a default account. A weak audit setting may allow the attacker to bypass audit trails or remove any traces of his or her activities. Low identification patterns allow attackers to identify themselves as legitimate users of databases by stealing or obtaining login credentials.

6) Weakness of the Native Audit [11]:

Automatic registration of all sensitive and / or unusual database transactions should be the underlying basis of any database deployment. A weak database auditing rule represents a serious organizational risk at many levels.

Regulatory risk - Organizations using weak (or sometimes non-existent) database auditing mechanisms will increasingly realize that they do not comply with government regulations. The Sarbanes-Oxley (SOX) regulation in the area of financial services and the Healthcare Information Portability and Accountability Act (HIPAA) in the health field are just two examples of government regulationswith clear requirements for auditing databases.

»**Deterrence -** Like video cameras that record the faces of people entering a bank, database auditing mechanisms serve to deter attackers who know that monitoring database audits provides to the investigators forensic information on the perpetrators of a crime.

»**Detection and Recovery -** The audit is always the last step to defend the databases. If the attacker manages to circumvent other defense systems, the results of the audits can identify the existence of a violation after the attack. Audit results can then be used to link a violation to a particular user and / or repair the system.

Database software platforms typically incorporate basic auditing features but have multiple weaknesses that limit or prevent their deployment.

»**Lack of User Accountability -** When users access a database from web applications such as Oracle, SAP, or PeopleSoft, typically a native auditmechanisms do not know the identities of specific users . In this case, all the activities of a user are associated with the account name of the Web application. Therefore, when the results of the native audits reveal the existence of fraudulent database transactions, no link can be made with the responsible user.

» **Performance Degradation -** Native database auditing mechanisms are known to consume CPU and hard disk resources. The performance degradation seen when auditing features are enabled forces many organizations to reduce the number of audits or simply remove them.

»**Feature Separation -** Users with administrative access rights (obtained either legitimately or maliciouslyto the database server can easily disable the audit feature to hide fraudulent activity. Ideally, the audit functions should be separate from those of the database administrators and those of the database server platform.

»**Limited Granularity -** Most native auditing mechanisms do not record the necessary information to support the detection of an attack even forensic analysis and recovery.For example, the database client application, source IP addresses, query response items, and failed queries (an important attack recognition flag) are not registered by many native mechanisms.

»**Owner -**The auditing mechanisms are specific to the database server platform. Oracle results are different from MS-SQL results, MS-SQL results are in turn different from Sybase results, and so on. For organizations that combine database environments, it literally eliminates the implementation of uniform and scalable auditing procedures in the enterprise.

7) *Denial of Service [12]:*

Denial of Service (DOS) is a general attack category that denies access to network applications to certain users. Denial of service conditions can be created by many techniques, many of which are related to the aforementioned vulnerabilities. For example, denial of service can be achieved by taking advantage of the vulnerability of a database platform to drop a server. Other common denial of service techniques include data corruption, network congestion, and server resource overhead (memory, CPU, etc.). Overloading resources is a very common technique in database environments. The motivations behind denial of service attacks are also diverse. Denial of service attacks are mostly linked to extortion attempts by which a hacker remotely installs servers until the victim places his funds in an international bank account.Denial of service can also be linked to an infection with a computer worm. Whatever the source, denial of service is a serious threat to many organizations.

8) Vulnerabilities of Database Communication Protocols [13]:

An increasing number of security vulnerabilities are identified in database communication protocols designed by all database providers. Fraudulent activities targeting these vulnerabilities can range from unauthorized data access, data corruption, and denial of service. The SQL slammer2 computer worm, for example, took advantage of a flaw on the Microsoft SQL server protocol to force a denial of service. In order to make the situation more complicated, there is no record of these native audit journalism fraud vectors, because not all protocol operations are covered by the majority of database audit mechanismsnative.Prevention of database communication protocol attacks. Database communication protocol attacks can be overcome by a technology, commonly known as protocol validation. Protocol validation technology essentially breaks (disassembles) database traffic and compares it to traffic forecasts. In the case where the actual traffic does not correspond to the forecasts, alerts or blocking actions can be put in place.

9) Unauthorized Copying of Sensitive Data [14]:

Many companies are striving to locate and properly maintain an inventory of all their databases. New databases can be created without the security team being aware and sensitive data copied to these databases can be exposed if the necessary controls are not applied. These "hidden" databases can contain potentially sensitive data such as transaction details, as well as customer and employee contact information. However, if data security people do not know the contents of these databases, it is difficult to ensure that the necessary controls have been applied. Whether intentionally or unintentionally, employees or hackers can then illegally access sensitive data. Old databases that have been forgotten and left out of the scope is an example. If no one manages these databases, the data is left unattended in view of prying eyes that should not access this data.

10) Exposure of Backup Data [15]:

Auxiliary database backup devices are generally not protected against possible attacks. As a result, several major security breaches have emerged, including theft of hard disks and database backup tapes.

A. Risk Management

Attacks on the SGBD itself can provoke:

- Known classic flaws (buffer overflows, authentication bugs ...).
- Vulnerabilities in the associated applications: Administration Web servers, Simple Network Management Protocol (SNMP) daemons, setuid root programs installed by the SGBD .

Misconfigurations

- Gradient authentication modes (.rhosts ...).
- Default passwords.
- Unsecure DB files (read by everyone).

Interception of passwords

- Listening to the network.
- By reading configuration files on disk.

Attacks on the applications

- SQL injection on web applications.
- Make misappropriation of requests made by an ERP.
- Make permissions too wide.

Attacks on the OS via the SGBD

- write / read files, execute commands.
  - The database runs with different privileges
  - Bypassing the security policy: PHP's 'safe_mode' eg.
- Critical in shared web hosts.

III. SECURE DATABASES AND ENSURE COMPLIANCE

A. Detection and Recognition

- We only secure what we know.
- Have a good mapping of sensitive resources (database instances, sensitive data).
- Automate recognition because the location of sensitive data keeps changing:
  - New applications or modified applications,
  - Mergers and acquisitions, etc.

B. Evaluate Vulnerabilities and Configurations

- Ensure security and absence of vulnerabilities:
  - Check how the database is installed in the operating system,for example, the privileges for files and executables for configuring the database
  - Check configuration options within the database itself, for example, at the end of which a number of connection failures will end up

being locked, or which privileges have been assigned to the critical tables.

- Verify that the database versions that you are running do not have known vulnerabilities.

C. *Strengthen Security*

- Vulnerability assessment often gives rise to a set of specific recommendations.
- This is the first step in strengthening the security of the database.
- This reinforcement has other elements that involve removing all unused functions and options

D. *Audit Changes*

- Once you have a reinforced configuration, you have to constantly monitor the security of the comic strip.
- Use audit tools:
  - Compare snapshots of configurations
    - ○ At the operating system level
    - ○ At the comic strip level.
- Immediately alert the administrator when a
  change likely to affect the security of the database.

E. *Monitor Database Activity (Database Monitoring: DAM)*

- Real-time monitoring of database activity is fundamental to mitigate risk
- Immediately detect intrusions and misuse
- The DAM can alert you to unusual access behaviors, potential indications of an SQL injection attack, unauthorized changes to financial data, increased privilege levels on accounts, or changes to the configuration that are run through SQL commands.
- Some DAM technologies provide application layer monitoring, which allows you to detect fraud through multi-tiered applications such as PeopleSoft, SAP and Oracle e-Business Suite, rather than through direct database connections

F. *Audit*

- Secure and indisputable control traces must be generated and maintained for all database activities that impact security, data integrity, or sensitive data retrieval.
- Most organizations use some form of manual auditing, using the native logging capabilities offered by their database.
- Non-tightness of responsibilities (it is easy for administrators todatabase to alter the logs)
- Need to acquire and manage considerable storage capacity to process massive amounts of unfiltered transaction information.

Fortunately, a new class of DAM solutions has emerged. These solutions allow for granular audits

- Independent of database systems
- With minimal impact on performance while reducing costs operational, through automation, centralization of SGBD rules and auditing standards, filtering and compression.

G. *Authentication, Access Control and Authorization Management*

Data and users are not all created equal.

- ADM must access users, guarantee the full responsibility of each of these users and manage privileges in order to limit access to the data.
- It must ensure that these privileges are respected, even for users of the database with the most privileges.
- It must also provide for a formal audit procedure in which it will regularly review enabling reports (also referred to as user rights clearance reports).

H. *Encryption*

Encryption makes it impossible to read sensitive data, which prevents attackers from accessing unauthorized data from outside the database.

Encryption must occur at several levels.

In-transit data must be encrypted to prevent network-level indiscretion and access when sending data to the database client.

But the resident data must also be encrypted, to prevent their extraction by an attacker, even if the latter manages to access the files.

IV. THE STRATEGIC AND TECHNICAL SECURITY MEASURES FOR GOOD DATABASE

A. *The Strategic Security Measures*

Never expose a BD server on the Internet:

If there is a need for direct access to the SGBD, it is necessary to use tunnels, a firewall with strong authentication opening the stream, ...

The administratorhas to be sure to filter the SGBD ports (1521, 1527, 3306, 1434, 135 ...)

and:

- Pay attention to cheap hosts!
- Never share a BD server.
- Hardening of the OS.

- Will limit the exploitation of faults.
- Apply standard procedures
- No installation of ancillary components, limitation of network services, regular application of security patches ...
- Do not install the examples, the related applications, ...

Hardening SGBD installation:

- Change the default passwords, delete the default accounts.

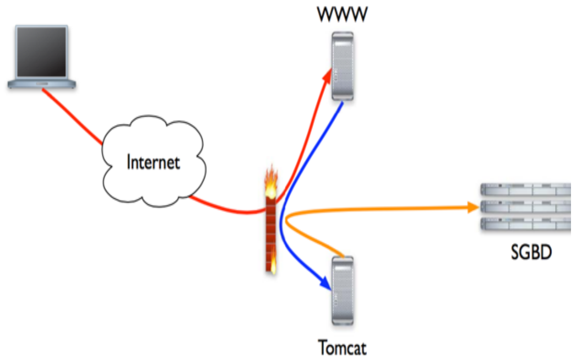

Figure 3: Apply the principle of defense in depth and partitioning by a layered architecture [16]

Separation of privileges:

- Register the roles in the application (admin, update, read ...)
- apply these roles in the assigned privileges.

Application audit:

- Talk about security with the developers (SQL Injection, buffer overflows, validation of entries ...).
- Research critical points, user flows ...
- Audit of sources Java, ASP, PHP, Perl, C ,etc

B. Technical Security Measures

This part represent different technical security mesures to ensure the security of database

1) Prevention of SQL Injection:

Three techniques can be combined to effectively combat SQL injection: Intrusion Prevention Technology (IPS), Request Access Control (see Excessive Privilege Override), and Event Correlation. IPS mechanism can identify most vulnerable stored procedures or SQL injection strings. Generally, IPS alone is unreliable since SQL injection give a lot of false positives.Security officials who rely solely on IPS technology would be bombarded with alerts about "possible" SQL injections. Practically, by constructing an SQL injection signature with another type of violation, such as a request access control violation, a very real attack can be very accurately identified. An SQL injection signature and other type of violation are unlikely to appear in the same query during a typical business operation.

2) Prevention of Excessive Privilege Abuse:

A solution toExcessive privilege abuseis the elimination of excessive rights.This requires the ability to identify excessive rights, i.e. rights that are not necessary for the user to perform his function. This is done by extracting the rights from the databases, correlating the rights with the business user and finally by analyzing these rights. This is a discouraging procedure that, if done manually, requires both time and resources. An automated solution can significantly reduce the time and resources needed and shorten the analysis process.

In order to better enforce access rights, access controls for granular queries are also required. Query access control refers to a mechanism that restricts access privileges to databases to a minimum of SQL (SELECT, UPDATE, etc.) and data operations. The granularity of the data access control must be extended from the simple table to the specific rows and columns within the same table. A sufficiently granular query access control mechanism would allow the previously described malicious university principal to update student contact information, but would trigger an alert if the student attempted to modify the notes. Query access control is useful not only for detecting excessive privilege abuse by malicious employees, but also for preventing most of the top 10 threats described in this document.

3) Prevention of Abuse of Legitimate Privilege:

The legitimate privilege abuse solution is database access control, which applies not only to the specific access requests described above, but also to the database access context. By applying a control rule for client applications, the time and location of the access request, etc., it is possible to identify users who use legitimate database access privileges in a suspicious manner.

4) Prevention of Privilege Escalation:

Privilege escalation abuses can be prevented by combining a traditional Intrusion Prevention System (IPS) and request access control (see Excessive Privilege Abuse section previously described). IPS inspects database traffic to identify patterns that match existing vulnerabilities. For example, if a specific function is known to be vulnerable, an IPS technology can either block all access to the vulnerable procedure, or (if possible) block only procedures with built-in attacks. Unfortunately, targeting only database access requests with precisely integrated attacks can be difficult using IPS alone. Many vulnerable database functions are used for legitimate purposes. It is not recommended to block all occurrences of these functions.IPS technology precisely separates legitimate functions from functions that include attacks. In many cases, the infinite variations of attacks make this distinction impossible. Under these conditions, IPS systems can be used in alert mode only (and not in blocking mode) since there are chances of getting false positives. To improve accuracy, IPS technology can be combined with alternative attack indicators such as access control for queries. IPS can be used to check whether or not the database access request uses a vulnerable function, while request access control controls whether or not the query matches a typical user profile. If a single request indicates access to a vulnerable function or an unusual user profile, an attack attempt is certainly in progress.

5) Prevention of Exploitation of Vulnerabilities in Vulnerable or Incorrectly Configured Databases:

In order to limit the threat risk of uncorrected and vulnerable databases, one must first assess the security status of the databases and correct any identified vulnerabilities and security gaps. Companies should periodically scan databases for any vulnerabilities and patches that are missing. The configuration assessments should provide a clear overview of the current configuration status of the data systems. These assessments should also identify databases that do not comply with the defined configuration rules. Any missing security patches should be deployed as soon as possible. If a vulnerability is discovered while the patch is not yet available, either because it has not yet been launched by the vendor or because it has not yet been deployed, a virtual fix should to be defined. Such a solution blocks attempts to exploit these vulnerabilities. Reducing the exposure window achieved by applying a virtual patch will help protect the database from attempts to exploit until a patch is deployed.

6) Prevention of Weakness of Native Audit:

Network-quality auditing systems address most of the weaknesses associated with native auditing tools.
High performance - devices based on network quality can apply line speed without any impact on database performance. In fact, by shifting the responsibility of auditing procedures to network applications, organizations can hope to improve the performance of databases.

Network-based auditing devices can operate independently of database administrators, allowing appropriate separation of audit functions from administrative functions. Also, because network devices are independent of the network itself, they are also invulnerable to privilege escalation attacks by non-administrator users.

Network-based auditing devices typically support leading database platforms that can apply uniform criteria and centralized auditing procedures across large, heterogeneous database environments. Combined, these features reduce database server operating costs, load balancing requirements, and administrative costs. They also provide better security.

Regular monitoring of logs helps identify risks and threats that can harm databases. If, for example, a malicious user (intruder) is able to outperform other defense systems, audits can identify violations after an attack, and also logs and audits can be used to repair the system (with system updates) and go back to the identity of the author of the attack.

7) Prevention of Denial of Service:

Denial of service prevention requires protections at multiple levels. This paper deals with the protections specific to the database not to mention the protections network, application and database that are needed. In this specific point, the deployment of control flow connection, of technology IPS, access control applications and control response time is recommended. Removing unwanted features and by configuring only what is needed for a database, denial of serice (DoS) can be prevented to an extent. Resource limits is another preventive measure which can make it challennging for the attakers to attack the sytem.Security patches needs to be applied on regular basis and administrators must run security report to constantly check security vulnarablities to prevent DoS.

8) Prevention ofVulnerabilities of Database Communication Protocols:

Protocol validation technology can be usuful to deal with the Vulnerabilities of database communication protocol. In this technology, database traffic is parsed and compared it with what is really expected.Researchers are working to create a mechanism which can provide proactive validation of protocol messages when they flow from client to servers. Any suspicious message that does not comply with expected pattern are flagged and discarded. This mechanism will greatly help dectacting bugs and worms and will restrict both known and unknown vulnaeribilities.

9) Prevention of Unauthorized Copying of Sensitive data:

In order to maintain accurate inventory of databases and accurate location of sensitive data, organizations should identify all databases on the network that contain sensitive data. The second step is to find out which types of sensitive or classified data are contained in the objects in the databases. The classification of data represents two major difficulties, the first of which is to locate sensitive data among the large number and large sizes of tables. The second difficulty is to find combinations of data which in themselves are considered harmless, but which, when combined with other data, form a combination of data considered sensitive. In order to adequately protect sensitive data, the necessary controls must be defined in accordance with the organization's data access policies, once an accurate inventory of databases and the location of sensitive data is available.

10) Prevention of Exposure of Backup Data:

All database backups should be encrypted. In fact, some vendors have suggested that future database management systems should not support the creation of unencrypted backups. Encryption of information from online production databases is often suggested, but the performance and inconvenience of managing cryptographic keys often makes this solution impractical and is generally recognized as a modest substitute for copyright controls. granular access described above.

V. CONCLUSION

After making a brief historical reminder of the databases, talking about authenticity, confidentiality and integrity, having seen the attacks and their parades from both a technical and an organizational point of view, we realize that securing databases is a major concern for the CIOs. To guard against these attacks, although database information is vulnerable to a large number of attacks, it is possible to dramatically reduce risk by focusing on the most critical threats. In dealing with threats, companies should meet the compliance and risk limitation requirements of the most highly regulated global industries. They are also required to call on specialists (through audits) to check their IT security practices. It should be noted that neglecting security is far too often dramatic. Indeed, the omnipresence of IT in companies implies that all sensitive information is contained in databases or at least in a server or computer connected to the network and therefore they are potentiallypiratables. indicators tend to prove that in the near future attacks attempts will be more and more common and therefore, we must remember the importance of carrying out, in addition to all technical measures, prevention to users, especially those brought to handle sensitive data. If we forge a request, we can hijack the initial request to execute the code of our choice.

REFERENCES

[1] Malik, Mubina, and Trisha Patel. "Database securityattacks and control methods." *International Journal of Information* 6.1/2 (2016): 175-183.

[2] Shivnandan Singh, Rakesh Kumar Rai, A Review Report on Security Threats on Database, *International Journal of Computer Science and Information Technologies*, Vol. 5 (3) , 2014.

[3] Simanta Shekhar Sarmah, Data Migration, *Science and Technology*, Vol. 8 No. 1, 2018, pp. 1-10. doi: 10.5923/j.scit.20180801.01

[4] Bertino, Elisa, and Ravi Sandhu. "Database security-concepts, approaches, and challenges." *IEEE Transactions on Dependable and secure computing* 1 (2005): 2-19.

[5] Shah, Arun, Robert F. Novy, and Robert A. Ertl. "Database security." U.S. Patent No. 7,167,859. 23 Jan. 2007.

[6] Cook, William R., and Martin R. Gannholm. "Rule based database security system and method." U.S. Patent No. 6,820,082. 16 Nov. 2004.

[7] Bertino, Elisa, Sushil Jajodia, and PierangelaSamarati. "Database security: research and practice." *Information systems* 20.7 (1995): 537-556.

[8] Pernul, Günther. "Database security." *Advances in Computers*. Vol. 38. Elsevier, 1994. 1-72.

[9] Lunt, Teresa F., and Eduardo B. Fernandez. "Database security." *IEEE Data Eng. Bull*. 13.4 (1990): 43-50.

[10] Garvey, Thomas D., and Teresa F. Lunt. "Cover Stories for Database Security." *DBSec*. 1991.

[11] Davida, George I., et al. "Database security." *IEEE Transactions on Software Engineering* 6 (1978): 531-533.

[12] Murray, Meg C. "Database security: What students need to know." *Journal of information technology education: Innovations in practice* 9 (2010): IIP-61.

[13] Burtescu, Emil. "Database security-attacks and control methods." *journal of applied quantitative methods* 4.4 (2009): 449-454.

[14] Shulman, Amichai, and C. T. O. Co-founder. "Top ten database security threats." *How to Mitigate the Most Significant Database Vulnerabilities* (2006).

[15] Basta, Alfred, and Melissa Zgola. *Database security*. Cengage Learning, 2011.

[16] Sarmah, S. (2019). Data Migration. [online] Article.sapub.org. Available at: http://article.sapub.org/10.5923.j.scit.20180801.01.html [Accessed 6 May 2019].

[17] Sourav Mukherjee *"Popular SQL Server Database Encryption Choices"* International Journal of Engineering Trends and Technology 66.1 (2018): 14-19.

[18] MonaliSachinKawalkar, Dr. P. K. Butey "An Approach for Detecting and Preventing SQL Injection and Cross Site Scripting Attacks using Query sanitization with regular expression". International Journal of Computer Trends and Technology (IJCTT) V49(4):237-245, July 2017.