IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661,p-ISSN: 2278-8727, Volume 18, Issue 2, Ver. III (Mar-Apr. 2016), PP 01-04
www.iosrjournals.org

Challenges to Data Base Security – A Futuristic View

Rekha Ahirwar[1], Rashi Saxena[2], Pankaj Yadav[3]

[1] *(Department Of Computer Science, Lecture Govt. Polytechnic Katani (Mp)*
[2] *(Department Of Computer Science, Lecture Govt. Polytechnic Katani (Mp)*
[3]*(Department Of Information Technology, Lecture Indira Gandhi Govt. Engineering College Sagar (Mp)*

Abstract: *In present scenario, database security is on prime consideration, for any organization because of its implementation and complex nature of system. It is generally associated with the dealing of retrieval of data from data base by various users. if all these retrievals are authentic then there is no problem unautenticity imposed creating threat & security is necessary issue for organization as well as if new internal user would be added then it also create serious threat which might be lead to security problem, this paper highlights main criteria of data base threat & how it can be mitigate or control to a permissible limit thereby we can safe our database.*

***Key word**: database security, data retrieval, miscofugered data, and sensitive data*

I. Introduction

All organizations public, governmental or private, small or large| depend on computerized information systems for carrying out their daily activity. At the heart of each such information system, there is a database. At a very general level, we can define a database as a persistent collection of related data, where data are facts that have an implicit meaning. For instance, an employee's name, social security number, or date of birth is all facts that can be recorded in a database. Typically, a database is built to store logically interrelated data representing some aspects of the real world, which must be collected, processed, and made accessible to a given user population. The database is constructed according to a data model which defines the way in which data and interrelationships between them can be represented. The collection of software programs that pro-vide the functionalities for defining, maintaining, and accessing data stored in a database is called a database management system (DBMS).Beside access and processing functionalities, each DBMS must also provide security functionalities to ensure the secrecy, integrity, and availability of the stored data [3]. Providing secrecy means providing secrecy means ensuring that data will not be disclosed to unauthorized users. Providing integrity means ensuring that data will not be modified in an unauthorized or improper way. In particular, integrity ensures that the stored data correctly reflect the real world. Providing availability means ensuring that the database will always be accessible by legitimate users for the accesses they are authorized for. The level of protection is required to fulfill the security purpose consists following clause.

Protection level: - A DBMS usually needs to protect data at a fine granularity level
Object differences there is a greater variety of object types in a DBMS than in an operating system. The typical object type in an operating system is a file; in a DBMS there can be relations (tables), tuples (rows within a table), attributes (columns within a table), indexes, metadata, and others.

Data interrelationships: - A database may include many logical objects with complex semantic interrelationships that must be protected. By contrast, the number of physical objects that the operating system protects is less and no semantic interrelationships are supported.

Dynamic versus static objects: - Data objects in a DBMS can be obtained by dynamically aggregating data from different physical objects in an operating system. By contrast, less tend to be more static making their protection easier.

Lifetime of data: - The lifetime and frequency of access of data in a DBMS is quite different than the lifetime of data stored as files in an operating system. User views of data While in an operating system, users are either granted or denied access to data (files), in a DBMS it is possible to give access to a portion of an object by defining different views for different users. Because of these differences, it is clear that some security requirements is needed.

Database security problem clauses: - following are important clauses under which data base security is important issue.

1- Exploitation of Vulnerabilities and Mis-configured Databases
2- Limited Security Expertise and Education
3- Unmanaged Sensitive Data
4- Weak Audit Trail
5- Privilege Abuse

Above these issue are associated with data base protection & industrial management is very aware for mitigation to fulfill the required purpose .

Exploitation of Vulnerable, Misconfigured Databases

It is common to find vulnerable and un-patched databases, or discover databases that still have default accounts and configuration parameters. Attackers know how to exploit these vulnerabilities to launch attacks against your organization. Unfortunately, organizations often struggle to stay on top of maintaining database configurations even when patches are available. Typical issues include high workloads and mounting backlogs for the associated Database administrators, complex and time-consuming requirements for testing patches, and the challenge of finding a maintenance window to take down and work on what is often classified as a business-critical system. The net result is that it generally takes organizations months to patch databases, during which time they remain vulnerable.

Limited Security Expertise and Education

Internal security controls are not keeping pace with data growth and many organizations are ill-equipped to deal with a security breach. Often this is due to the lack of expertise required to implement security controls, enforce policies, or conduct incident response processes. According to the Ponemon Institute 2014 Cost of Data Breach Study, for 30 percent of data breach incidents, the main root cause was classified as the "human factor"—in other words, a Negligent employee or contractor.

Unmanaged Sensitive Data

Many companies struggle to maintain an accurate inventory of their databases and the critical data objects contained within them. Forgotten databases may contain sensitive information, and new databases can emerge—e.g., in application testing environments—without visibility to the security team. Sensitive data in these databases will be exposed to threats if the required controls and permissions are not implemented.

Weak Audit Trail

Automated recording of database transactions involving sensitive data should be part of any database deployment. Failure to collect detailed audit records of database activity represents a serious organizational risk on many levels. Organizations with weak (or sometimes non-existent) database audit mechanisms will increasingly find that they are at odds with industry and government regulatory requirements. For example, Sarbanes-Oxley (SOX), which protects against accounting errors and fraudulent practices, and the Healthcare Information Portability and Accountability Act (HIPAA) in the healthcare sector, are just two examples of regulations with clear database audit requirements. Many enterprises will turn to native audit tools provided by their database vendors or rely on ad-hoc and manual solutions. These approaches do not record details necessary to support auditing, attack detection, and forensics. Furthermore, native database audit mechanisms are notorious for consuming CPU and disk resources forcing many organizations to scale back or eliminate auditing altogether. Finally, most native audit mechanisms are unique to a database server platform. For example, Oracle logs are different from MS-SQL, and MS-SQL logs are different form DB2. For organizations with heterogeneous database environments, this imposes a significant obstacle to implementing uniform, scalable audit processes. When users access the database via enterprise web applications (such as SAP, Oracle E-Business Suite, or PeopleSoft) it can be challenging to understand which database access activity relates to a specific user. Most audit mechanisms have no awareness of who the end user is because all activity is associated with the web application account name. Reporting, visibility, and forensic analysis are hampered because there is no link to the responsible user. Finally, users with administrative access to the database, either legitimately or maliciously obtained, can turn off native database auditing to hide fraudulent activity. Audit capabilities and responsibilities should ideally be separate from both database administrators and the database server platform to ensure strong separation of duties policies.

Privilege Abuse

Users may abuse legitimate database privileges for unauthorized purposes. Consider an internal healthcare application used to view individual patient records via a custom web interface. The web application normally limits users to viewing an individual patient's healthcare history—multiple patient records cannot be viewed simultaneously and electronic copies are not allowed. However, a rogue user might be able to circumvent these restrictions by connecting to the database using an alternative client such as MS-Excel. Using Excel and their legitimate login credentials, the user could retrieve and save all patient records to their laptop. Once patient records reach a client machine, the data then becomes susceptible to a wide variety of possible breach scenarios.

II. Mitigate Vulnerabilities

If vulnerability is deleted and the database vendor hasn't released a patch, a virtual patching solution should be used. Applying virtual patches will block attempts to exploit vulnerabilities without requiring actual patches or changes to the current configuration of the server. Virtual patching will protect the database from exploit attempts until the patch is deployed. Again, focus on patching high-risk vulnerabilities that can facilitate a DoS or input injection attack.

III. Separation Of Duties

Implement a DAP solution that delivers the performance, scalability, and flexibility to meet the needs of the most demanding environments. DAP solutions operate independently of database administrators, making it possible to separate audit duties from routine system administration. In addition, they operate independently of the database server and are invulnerable to privilege elevation attacks carried out by non-administrators

IV. Cultivate Experienced Security Professionals

To defend against a growing array of internal and external threats, hire information security personnel that are well versed in IT Security and have experience implementing, administering, and monitoring security solutions. Ongoing education and training are also important for growing deeper security knowledge and skills. Consider outside IT security and specialists to help with implementation, conduct security assessments and penetration tests, and provide training and support for your administrators.

Educate Your Workforce

Train your workforce on risk mitigation techniques including how to recognize common cyber threats (e.g. a spear-phishing attack), best practices around Internet and email usage, and password management. Failure to enforce training and create a "security conscious "work culture increases the chances of a security breach. The end result is well-informed users who are trained to securely function when connected

Identify and Classify Sensitive Data

Once you have constructed a catalog of databases, it is critical to understand which databases contain sensitive data. Scan the objects, rows, and columns of databases to pinpoint sensitive data. Use data classification solutions that are aware of data types such as credit cards, email addresses, and national identity numbers, and which enable users to add custom data types as well. Classification results should include the IP address and host name of the asset, and indicate the existence of sensitive data on that server. Automatically identifying sensitive data and personally identifiable information helps narrow the scope of security and compliance efforts.

V. Real-Time Alerting And Blocking

Monitor all database access activity and usage patterns in real time to detect data leakage, unauthorized SQL and Big Data transactions, and protocol and system attacks. When attempts to access unauthorized data occur, generate alerts or terminate the user session. Use a solution that leverages policies – both pre-defined and custom – that inspect database traffic to identify patterns that correspond to known attacks, such as DoS attacks, and unauthorized activities. Security policies are useful for not only detecting excessive privilege abuse by malicious, compromised, or dormant users, but also for preventing most of the other top ten database threats.

Detect Unusual Access Activity

Establish a comprehensive profile of each database user's normal activity. Monitoring for deviations from these baselines enables detection of DoS, malware, input injection, and anomalous activities. If any user initiates an action that does not fit their profile, log the event, generate an alert or block the user. Creating activity-based user profiles increases the likelihood of detecting inappropriate access to sensitive data.

VI. Data Protection

Automate the long-term data archival processes. Use solutions that can be configured to periodically archive data to external mass storage systems. Data should be optionally compressed, encrypted, and signed prior to archival. Encrypt sensitive data across heterogeneous database environments. This allows you to secure both production and backup copies of databases, then audit the activity of and control access to sensitive data from users who access databases at the operating system and storage tiers. By leveraging database auditing along with encryption, organizations can monitor and control users both inside and outside of the database.

VII. Conclusion

Database security is the greatest problem in present scenario , the above process is apply for protection which detecting the insecurity area of data & solution out line and its implementation in respect of threat can be protected the valuable sensitive data's **.**

Reference

[1]. Burtescu, E. Problems of Inference in Databases in „Education, Research & Business Technologies“. The Proceeding of the 9th International Conference on Economic Informatics, INFOREC Printing House, Bucharest, 2009

[2]. Burtescu, E. Databse security, in „Knowledge Management. Projects, Systems and Technologies“, International Conference „Knowledge Management. Projects, Systems and Technologies“, Bucharest, November 9-10, 2006, INFOREC Printing House, Bucharest, 2006

[3]. Hsiao, S.B. and Stemp, R. Computer Security, course, CS 4601, Monterey, California, 1995

[4]. McCarthy, L. IT Security: Risking the Corporation, Prentice Hall PTR, 2003

[5]. Proctor, P.E. and Byrnes, F.C. The Secured Enterprise, Prentice Hall PTR, 2002

[6]. Security Complete, Second Edition, SYBEX Inc., 2002