Journal of Network and Information Security
Volume 05 Issue 01, 2017
ISSN.: 2321-6859

Prevention of Insider Attack Against Database Access Control Mechanism

S. Vijayan[1], S. Madhusudhanan[2]
[1]M.Tech Cyber Security, Dept. of CSE, SNGCE, Kadayiruppu, Kerala, India.
Email: sheemamadhu@gmail.com
[2]Assistant Professor, M.Tech Cyber Security, Dept. of CSE, SNGCE, Kadayiruppu, Kerala, India.

Abstract: Existing SQL Access Control Mechanisms are extremely limited. Attackers can access the sensitive information through escalating their privileges. Practical attacks against existing database are increasing nowadays by using database features such as triggers and integrity constraints. Moreover the theories behind access control mechanism lack adequate security policies and attackers capabilities. This paper addresses the three main privilege escalation attacks by using database features and its prevention methods.

Keywords: DBMS, DB security, Access control mechanism, Insider attacks, Privilege escalation, SQL.

I. Introduction

Database maintenance has become an important issue in today's world. Addition or alteration of any field to an existing database schema cost high to a corporation. Whenever new data types are introduced or existing types are modified in a conventional relational database system, the physical design of the database must be changed accordingly. Data protection mechanisms are used to protect the data from external or internal attackers and many number of mechanisms are there for the protection. Like database access control mechanisms, data encryption standards, user authentications, database inferences are some of them. From the above protection mechanisms access control is the relevant one, thus we can add privileges to all the database users concerning about their database usage. But the attackers are too intelligent and strong now a days they try to attack the database both internally and externally. Commonly we can identify the external attacks by means of some antivirus software. In the case of internal attacks, here the attacker is a well-known party and he has the ability to access the database by using access control mechanism some privileges are given to all users. And the attackers escalate their privileges and violates security policies. So that we can tell existing access control mechanisms are extremely limited. Attackers can simply perform privilege escalation attacks by the use of database features such as triggers, views and integrity constraints. Databases in general usually store sensitive data, such as medical records, credit card numbers and government top secret information's. For this reason, the database administrators should ensure proper protection of sensitive data [1].

It is essential to provide adequate security measures to protect the stored data from both malicious outsider attacks also the well-known inside attacks. For this reason, the database administrators should ensure proper protection of sensitive data. To achieve database security it is dependent on the following properties:

- Confidentiality: Information should not be accessible to unauthorized users.
- Integrity: Data cannot be corrupted; only authorized users should be able to modify the data.
- Availability: Authorized users should be able to access the data reliably at all times.

Data protection is ensured by different components of a database management system (DBMS). In particular, an access control mechanism ensures data confidentiality and integrity. Whenever a subject tries to access a data object, the access control mechanism checks the rights of the user against a set of authorizations, stated usually by some security administrator. An authorization states whether a subject can perform a particular action on an object. Authorizations are stated according to the access control policies of the organization. Recently, the use of access control techniques has gained a lot of interest in the context of data privacy and management. There are two major challenges in an access control system:

- Defining correct and complete policies to control users access to the system and its resources, and
- Ensuring the resulting policies comply with the system requirements an high-level security/privacy policies.

An access control system is typically described in three ways: access control policies, access control models and access control mechanisms. Access control policies define rules concerning who can access what information, and under what conditions. These policies are enforced via a mechanism that mediates access requests and makes grant/deny decisions. The access control mechanism defines the low-level functions that implement the controls imposed by the policies. It must work

as a reference monitor, a trusted component intercepting each and every request to the system. An access control policy is comprised of a set of access control rules. A rule can have various modes (e.g., allow/deny/oblige/refrain). Since allow and deny rules are the most common ones, this paper focuses on these two kinds of rules. Allow rules authorize a subject to access a particular object. Deny rules explicitly prohibit a subject from accessing a particular object. When a subject requests to perform an action on an object, the corresponding rules are evaluated by the enforcement engine for that request [2].

Data is valuable assets of an organization. So its security is always a big challenge for an organization. Different governmental, non-governmental, and private and many other organizations have sensitive data on web servers that really need to be protected from attacker or intruders. Different access control mechanisms are used to protect data from these attacks, but the thing is they are limited and existing mechanisms are failed to prevent strong attacks by using database features such as triggers, views and integrity constraints. Finding and preventing these attacks is very important like what, how, when and where these attacks is performed.

II. Background

We first give formal introductions and information's regarding database threats and the security of the data. Now we will see background information regarding the proposed system.

A. Threats to Database Security

A threat can be identified with a hostile agent who either accidently or intentionally gains an unauthorized access to the protected database resources . In organizations there are top ten type of threats are recognized with can't only increase the risk of database exposure but also cause disastrous consequences on the entire organization. Some of these threats are described below [3].

- *Threat 1 - Excessive Privilege Abuse*

  When users (or applications) are granted database access privileges that exceed the requirements of their job function, these privileges may be abused for malicious purpose.
- *Threat 2 - Legitimate Privilege Abuse*

  Users may also abuse legitimate database privileges for unauthorized purposes.
- *Threat 3 - Platform Vulnerabilities*

  Vulnerabilities in underlying operating systems may lead to unauthorized access, data corruption, or denial of service.
- *Threat 4 - SQL Injection*

  In a SQL injection attack, attacker typically inserts unauthorized database statements into a vulnerable SQL data channel.
- *Threat 5 - Denial of Service*

  Denial of Service (DOS) is a general attack category in which access data is denied to intended users.

By these threat models we focused on Excessive privileges Privilege of database can be abused in many ways. User may abuse privilege for unauthorized purpose. Privilege abuse comes in different flavor: Excessive privilege abuse, legitimate privileges abuse and unused privilege abuse. This type of threat is most dangerous because authorized users are doing misuse of data. These privileges can be abused and creates unnecessary risk. Granting excessive permissions is problematic for two reasons. About 80% of the attacks on company data are actually executed by employees or ex-employees. Granting too many privileges or not revoking those privileges in time makes it unnecessarily simple for them to execute their wrongdoing. Some of these actions might even be executed inadvertently or without the perception of those actions being illegal Abuse of legitimate privileges can be considered database vulnerability, if the malicious user misuses their database access privileges.

B. Access Control

The purpose of access control is to limit the actions or operations that a legitimate user of a computer system can perform. Access control constraints what a user can do directly, as well as what programs executing on behalf of the users are allowed to do. In this way access control seeks to prevent activity that could lead to a breach of security. There are two classes of resources in any computer system: (active) subjects and (passive) objects. The ways a subject access an object are called access privileges. Access privileges allow subjects to either manipulate objects (read, write, execute, etc.) or modify the access control information (transfer ownership, grant and revoke privileges, etc.). Access Control models are broadly classified into three:

a. Discretionary Access Control (DAC)

Specify the rules, under which subjects can, at their discretion, create and delete objects, and grant and revoke authorizations for accessing objects to others [6]. That is: Govern the access of users to information on the basis of user's identity and predefined discretionary rules" defined by security administrator. The rules specify, for each user and object in the system, the types of access the user is allowed for the object. The request of a user to access an object is checked against the specified authorizations; if there exists an authorization stating that the user can access the object in the specific mode, the access is granted; otherwise it is denied. The policies are discretionary in that they allow users to grant other users authorizations to access the objects.

b. Mandatory Access Control (MAC)

MAC security policies govern the access on the basis of the classifications of subjects and objects in the system [6]. Objects are the passive entries storing information for example relations, tuples in a relation etc. Subjects are active entities that access the objects, usually, active processes operating on behalf of users. Access control in mandatory protection systems is based on the following two principles:

- No read-up/ Read down: A subject can read only those objects whose access class is dominated by the access class of the subject.
- No write-down/Write up: A subject can write only those objects whose access class dominates the access class of the subject.

c. Role – Based Access Control (RBAC)

In, Role-Based Access Control (RBAC), permissions are associated with roles, and users are made members of appropriate roles. This greatly simplifies management of permissions. Roles are closely related to the concept of user groups in access control. However, a role brings together a set of users on one side and a set of permissions on the other, whereas user groups are typically defined as a set of users only. RBAC Terminology is mainly based on objects, operations, permissions and roles [4].

III. Proposed System

Presents an effective security mechanism for database attacks, which affects access control mechanism. First we observe what all are the inside attacks by the use of database features such as triggers, views and integrity constraints. For that each attack contains one MySQL database with privileges. The insider attacks are more vulnerable and it is difficult to find out. This framework contains different access control privileges set to the genuine users from that privileges they escalate it and perform vulnerable actions. First of all the administrator divides all permissions to the database users, here we considering about insider attacks. That the database features such as triggers, view and primary key constraints. Database admins provide each user to use such features only for their work proceedings but the attacker minded people use those privileges not only for their work but also for the privilege escalation [5].

A. System Model

The proposed system consists of major insider attacks that affects database security features like integrity and confidentiality. This method mainly focusing on confidentiality attacks or data leakage attacks because we use MySQL database of version 14.14 it cannot allow the attacks by using database views so these kind of integrity attack is not possible.

a. Integrity Attacks

Integrity attacks allow an attacker to perform non-authorized changes to the database.

- *Triggers with Activators Privileges.*

Consider a database with two tables P and S and two users u1 and u2. The attacker is the user u1, whose goal is to delete the content of S. The policy is that u1 is not authorized1 to alter S, u1 can create triggers on P, and u2 can read and modify S and P. The attack is as follows:

1. u1 creates the trigger: Create trigger t on P after insert delete from S;

2. u1 waits until u2 inserts a tuple into the table P. The trigger will then be invoked using u2's privileges and u2 cannot insert any values into its table.

The prevention of this method includes if such attack happens then suddenly that information is given to administrator and admin add that attacker details into a blocked list and also he cant perform any trigger functions during his work. Also administrator can take actions against all the participants in the blocked list easily and he can easily find out who is an attacker.

b. Confidentiality Attacks

Confidentiality attacks allow an attacker to learn sensitive data.

- *Table Updates and Integrity Constraints*

Consider a database with two tables P and S. Suppose the primary key of both tables is the user's identifier. Furthermore, the set of user identifiers in S is contained in the set of user identifiers in P, i.e., there is a foreign key from S to P. The attacker is the user u whose goal is to learn whether Bob is in S. The access control policy is that u can read P and insert tuples in S. The attacker u can learn whether Bob is in S as follows:

- He reads P and learns Bob's identifier.
- He issues an INSERT statement in S using Bob's id.
- If Bob is already in S, then u gets an error message about the primary key's violation. Alternatively, there is no violation and u learns that Bob is not in S.

The preventive mechanism for this attack is if it happens administrator gets the message like an attack is detected and he can prevent it by blocking the privilege user id of all the users from the attacker. He can find out any ones identifier from his privileged table. Admin gently blocks the attacker from viewing the user id.

- *V. Triggers with Owners Privileges*

Consider a database with three tables N, P, and T. The attacker is the user u, who wishes to learn whether v is in T. The policy is that u is not authorized to read the table T, and he can read and modify the tables N and P. Moreover, the following trigger has been defined by the administrator.

CREATE TRIGGER t ON P AFTER INSERT FOR EACH ROW

IF exists(SELECT * FROM T WHERE id = NEW.id)

INSERT INTO N VALUES (NEW.id);

The attack is as follows:

- u deletes v from N.
- u issues the command INSERT INTO P VALUES (v).
- u checks the table N. If it contains v's id, then v is in T. Otherwise, v is not in T. The prevention for this attack is administrator is able to delete the trigger and also find the attacker.

IV. Conclusion

Database security is an important goal of any data management system. Database security is based on three important constructs confidentiality, integrity and availability. Access control maintains a separation between users on one hand and various data and computing resources on the other. In this work, different database attacks like data leakage and modification that affects to the security of database is presented. In the current scenario, there are limited access control mechanisms solve the three main privilege escalation attacks. Attackers act as genuine users with malicious intentions. These kind of issues is common now a days and in this work we collect mostly happened confidentiality and integrity attacks against access control methods and provide a preventive mechanism.

References

[1] Z. Nick, "Database security and cryptography," National Technical University of Athens, 2000.

[2] Q. He, and A. I. Anton, "Requirements-based Access Control Analysis and Policy Specification (ReCAPS)," *Information and Software Technology,* June 2011.

[3] T. F. Lunt, and E. B. Fernandez, "Database security," *ACM SIGMOD Record 19,* no. 4, pp. 90-97, 1990.

[4] Kriti, and I. Kashyap, "Database security and access control models: A brief overview," *International Journal of Engineering Research & Technology,* vol. 2, no. 5, pp. 743-751, May 2013.

[5] M. Guarnieri, S. Marinovic, and D. Basin, "Strong and provably secure database access control," *IEEE European Symposium on Security and Privacy,* January 2016.