Abstract—Databases are characterized as collections of data with sets of rules, that stipulate established relationships among the data. Many databases are organized around tables, comprised of rows and columns. Essentially, databases are software repositories, arranged so that users can update, add or remove data, at will. And just like any other computer system, databases have security requirements, averred as confidentiality, integrity and availability. Infelicitously, all of these requirements must be dealt with.

Most organizational processes revolve around some form of a database. A compromised database has the potential to do grave damage to an organization, not to mention the public notoriety relating to security breaches. With this in mind, I present work in Multilevel Security with respect to relational databases. I indicate how role based access combined with database views have the capability to deal with Multilevel Database Security. My objective is to seek a solution to database security that obdurately responds to the challenges and demands of operating in a hostile environment. It is my desire that this study will provide insight and inspire confidence to companies in their ability to protect their sensitive data.

Keywords—Database Security; Multilevel Security; Role Based Security; Database Views

I. INTRODUCTION

Information security is a substantial subject and permeates many areas of computer science. One subject of information security ensures no one can look at specific information without authorization. Another subject of information security calls for detecting unauthorized changes to data. It is also axiomatic that information security includes legitimate messages that have been sent, but were intercepted and replayed. As a result, computer security includes a multitude of different areas, all with many different vulnerabilities.

Many security problems are caused by someone with malicious intent, looking for financial gain, to gain notoriety, or in the case of a recent denial of service attack, to silence an adversary. Cyber criminals come from many diverse backgrounds, for example, the precocious student looking to prove something, to disgruntled employees and businessmen, all the way up to and including sovereign states. Nevertheless, most cyberattacks originate from insiders wishing to inflict harm. With the typical cast of characters, combined with prodigious amounts of spyware, malware, and insecure software, one should give serious thought about operating any application consisting of confidential information in an global web environment. Many applications are centered around databases, meaning, if the database is compromised, all the valuable data that is stored in the database is compromised.

The goal of any secure computer system, including database security revolves around three principal themes: confidentiality, integrity and availability. Confidentiality disallows unauthorized reading of data, integrity forbids unapproved writing of data, and availability centers on denial of service attacks, which are used in an attempt to diminish access to information.

There exist a smorgasbord of techniques in fulfilling database security goals; from encrypting the data, to enacting some kind of access control. Without the use of these methods, organizations are at risk of significant legal and financial consequences. The corruption, unauthorized disclosure, or theft of corporate resources could disrupt an organization’s operations, and have immediate, serious financial, legal, human safety, personal privacy and public confidence impact [1]. Although the mention techniques represent improvements in security, one should also mention the downsides, primarily to encryption. When data is encrypted there may be a degradation in system performance, for that reason, access control plays an important part in MDS (Multilevel Database Security).

Access control largely pertains to authentication and authorization. Authentication covers the use of passwords, biometrics and smart-cards. Authorization examines restrictions on users, leading to MLS (Multilevel Security). This notion of security is necessary when users and items at different levels access the same resources. The motivation behind MLS is the execution of some form of access control by restricting users to items for which they have the required clearance. Since there are many different ways of implemented an access control system, a large number of organizations have different security policies addressing each method. As a consequence, an access control system may fail to make the best use of resources, causing inefficiencies in a database system.

Within this framework, criterion-based multilevel database access control is proposed. This method introduces a manner of converting the security policy into a criterion expressions, i.e., boolean expressions, in terms of the security criteria. The security criterion expressions aims to combine the authorization rules for an object, subsequently, depicting an accurately representation of the authorization rules. This approach provides methods of using security criterion subsets
to identify a users’ security attributes that are dependent on a user job responsibilities. Finely grained control is carried out by the use of security criterion expressions and security criterion subsets.

Although criterion-based access control represent a unconventional way of looking at MDS, there some limitations to this method.

One limitation relates to the security mechanism used between users and the database. One must ask how robust is the security mechanism. Approximately, on August 15, 2009, a Georgian blogger was the target of a denial of service attack that targeted several social networking sites. Twitter was down for several hours beginning early Thursday morning, and it suffered periodic slowness and time-outs throughout the day [2]. Availability of service should be considered, because many commercial enterprises can not afford to be down or not operational for large periods of time. Thus, an important question is how attacks on the security mechanism mitigated.

Another limitations concerns the logical structure of the table. An additional column and row is added to the original table to store the rules. Databases are optimized for efficiency and consistency, so what would be the effect on database normalization.

And the last limitation deals with subtle database attacks. How does the security mechanism handle attacks such as inference attacks or, direct or indirect attacks? Direct attacks can take the form of looking for sensitive information just by performing a query that returns a limited number of records. Indirect attacks looks for results based on statistical result.

The goal of any secure computer system, including database security is the prevention of unauthorized disclosure of data while operating in a hostile environment. One security solution may satisfy one organization, but inappropriate for another. With this in mind, I look at the various techniques in securing a database with emphasis on role based access combined with database views. The next section outlines previous research concerning MDS. I consider the pros and cons and confer why my approach is reasonable. Then, I concluded with a summary, addressing drawbacks that I neglected.

II. RELATED WORK

There are three different categories of databases: relational databases, XML databases and object oriented databases. The difference between the database categories have a bearing on how they are structured. Relational databases are fully structured, XML databases are structured as semi and object oriented databases do not have any structure. Due to these differences, the database categories have different security needs.

However, they all have common database security requirements. Of course, there is always the need to physically secure the database from theft or power failure. Database integrity defining the structure, and element integrity describing the accuracy of the data, is another important concern. Also, access control, user authentication procedures, data availability, and data confidentiality are other characteristics of database security requirements. Because of the different security requirements, one solution may not be applicable in all situations. It is this reason that database security is addressed at different levels.

What makes database security a problem has to do with the sensitivity of each item in a database. This is due to the sensitivity of the same type of items may differ. Another problem with database security concerns the sensitivity of a combination of items, which may differ from the sensitivity of individual items. To keep an item sensitivity level, a strategy must outline which user have access to which item.

There are many database security solutions to include using encryption, partitioning the database, and using some kind of sensitivity lock scheme. Encryption can be performed on many aspects of database security. From password authentication, to actually storing encrypted data, cryptography has a function in database MLS. Yet, the use of encryption comes at a cost, primarily due to the need to decrypt the data for use. Partitioning the database divides the database into different security levels. The different databases are separated based on the classification of the data. Any data at the lower level can be reproduced at a higher level, allowing any users to see the complete database with respect to what their clearance allow. The downside of this model is the sizable amount of storage overhead, and the resource intensiveness of transmitting data from a lower level to a higher level. Nevertheless, this is one of the most secure models to be implemented [3]. In the scheme of using a sensitivity lock, each item has a lock which combines an identifier and a label. The negative aspect of this scheme is the susceptibility to trojan attacks.

There are other solutions with many advantages and disadvantages. I look at role based security combined with database views as a solution to MDS. This approach works well independently, as well as combined. This approach does not depend on an abstract security mechanism. Moreover, data encryption amalgamated with this solution offers added database security protection.

III. SOLUTION

I put forth a role based access control solution combine with database views to deal with MDS. For viewing data securely, this approach fills a niche. In the discussion below, I make the assumption pertaining to MLS, subjects are the users and objects are the items or data that is to be protected. Most businesses utilizing databases have a need for this type of arrangement. Suppose that a business have data restricted to upper management, other data available only to employees, while nonproprietary data available to the public. If the data is stored in one system the business must have a policy to deal with MLS.

---

Thelxione LLC
According to Ferraiolo and Kuhn, significant attention has been given to addressing the security needs of commercial and civilian government organizations. Despite the fact that civilian governments and commercial firms concerns are the protection of their data, such as customer data, employee data and intellectual property, these same organizations do not have the technology or know how to effectively manage the security risk involved for the protection of their data. Over ninety-five percent of survey respondents reported they would find value in solutions that helped them understand and prioritize database security needs within their infrastructure [4]. With the wide variety of businesses, comes a wide variety of security needs, thus the conventional MLS models using mandatory access control or discretionary access control may be a poor fit.

In many organizations the user does not own the data that constitutes the database. The organization is the owner, and control is based on what the employee does, instead of the data owners. So in some ways, access control can be determined by an individual’s role, for example geographic information system analyst or a doctor. A role based access control system places control decisions on what functions a user is allowed to perform within its organization. If compared to a discretionary access control system, in a role based access control system, users are restricted from passing permissions to other users.

Several advantages of role based access are that they can be created to contain group privileges and can be used to reduce security maintenance by not having to explicitly grant privileges directly to the user. Roles could contain the database administrator, users and others. For the database administrator and a user, it is obvious what should be granted to these individuals, but to others it may not be so clear. Other users could be restricted in what they have access to, thus a role could be assigned, instead of granting all control to a particular process.

A role can be regarded as exchanges a user conducts within the framework of an organization. The system administrator or database administrator allocates the roles and the exchanges, for example, this could include the ability of a doctor to practice medicine and add an entry to a record. Memberships in roles are granted and are also revoked by the system administrator or database administrator.

Role based access control systems have widespread use. From database use in controlling access to cryptographic authentication used in banking, commercial enterprises have used some form of role based access control. Role based security has been used to support integrity as defined by Stamps, integrity is concerned with preventing unauthorized writing [5].

As mentioned earlier, access control using roles are versatile in that it is easy for an administrator to grant and revoke membership to a set of roles in a computer system. If a new person comes into an organization, the administrator grants membership to a role. When the person leaves the organization, the role can be deleted. If an organization have comprehensive turnover rates, role based access control is a natural choice.

Separation of duties are easily achieved by the use of roles. A requirement may call for the separation of duties such that no one individual is allowed to perform all transactions for a particular item. One example may be a type of arrangement needed when one individual has to initiate a payment, and another person needs to authorize the payment. A single individual should not be allowed to execute both transactions. A role could be designated, one for each type of arrangement.

Role based access control system can be used in many setting and all with a minimum amount of effort. Roles are easily implemented and are easily understood. Role based access control provides a way of naming and detailing relationships between individuals and the rights each individual has.

### B. Database Views

Database views, in contrast, are used to limit the data that is presented to users. Views use results from a query to fill the contents of another table. For example, what if you have a table that includes point data, and attributes describing the data. After, you hire someone for only three months to write a report concerning the point data and its attributes. If the temporary worker is given complete access to the table, there is the increased risk of other data, such as confidential or intellectual property being compromised. To prohibit authorized access, a view is created, containing only the information the temporary worker is required to complete his or her job. Database views allow a higher level of data abstraction than the physical data, and also allow the enforcement of contextual relationships. With the use of views, finely grained security is easier achieved.

All data in the database is assigned a classification. Furthermore, each user has a clearance. If the user wants to access the data, they must have the correct clearance. The access classification that is connected with the data and the user is identified by a security label. Views need not be rescinded if a user’s clearance changes. Views can be constructed that will not return any data to the user unless the access classification of the user has a higher access classification of the data.

There are two types of views: access views and classification constraint views. Access views allow retrieval of data up through the user’s clearance from a relation that also may have higher-level data, without the need to retrieve from a higher-level container [6]. Joins can be used with access views and manipulated for display. Also, the retrieved data can be multilevel with classification down to each item. Data is entered through access views and are labeled on entry corresponding to the classification restrictions. Classification constraints clearly identify access relationships among stored and derived data, providing a way of handling contextual relationships and inferences. There is a third set of views called aggregation constraints, used to delineate and control access to a collection of information.

A view is a mapping from a database to a relation. Views, on occasion, are referred to the actual data returned when a view specification is requested from one instance of the database. If the access classification of the view specification
is not influenced by access classification of the user, then the view should not be applied to that user.

For some applications, users may want data presented statistically, visually or want to perform some complex calculation on the data. Users may create views and use numeric operators for formatting and displaying. Many of these operators will be hidden to the user and implicit in the retrieval of the data. The use of views represents, and filters a user’s subset of a database. Hence, views can be used to enforce MDS requirements. The drawback of this method is the complexity of creating and maintaining the views [7].

IV. Conclusion

Based on practical considerations such as ease of use and a broad knowledge base I consider the combination of role based access control coupled with views offer an alternative to criterion based access control. Users of this system have the ability to provide views based on authorization and roles. With the use of database views and roles, a multilevel database can be updated according to any users’ privileges.

MDS, I believe is better achieved by role based access with database views. This method does not depend on an abstract security mechanism and can provide strong security, including integrity. By acknowledge the existence that many applications implement roles intuitively, MDS is effortlessly achieved. Using role based access combine with database views is a simple and efficient solution.

Although a combination of role based access with views to institute MDS is viable, there are some issues that have not been clearly addressed and need further work.

- Inference
- Aggregation

Inference involves queries from data of a sensitive nature. There are several categories of inference: such as inference from queries based on sensitive data, queries based on statistical databases, and inference from data combined with metadata. The tools and techniques vary in combating inference channels. One technique uses semantic data modeling to detect inference channels and redesign the database to eliminate the channel. Another tools searches for inferences at query processing time. This has its drawbacks because the need to search the entire database.

Aggregation problems arise because data sensitivity can be deduced from the aggregate that can not be deduced from one individual member. Once this has been identified, there is wide variation in how to handle aggregation. Some examples include watching for ways in which the data changes over time.

REFERENCES