Data Warehouse Vulnerability and Security

Dr. S.L. Gupta, Sonali Mathur, Palak Modi

Abstract— The aim of this paper is to provide a view on some of the vulnerabilities existing in the data warehouse along with various security models and approaches to follow in order to make the data warehouse secure. Presently, a large amount of data is available over the internet, and since a data warehouse contains processed data from multiple sources, so its security has become a concerning issue. Following the introduction, various security models which can be adopted to provide security for data warehouse are briefly described. In the end, the metadata driven approach for providing security to a sample logistics datawarehouse is discussed.

Index Terms— Datawarehouse, Security, Security Model, Vulnerability, Metadata, Logistics Schema

1 INTRODUCTION

Due to the wide availability of huge amount of data, and to provide a way to integrate meaningful data from multiple sources, data warehouse has become a necessity for every organization. During the last several years companies are building up large databases and information repositories, with a way to have access to corporate data readily and easily to support decision making processes. Though the availability of data has led to ease of information access, but its security possesses numerous threats to this gargantuan data.

While there are many definitions of the primary requirements of security, the classical requirements are summarized by the acronym CIA. CIA is the acronym for confidentiality, integrity, and availability. All other security requirements such as nonrepudiation can be traced back to these three basic properties [1]. Confidentiality refers to limiting and providing access only to the authorized users, the users for whom the information is actually meant. Integrity questions the originality of data, whether the data has came from authentic resources or even those resources have entered the right data. Availability, unsurprisingly, means that information is up all the time.

2 THE DATA WAREHOUSE

2.1 Defining the Data Warehouse

According to William H. Inmon, "A data warehouse is a subject oriented, integrated, time variant, and nonvolatile collection of data in support of management's decision making process"[2]. This short but meaningful definition encompasses all the major functions provided by a data warehouse.

Subject Oriented: A data warehouse does not focus on all day to day operations of a data warehouse rather it considers only specific subjects which helps in decision support process.

Integrated: A data warehouse is usually made up of data from integrating various sources, whose underlying structure for representing data is different from each other.

Time Variant: Every data warehouse has an element of time associated with it; data stored in the data warehouse generally is the amalgamation of various historical data (e.g., the past 5-10 years).

Nonvolatile: A data warehouse is always a physically separate store of data transformed from the application data found in the operational environment. Due to this separation, a data warehouse does not require transaction processing, recovery, and concurrency control mechanisms. It usually requires only two operations in data accessing: initial loading of data and access of data [3].

A key challenge for data warehouse security is how to manage the entire system coherently – from sources and their export tables, to warehouse stored tables (conventional and cubes) and views defined over the warehouse [4].

2.2 Security Restrictions

A data warehouse by nature is an open, accessible system. The aim of a data warehouse generally is to make large amounts of data easily accessible to users, thereby enabling them to extract information about the business as a whole. Any security restrictions can be seen as obstacles to the goal, and they become constraints on the design of the warehouse. There may be sound business reasons for any security restrictions applied to the data warehouse, but it is worth noting that they may lead to a potential loss of information. If analysts have restricted access to data in the data warehouse it may be impossible for them to get a complete picture of the trends within the analyzed area. Checking security restrictions will of course have its price by affecting the performance of the data warehouse environment, because further security checks require additional CPU cycles and time to perform [5].

2.3 Security Requirements

For achieving the required security for your data warehouse system it is important to specify all the security requirements before the design of the data warehouse. Freezing the security

- *Dr. S.L. Gupta is currentlyworking as Professor in Birla institute of Technology, Noida and is Head of Research Division. PH-(+91)9811230453. E-mail:drslgupta@gmail.com*
- *Sonali Mathur is currently pursuing Doctor of Philosphy degree program in Computer Science and Engineering from Birla Institute of Technology,Mesra, Ranchi, India, PH-(+91)9811251170. E-mail:sonali.mathur10@gmail.com*
- *Palak Modi is currently pursuing Bachelors degree program in Computer Science Engineering from JSS Academy of Technical Educ ation, Noida, India, PH-(+91)9837074908. E-mail: palakmodi25@gmail.com*

requirements at the beginning of the requirement phase greatly improves system design.

3 VULNERABILTIES IN THE DATA WAREHOUSE

This phase requires the identification and documentation of vulnerabilities associated with the Data Warehouse environment. Identifying the data vulnerabilities associated with the data warehouse and the security measures to tackle these vulnerabities.

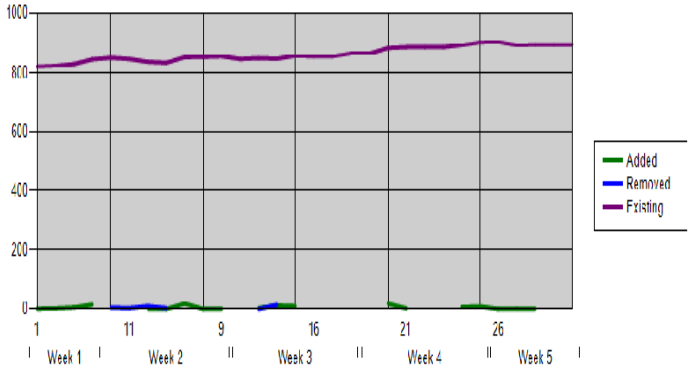

Figure 1: Vulnerabilities in the data warehouse [6]

The green line in Figure 1 represents the number of new vulnerabilities being discovered in the test environment. The blue line represents vulnerabilities fixed and removed compared to previous scans. The purple line represents the total number of vulnerabilities present. As can be easily seen, even though risks are being mitigated (blue), the pace is not fast enough to keep up with the total number of new vulnerabilities (green) being identified and the environment is progressively getting worse (purple)[6].

4 MODELS OF SECURITY

One of the pioneer foundations of a comprehensive security strategy involves implementing an appropriate level of access control to all data warehouse systems in an organization or an enterprise.

Access control restricts the scope of visibility of data for the user. A good access control mechanism ensures the user that there is only that much amount of data present in the warehouse which he is able to access, other data is completely invisible to that user. There are mainly four types of security models:

4.1 Mandatory Access Control

Mandatory Access Control (MAC) is the strictest of all levels of control. In computer security MAC is a type of access control in which only the administrator manages the access controls. When a user attempts to access a resource under Mandatory Access Control the operating system checks the user's classification and categories and compares them to the properties of the object's security label. If the user's credentials match the MAC security label properties of the object access is allowed. It is important to note that both the classification and categories must match. A user with top secret classification, for example, cannot access a resource if they are not also a member of one of the required categories for that object.

The obvious disadvantage MAC is that it requires a lot of planning before its implementation. Once implemented it also imposes a high system management overhead due to the need to constantly update object and account labels to accommodate new data, new users and changes in the categorization and classification of existing users [7].

4.2 Discretionary Access Control

In Discretionary Access Control (DAC) each user or user group is allowed to control access of their own data. Instead of a security label in the case of MAC , a user has a list known as the access control list(ACL) through which it can decide which user to give permission to access its individual data together with the level of access provided to that user (whether read only or read, write etc). A user can also modify the access control list, but only for those resources which the user owns. Flexibility is the key strength of Discretionary Access Control (DAC).

Limitation of DAC:

Global policy: DAC let users to decide the access control policies on their data, regardless of whether those policies are consistent with the global policies. Therefore, if there is a global policy, DAC has trouble to ensure consistency.

Malicious software: DAC policies can be easily changed by owner, so a malicious program (e.g., a downloaded untrustworthy program) running by the owner can change DAC policies on behalf of the owner [8].

4.3 Role Based Access Control

Role Based Access Control (RBAC), as the name suggests, is the access control based on user's role according to their job function within the organization to which computer system belongs. RBAC is the most widely used access control mechanism and it takes a real world approach in constructing the access control. A user or a group of users are provided with permissions according to their role in the Data warehouse. The disadvantage of following this approach is, an individual person can't change the permission provided to it according to his role. All people belonging to a particular role have same permission defined for them.

4.4 Rule Based Access Control

In Rule Based Access Control, a set of rules are defined, for example, rules for permitting access for an account or group to a network connection at certain hours of the day or days of the week. Like discretionary access control, rules are defined in an access control list (ACLs) associated with each resource object. Though, unlike mandatory access control, rules are not stiff and can be modified as a when needed.

5 DATA WAREHOUSE SECURITY BASED ON METADATA

Metadata is the most important part of the data warehouse, as it contains all the information related to the structural as well as access related aspects of the data warehouse. It is stored as a normal data in files known as metadata repositories. These repositories include information on the contents of the data warehouse, all the background processes going on in the data warehouse like backups, cleaning, and semantics of the data and information relating to the security of the data warehouse. Figure shows a simplified diagram of how a metadata points to all information in a data warehouse.

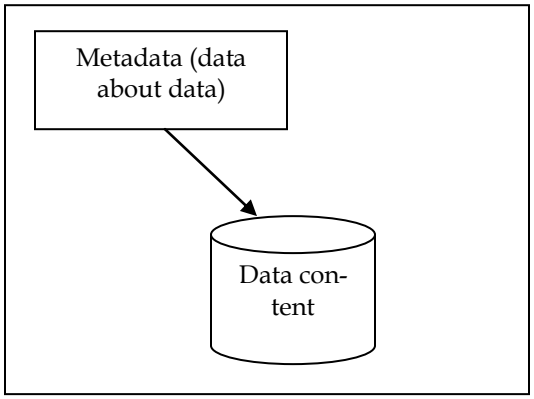

Figure 2: Metadata in the data warehouse

As an example a data warehouse on logistics data [9] is taken as shown in Figure 3, the typical structure of the metadata file on this particular schema is discussed.

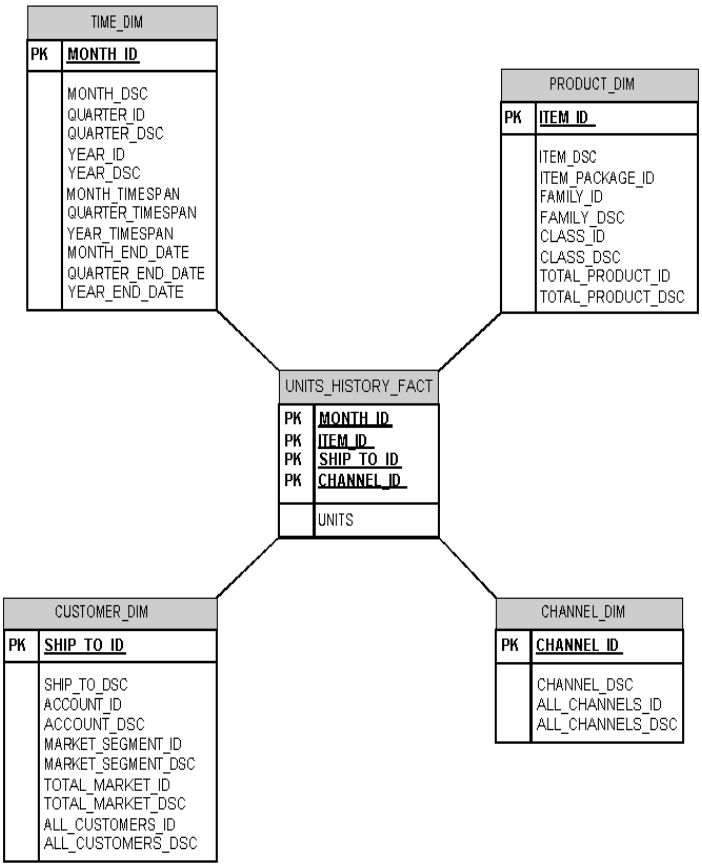

Figure 3: Star Schema of Logistics Data Warehouse [9]

The general description of the data in the data warehouse is discussed as:

```
dwh(
description=(
lastupdate="Mon,27-Feb-201, 12:02PM"
currency=(
USD=('usdollar" 'dollar")
)
Month=(
("Jan","Feb","Mar","Apr","May","June","July","Aug","Sep",
"Oct","Nov","Dec")
)
Ini=(
User=admin
Database=Logistics
)
)//EOF description
```

MODULE 1:

After this general description, the characteristic number (facts) is specified, consisting of composite primary key and the column unit. The composite primary key is composed of foreign key attributes MONTH_ID, ITEM_ID, SHIP_TO_ID, and CHANNEL_ID.

```
facts=(
(
    id="UNITS_HISTORY_FACT"
    sql="UNITS_HISTORY_FACT"
```

```
    des="Fact table consisting of primary key and number of
    units
    dim=(
    sql="TIME_DIM"
    key="MONTH_ID"
    )
    (
    sql="PRODUCT_DIM"
    key="ITEM_ID"
    )
        ...
    )
)
```

MODULE 2:
Now, the description of the attributes in the fact table follows. To the description of the attribute belongs:
-Its Identifier (id tag)
-Its sql description (sql tag)
-Its format (format tag)
-Its type (type tag) [4]
For example,

```
attr=(
(
    id="units"
    sql="units"
    fomat="#####"
    type=additive
    des=("Number of units', "units")
    )
    …
)
```

MODULE 3:
Next, all dimension table attributes are described.

```
dim=(
(
  sql="CUSTOMER_DIM"
  key="SHIP_TO_ID"
  des=("Customer details" "Customer")
  )
  …
)
```

MODULE 4:
Further the aggregations of each dimension are described are described (agg). Every aggregation possesses a predecessor and a successor field (prev, next). Example describes the MARKET_SEGMENT_DSC field of the CUSTOMER_DIM.

```
(
 prev=("MARKET_SEGMENT_ID")
 next=("TOTAL_MARKET_ID")
 sql=("MARKET_SEGMENT_DSC")
 des=("displaying market segment", "MAR-
 KET_SEGMENT_DSC");
 )
```

The aim of metadata file is to completely describe the data warehouse. Like in the security model just discussed, different users are provided with the different view of the data warehouse by providing metadata files derived from the original metadata file.

A user can "drill down" or "roll up" their data warehouse, without knowing that more data other than this is present in the data warehouse. The security, thus implemented, eliminates the chances of further inspection measures.

.

6 Conclusion

Since a data warehouse is generally recognized as a tool for managers or decision makers to quickly analyze large, multi-dimensional data, its consistency and reliability is of utmost importance. Security of a data warehouse can be achieved by first defining the security restrictions and requirements along with the possible vulnerabilities of the data warehouse. Various security models according to the organization's requirement can be followed. All security models aims to achieve a level of isolation for the user so that if a user uses operations such as "drill down" or "roll up" into the datawarehouse he is able to access only the information which he is authorized to see.

7 Future Work

Through this paper we have tried to combine all the major security models and security techniques in relevance to a Data Warehouse. We have also explained how metadata approach helps in increasing the security of a data warehouse. An example on logistics using metadata has been described in this research paper. Further in future, we plan to expand this model and encompass various other security techniques which will thereby provide a more reliable, secure and consistent Data Warehouse.

References

[1] Edgar R. Weippl, "Security in Data Warehouses", IGI Global, Data Warehousing Design and Advanced Engineering Applications, Ch 015, 2010
[2] W. H. Inmon, "Building the Data Warehouse", QED Technical Publishing Group, Wellesley, Massachusetts, 1992
[3] Jiawei Han and Micheline Kamber, "Data Mining Concepts and Techniques", Second edition, Morgan Kaufmann Publishers
[4] Arnon Rosenthal, Edward Sciore, "View Security as the Basis for Data Warehouse Security", Ceur Workshop Proceedings,Vol-28, 2005
[5] N. Katic, G.Quirchmayr, J.Schiefer, M.Stolba, A.M. Tjoa, "A Prototype Model for Data Warehouse Security Based on Metadata", in Proc. DEXA 1998, Ninth Int. Workshop on Database and Expert Systems Applications, IEEE Computer Society, pp. 300-308, Vienna, Austria, 1998
[6] Morey Haber, "Vulnerability Management in a Data Warehouse", 2010
[7] Mandatory, Discretionary, Role and Rule Based Access Control, www.techotopia.com, 2009
[8] Mandatory Access Control, " Computer Security (Syracuse Universi-

ty)", April 13,2009
[9] Technical Fundamentals for Implementation , Oracle® Business Intelligence Concepts Guide, " www.docs.oracle.com", 2005