International Journal of Security and Its Applications
Vol.8, No.1 (2014), pp.193-200
http://dx.doi.org/10.14257/ijsia.2014.8.1.18

Knowledge Based Secure Data Streaming in Virtual Environment

Anand. R[1] and Dr. S. Saraswathi[2]

[1]*Research Scholar, Computer Science And Engineering, Manonmaniam Sundaranar University, Tirunelveli, Tamilnadu, India*
[2]*Information Technology, Pondicherry Engineering College, Pondicherry, India*
nowhereanand@yahoo.com, swathi@pec.edu

Abstract

The process of transferring data within any physical environment is highly vulnerable and liable to face many security issues leading to sniffing of highly confidential data by any of the prevalent cyber attacks. Presently, there are various classic data retrieving methodologies such as CASE, PIVOT and SJP within a physical environment. However, these methods are not completely fail proof against unauthorized data retrieval. This paper aims to implement the concept of Knowledge Cube generation for securely transfer data with the help of Horizontal Aggregation method in virtual environment. The concept of 'Knowledge Cube' is referred as a multidimensional model composing of cubes and dimensions, which can be further interpreted so as to execute complex queries during any data retrieval process in both physical and virtual environment.

Keywords: *Knowledge cube, Virtualization, Virtual services*

1. Introduction

RDBMS has become a de facto standard for storing and retrieving large amount of data. This data is permanently stored and retrieved through front end applications. The applications can use SQL to interact with relational databases. Preparing databases needs identification of relevant data and then normalizing the tables. Aggregations are supported by SQL to obtain summary of data. The aggregate functions supported by SQL are SUM, MIN, MAX, COUNT and AVG. These functions produce single value output. These are known as vertical aggregations. This is because each function operates on the values of a domain vertically and produces a single value result. The result of vertical aggregations is useful in calculations or computations. However, they can't be directly used in data mining operations further. In fact summary data sets can be prepared and they can be used further in data mining operations [1]. The summary data can also be used in statistical algorithms [2, 3]. Most of the data mining operations expect a data set with horizontal layout with many tuples and one variable or dimension per column. This is the case with many data mining algorithms like PCA, regression, classification, and clustering [4, 2].

In relational databases, the data is not stored in hierarchical manner and hence significant effort is required to mine the data, by giving the exact data set as the input [1, 2]. It is common in relational databases to tune the databases before mining and in such a case, the desired physical data layout would be chosen, so as to give good performance during the most frequently run operations. This paper introduces a new class of aggregate function which is used to create data set in a layout, by automating SQL query writing. Since evaluating a horizontal aggregation is a challenging problem, we have introduced an alternative method

ISSN: 1738-9976 IJSIA

for data mining which extracts data with the help of Knowledge cubes and dimensions to achieve efficient evaluation.

2. Existing System

Existing SQL aggregations have certain limitations to prepare data sets because they return one column per aggregated group [3, 4]. It cannot handle complicated relationship between features. Existing SQL aggregations involve complex queries to build the data set and are time consuming since PIVOT operator can transpose by only one column [4]. No aggregation system focuses in the Data Mining area with involvement of Knowledge Cubes.

2.1. Issues in the Existing System

In the existing system, data are retrieved from the server, both in physical environment or in virtual environment. However, sequential queries are not used in this system because they are not secure during data streaming. Some of the issues in the existing system are as follows:-

- Sequential queries are not possible in CASE system.
- PIVOT operator can transpose by only one column in existing system.
- PIVOT operator requires removing unneeded columns (trimming) from the input table for efficient evaluation.
- One column per aggregation only can be achieved in SPJ methodology.

3. Proposed System

The proposed system involves multidimensional model composing of cubes and dimensions, to execute complex queries during any data retrieval process in both physical and virtual environment. This process is initiated with knowledge cube generation. Once the cube is generated, the option of pivoting the data or transferring the data from row to column and column to row is achieved. In addition, the option of Horizontal [4, 5] improvisation of Aggregated items will be achieved using the Pivot operator and generic aggregated methods can be achieved with the help of MDX querying concept. This can be achieved with the tool 'SQL Server Analysis Services' in which the data will be taken and transformed into knowledge cubes.

Subsequently, the performances of the existing and proposed data mining techniques on a common datum are analysed. The existing data mining techniques and the 'Knowledge Cube' technique are mutually incorporated as services in the proposed data mining system along with stringent security measures.

3.1. Architecture and Design of Proposed System

The concept of projecting data in different dimensions is done with the help of 'Data Sets', which are built by horizontal aggregations in a de-normalized layout (Eg: Point-Dimension, *etc.*,). Here we propose three fundamental methods to evaluate horizontal aggregation, which can be achieved by **CASE** (Exploiting the programming CASE construct), **SPJ** (Based on standard relational algebra operators (SPJ queries)) and **PIVOT** (Using the PIVOT operator, which is offered by some DBMSs).

Additionally, a methodology known as Knowledge cube aggregation is also integrated as a part of storage service along with other aggregations such as CASE, PIVOT and SPJ to obtain data and transform it into multi dimensional data sets for the purpose of data mining analysis.

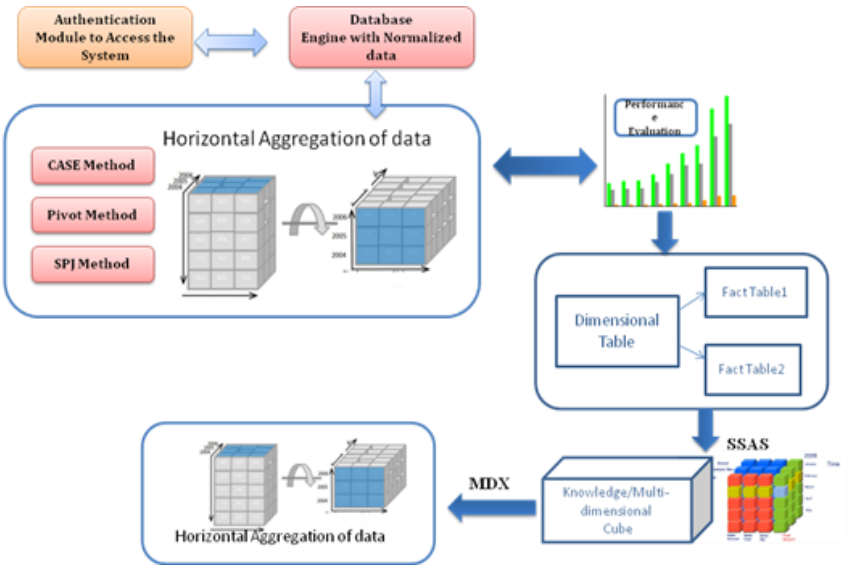


Figure 1. System Architecture

Securing the data transfer process is achieved at the Transport layer at both physical and virtual environment to avoid phishing/loss of data. Here we have explained a scenario of screaming data from one virtual machine to another. The first virtual machine namely vm1 runs as an instance of SQL and is hosted on the physical server. The second virtual machine namely vm2 contains all the front end tools (Eg: Visual Studio.net, java, *etc*.,). Initially, the configuration details are stored in vm1 SQL database. The applications running on vm2 sends a request for the configuration data from vm1, where the complete configuration details are stores as SQL database. Implementing security measures at Transport layer authorizes no data loss during streaming of data from one virtual machine to another. The configuration details are mined from vm1 to vm2 using SPJ and Knowledge cube methodologies to achieve rapid multidimensional data retrieval.

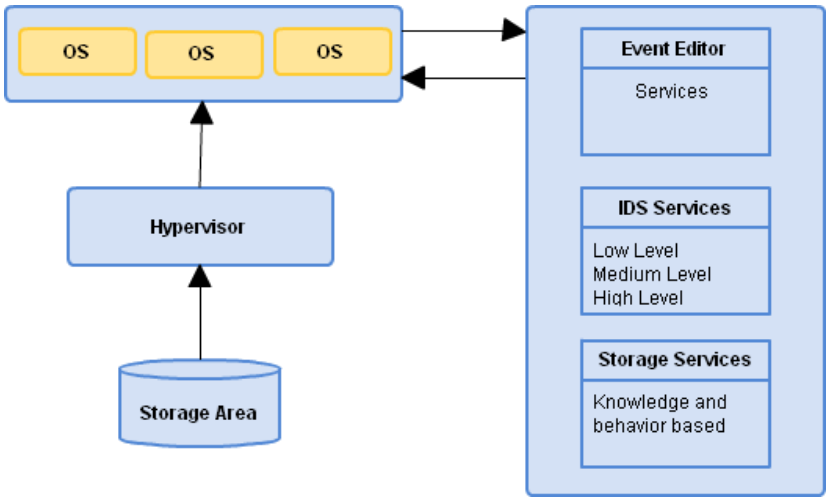


Figure 2. Security Analysis Model

In contrast to the behavior-based system, evaluation of knowledge based system includes both log system and communication system. Series of rules are created and applied to provide security and to illustrate security policies that the IDS should monitor. Audit data referring to a route discovery service, service discovery, and service request and response is collected. The series of policies that has been created tests the system's performance, although our scope does not include discovering new kinds of attacks or creating an attack to database.

In the virtual environment, data can be streamed from SQL server (back end) to any front end applications. During data transfer and streaming of data, some of the typical attacks like DDOS attacks, Service Identity linkage Attack, Attribute composition Attack, Service Status availability Attack, Untrusted Service provider Attack are possible.

To provide Virtual trust and Virtual security, intrusion detection system is used in addition to firewall in a virtual environment. An intrusion detection system (IDS) differs from a firewall [7]. Firewall limits access between networks to prevent intrusion and not to signal an attack inside the network [8, 9]. IDS monitor the attacks that are being generated within the system. Once when IDS finds the attack, it signals an alarm that an attack has occurred. Hence the steps to prevent attack are taken to ensure security. A system that terminates a connection is called an intrusion prevention system, and is another form of an application layer firewall.

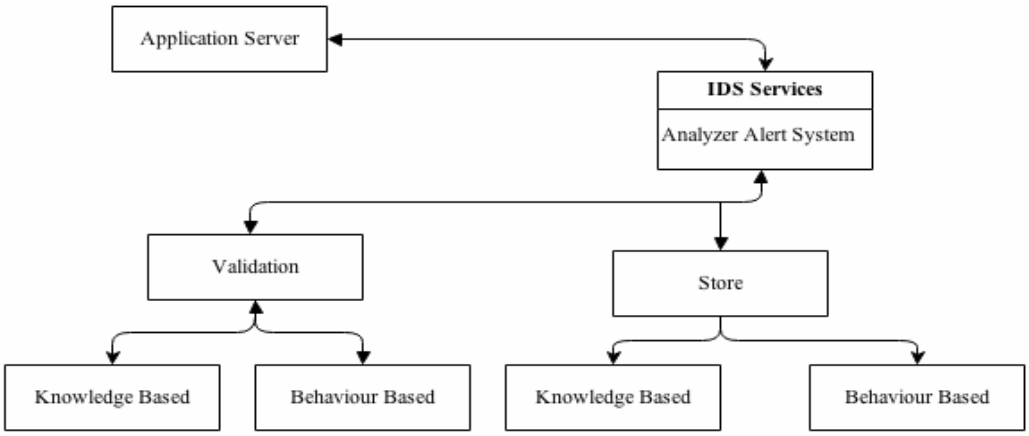


Figure 3. Detailed Security Analysis Model

For all trusted and untrusted transactions, there is a security risk due to vulnerability in the network and information domain. System administrators declare service providers for the trusted and untrusted services. Before starting the service end user agreement is verified for each client. Collision among many services over a session will increase the trust level or decrease the uncertainty level of the client and the associated data as well. The equation illustrates the inter-relationship between many services through their providers and the trust at any time will vary exponentially, it will be zero and equal to the original trust unchanged if there is no chance for collision. The virtualized network and the virtualized data provide a secured platform for the incoming queries of different nature. The context awareness is embedded in the records by elaborating the data set with additional fields. The context is used as a varying index for elaborating the data set which will be limited as per the level of privacy or confidentiality needed.

$$\sigma_{t+} = \sigma_{t-} + e^{-t/r\delta}$$

Where σ_{t+} is a Trust at time t,

σ_{t-} is an initial Trust at time t,

$e^{-t/r\delta}$ Is a current trust at time t, recovery time r, and collaborative factor δ.

Existing SQL aggregations have limitations to prepare data sets. In general, a significant manual effort is required to build data sets, where a horizontal layout is required. We propose simple, yet powerful, methods to generate SQL code to return aggregated columns in a horizontal tabular layout. This new class of functions is called horizontal aggregations.

Here, the data will be taken and it will be transformed into knowledge cubes. This can be achieved with Multi Dimensional queries. In addition to that, this paper introduces a performance evaluation on three methods CASE, SPJ and PIVOT. Experiments with large tables compare the performance of proposed method with existing three methods.

4. Implementation

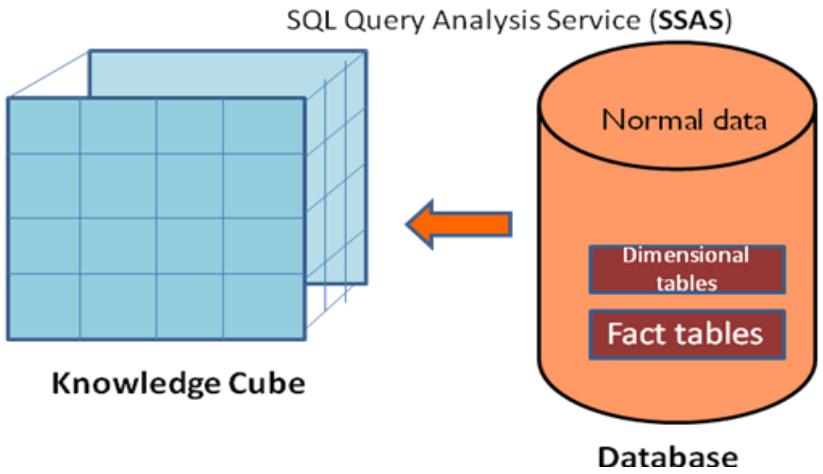
Implementation of the said system involves the following stages:-

- Knowledge Cube Construction
- MDX Query Check
- Performance Comparison and Evaluation
- Security Analysis

4.1. Knowledge Cube Generation

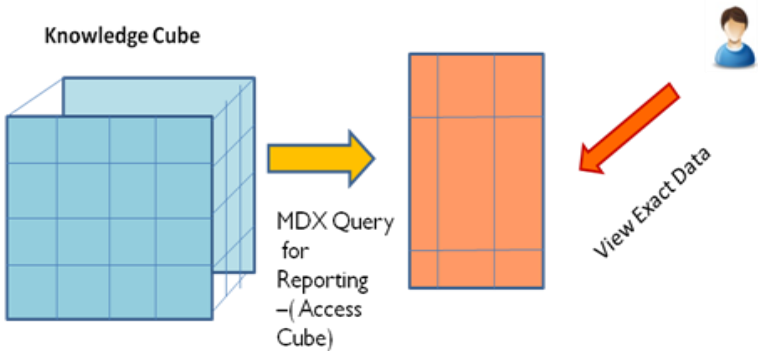
In this module, we are going to generate knowledge cubes from normal data by using SQL Server Analysis service (SSAS). These knowledge cubes have different dimensions to retrieve data at a fast rate.

- Configure Data source
- Configure Dimensions
- Configure Cubes



4.2. MDX Query Check

By using Multidimensional query we are to provide, a data security where the owner of that data can view exact data where as other users can view only a partial data. Here we can access Cube instead of table.

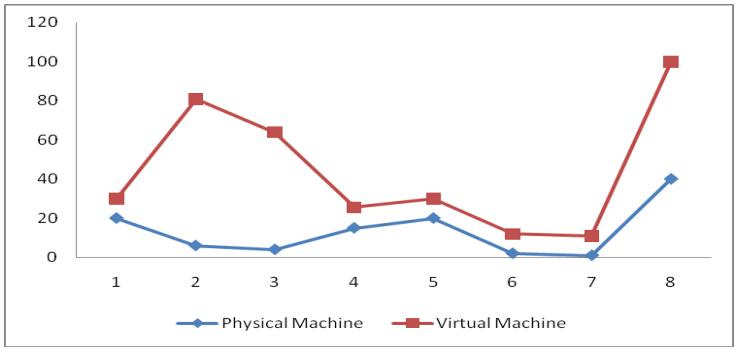


4.3. Performance Comparison and Evaluation

In this module, we are going to compare the performance of SPJ, CASE and Pivot method along with Knowledge Cube to find the efficiency of each and every method chronologically:-

Table 1. Comparison of CASE, PIVOT, SPJ and Knowledge Cube

System	Case, Pivot, SPJ				Knowledge Cube			
	Compilation Plan (ms)	Execution Plan (ms)	CPU Utilization	Maximum work done (%)	Compilation Plan (ms)	Execution Plan (ms)	CPU Utilization	Maximum work done (%)
Physical Machine	20	6	4	15	20	2	1	40
Virtual Machine	10	75	60	10.6	10	10	10	60



Graph 1. Comparison of Knowledge Cube Mechanism Physical Machine and Virtual Machine Environment

Table 1 and Graph 1 shows ,It is observed that Maximum work done is more than double the amount when Knowledge cube technique is used compared to Case, Pivot, SPJ. Relatively CPU utilization and Execution plan taken by knowledge cube is lesser both in case of physical machine and the virtual machine. In both physical and virtual environment Knowledge cube technique is efficient in amount of work done. Utilization of resources in both the technique differs from each other wherein Knowledge cube is very effective.

5. Conclusion

In this paper, we have implemented a new class of extended aggregate function called Knowledge Cube with the help of Horizontal aggregation method. This helps building up the complete data set for data mining. Additionally multilayered IDS at transport layer are implemented at transport layer to enhance and implement stringent security measures. The subsequent data streaming from source to destination is of greater trust and security, which has been analyzed. In addition, the performances of other aggregations such as CASE, PIVOT and SJP have also been evaluated. We have considered various other factors like scalability of number of features and instances, automation for handling large and heterogeneous data securely.

References

[1] C. Ordonez, "Statistical Model Computation with UDFs", IEEE Trans. Knowledge and Data Eng., vol. 22, no. 12, **(2010)** December, pp. 1752-1765.
[2] C. Ordonez, "Data Set Preprocessing and Transformation in a Database System", Intelligent Data Analysis, vol. 15, no. 4, **(2011)**, pp. 613-631.
[3] C. Ordonez and S. Pitchaimalai, "Bayesian Classifiers Programmed in SQL", IEEE Trans. Knowledge and Data Eng., vol. 22, no. 1, **(2010)** January, pp. 139-144.
[4] C. Ordonez and Z. Chen, "Horizontal Aggregations in SQL to Prepare Data Sets for Data Mining Analysis", IEEE Transaction on Knowledge and Data Engineering, vol. 24, no. 4, **(2012)** April, pp. 678-691.
[5] J. Gray, A. Bosworth, A. Layman and H. Pirahesh, "Data Cube: A Relational Aggregation Operator Generalizing Group-by, Cross-Tab and Sub-Total", Proc. Int'l Conf. Data Eng., **(1996)**, pp. 152-159.
[6] Y. Wang and L. Li, "Two-dimensional trust rating aggregations in service-oriented applications", IEEE Transactions on Services Computing, vol. 4, no. 4, **(2011)**, pp. 257-271
[7] J. L. Cacheiro, C. Fernández, E. Freire, S. Díaz and A. Simón, "Providing Grid Services Based on Virtualization and Cloud Technologies", LNCS, Springer, Heidelberg, vol. 6043, pp. 444-453.
[8] H. Benzina and G.-Larrecq, "J.: Some Ideas on Virtualized System Security, and Monitors", LNCS, Springer, Heidelberg, vol. 6514, **(2011)**, pp. 244-258.
[9] C. Wang, Q. Wang, K. Ren, N. Cao and W. Lou, "Toward Secure and Dependable Storage Services in Cloud Computing", Services Computing, IEEE Transactions, **(2012)**, pp. 220-232.

[10] W. Li and L. Ping, "Trust Model to Enhance Security and Interoperability of Cloud Environment", Lecture Notes in Computer Science, Springer, Cloud Computing, vol. 5931, **(2009)**, pp. 69-79.
[11] V. Pradeep Kumar and Dr. R. V. Krishnaiah, "Horizontal Aggregations in SQL to Prepare Data Sets for Data Mining Analysis".

