International Journal of Applied Engineering Research ISSN 0973-4562 Volume 10, Number 21 (2015) pp 42162-42167

Multi Attribute Schematic Relational Mapping Based Integrity Management For Security Enhancement Of Data Warehouse

G. Thangaraju

Research Scholar, Department of Computer Science, Karpagam University, Coimbatore-641 021

Dr. X. Agnise Kala Rani

Professor, Department of Computer Applications, Karpagam University, Coimbatore-641 021,

ABSTRACT:-
As the growing relational database becomes more sophisticated and grouped under a data warehouse, maintaining integrity of relational database has become a challenging task. To solve the problem of integrity management, a novel multi attribute schematic relational mapping (MASRM) algorithm is discussed in this paper. The method preprocesses the query to identify the relational mapping required to perform query processing and identifies a set of relational database to be accessed. Based on the rule mapping and the list of objects needs to be accessed, the user profile is verified for the access before performing the query processing. The user will be returned with the result, only if he has access to all the relational mappings required to produce the result. The method maintains a set of relational mappings for different possible queries to be produced as a rule set. From the rule set, for the input query?, required relational mappings are produced and mapped. For each of the rule matched and with the meta information of the rule, the method verifies the required access fields before producing result to the user. The proposed method improves the performance of the security management in data warehouse in an extended manner.

Index Terms: Data warehouse, Schematic Relations, Integrity Management, Relational mapping.

I. INTRODUCTION

The increasing volume of relational data needs to be stored in data warehouse where large relational data can be stored. The data present in warehouse can be used in various ways, particularly in generating business intelligence' which supports business peoples in many ways. For example, a business man may think of getting knowledge about the market sale of his product in various regions in different time window. By producing a query in the warehouse about generating intelligence and using the intelligence produced, he can change the market strategy of the product. The product manager or the business man can focus on weak zone to improve the market sale of a particular product. Sometimes the business intelligence can be used to identify a group of people who purchase his product in more number, basing on their knowledge. He can thus produce a different range of products towards different group of people where they are grouped according to economic status.

Such business information can be accessed by other people also. But the user from different perspective has to be restricted to maintain the data integrity. The warehouse information has a variety of personal information of their clients or users who purchase the product. The personal information has to be secured from the external user and has to be restricted at different levels. Such a security management is necessary in large scale data warehouse. But the data warehouse has access to different peoples who can execute a variety of queries to get a set of knowledge. The problem is how the external users at different levels are restricted from accessing different attributes or relational objects of the data warehouse.

The data warehouse has a number of relational databases where each has a specific schematic relation, and the whole warehouse can be presented in the form of a set of relations. Each database can be lined to different data base and they can have relation between them. The user query can have any number of levels in nested forms. By separating them, one can identify the number of relations it has and number of database the query needs to access. In any data relation, there will be a number of attributes and each has different access permission. Not all the attributes have same levels of access but have different access restrictions. Similarly not all the users of the system have the same level of access permission and everybody has a different access permission which will vary according to the user profiles.

Relational mapping is the process of identifying the relational database the query needs to access and identifying the number of unidentified attributes being accessed by the query. By performing relational mapping, the process can identify the relational schema the user has access and the relational scheme the query needs to access and so on. By performing this process, the user query can be categorized as authorized or overriding.

II. RELATED WORKS

There are a number of security protocols which have been discussed for the enhancement of security in data warehouse and are discussed few of them a here in this section.

A Schematic Technique Using Data type Preserving Encryption to Boost Data Warehouse Security [1], presents a method of in progression fortification based on an encryption scheme which preserves the data type of the plaintext resource. We suppose that this method is particularly

companionable for multifaceted data warehouse environments. The first processing step involves replacing each plaintext character in the string by an integer that represents its position, or index, within the chosen alphabet. This number is between zero and one less than the total number of characters in the alphabet. If a plaintext character is not in the valid alphabet, it is copied to the output and removed from the string to be encrypted.

Integrating multiple sources of information in text classification using whirl [2], presents an approach for corpus-based text classification based on WHIRL, a database system that augments traditional relational database technology with textual-similarity operations developed in the information retrieval community. Not only does the approach perform competitively when compared to state-of-the-art text classification methods, we show that it enables the incorporation of a range of hitherto unexploitable sources of information into the classification process in a fairly robust and general fashion.

Balancing Security and Performance for Enhancing Data Privacy in Data Warehouses [3], propose a data masking technique for protecting sensitive business data in DWs that balances security strength with database performance, using a formula based on the mathematical modular operator. This solution manages apparent randomness and distribution of the masked values, while introducing small storage space and query execution time overheads. It also enables a false data injection method for misleading attackers and increasing the overall security strength. It can be easily implemented in any DataBase Management System (DBMS) and transparently used, without changes to application source code. Experimental evaluations using a real-world DW and TPC-H decision support benchmark implemented in leading commercial DBMS Oracle 11g and Microsoft SQL Server 2008 demonstrate its overall effectiveness. Results show substantial savings of its implementation costs when compared with state of the art data privacy solutions provided by those DBMS and that it outperforms those solutions in both data querying and insertion of new data.

DES with any reduced masked rounds is not secure against side-channel attacks [4], focus on the security of DES with reduced masked rounds against side-channel attacks; we propose differential based side-channel attacks on DES with the first 5, 6 and 7 rounds masked: they require 217. 4, 224, 235. 5 chosen plaintexts with associate power traces and collision measurements, correspondingly. Our attacks are the first known side-channel attacks on DES with the first 5, 6 and 7 rounds masked; our attack results show that DES with any reduced masked rounds is not secure against side-channel attacks, i. e., in order for DES to be resistant to side-channel attacks, entire rounds should be masked.

The Forrester Wave: Enterprise Data Warehousing Platforms [5], found the EDW market increasingly competitive, as illustrated by tighter clustering of top vendors. Teradata, Oracle, Sybase (SAP), and IBM lead by offering high-performance, scalable, flexible, and robust EDW solutions. Teradata provides the most scalable, flexible, cloud-capable EDW solution in today's market. Oracle has built its Exadata Database Machine into a formidable new product family. Sybase, recently acquired by SAP, continues to enhance IQ's massively parallel columnar technology for real-time analytics. IBM has ramped up its EDW solution focus and now sets the pace on petabyte-scale Hadoop integration. SAP is rapidly evolving and converging BW and BWA into a high-performance EDW with an in-memory, columnar infrastructure optimized for real-time analytics. EMC Greenplum demonstrates solid execution and continued innovation. Netezza (recently acquired by IBM) has integrated in-database analytics into its high-performance EDW appliances. Microsoft has launched cost-effective EDW appliances for midmarket and large enterprises, and Strong Performer Vertica Systems continues to enhance its high-performance all-columnar EDW architecture.

Oracle Advanced Security Transparent Data Encryption Best Practices [6], provides best practices for using Oracle Advanced Security Transparent Data Encryption (TDE). Oracle Advanced Security TDE provides the ability to encrypt sensitive application data on storage media completely transparent to the application itself. TDE addresses encryption requirements associated with public and private privacy and security mandates such as PCI and California SB1386. Oracle Advanced Security TDE column encryption was introduced in Oracle Database 10g Release 2, enabling encryption of application table columns, containing credit card or social security numbers.

Generating Databases for Query Workloads [7], introduces MyBenchmark, an offline data generation tool that takes a set of queries as input and generates database instances for which the users can control the characteristics of the resulting workload. Applications of MyBenchmark include database testing, database application testing, and application-driven benchmarking. We present the architecture and the implementation algorithms of MyBenchmark. They also present the evaluation results of MyBenchmark using TPC workloads.

Implementing Log Based Security in Data Warehouse [8], proposes an implementation of behavior analysis based on logs. To ensure data privacy various solutions have been proposed and proven effective in their security purpose. However they introduce large overheads making them unfeasible for data warehouse. Therefore to avoid these overheads and to increase data security, data masking approach have been proposed. Solution manages the randomness of masked values which increases the overall security strength. Log Analysis for intrusion detection is the process use to detect attacks on a specific environment using logs as the primary source of information. For future perspectives it will be beneficial as the authors will come to know whether it is simple access or attack.

Security in Data Warehouses, IGI Global [9], discusses that the last several years have been characterized by global companies building up massive databases containing computer users' search queries and sites visited; government agencies accruing sensitive data and extrapolating knowledge from uncertain data with little incentive to provide citizens ways of correcting false data; and individuals who can easily combine publicly available data to derive information that – in former times – was not so readily accessible. Security in data warehouses becomes more important as reliable and appropriate security mechanisms are required to achieve the

desired level of privacy protection.

Balancing Security and Performance for Enhancing Data Privacy in Data Warehouses [10], propose a data masking technique for protecting sensitive business data in DWs that balances security strength with database performance, using a formula based on the mathematical modular operator. Our solution manages apparent randomness and distribution of the masked values, while introducing small storage space and query execution time overheads. It also enables a false data injection method for misleading attackers and increasing the overall security strength. It can be easily implemented in any DataBase Management System (DBMS) and transparently used, without changes to application source code.

Learning SQL for Database Intrusion Detection using Context Sensitive Modeling [11], propose a novel approach for modelling SQL statements to apply machine learning techniques, such as clustering or outlier detection, in order to detect malicious behavior at the database transaction level. The approach incorporates the parse tree structure of SQL queries as characteristic e. g. for correlating SQL queries with applications and distinguishing benign and malicious queries. We demonstrate the usefulness of our approach on real-world data.

Using Secret Sharing Algorithm for Improving Security in Cloud Computing [12], making use of the strongest cryptographic algorithm named Shamir's secrete sharing algorithm, has a number of advantages including security, client-side aggregation. It claims that security is maintained even when k or more servers collude. It is a fact that much research has been done to ensure the security of the single cloud and cloud storage whereas multi-clouds have received less attention in the area of security. We affirm the moving to multi-clouds due to its ability to decrease security risks that affect the cloud computing user. Rephrase the proposed work provides confidentiality, data integrity, improved availability and capacity to handle multiple requests at a time.

HAIL (High Availability and Integrity Layer) [13], which is a combination of Proofs and cryptography, presented in the year 2009 is used to control multiple clouds. It ensures data integrity and service availability. But the limitation of HAIL is that it needs code execution in their servers and it does not deal with multiple versions of data.

RACS (Redundant Array of Cloud Storage) is a protocol for intercloud storage presented in the year is 2010. This technique is similar to RAID and normally used by disks and file systems and replication offers better fault tolerance. But the problem is it is unable to cooperate with vendor lock-in and economic failure. Cachin [14] presented a design for intercloud storage named ICStore in 2010. ICstore is client centric distributed protocol which can handle data integrity issue but has poor performance in case of data intrusion and service availability. It is the same with the encrypted cloud VPN.

All the above methods have the problem of ensuring security and restricting access in data warehouse and produce fewer throughputs.

III. METHODOLOGY

The multi attribute schematic relational mapping (MASRM) process identifies the set of relations the query covers and the number of relational database the user has access and the number of attributes the user does not have access and so on. This verification is performed according to the relational rule set maintained by the proposed method. The method maintains set of user profiles and relational rule set, using which the method computes the relational access measure and relational violent score. Based on the computed measures, the method decides the status of query and returns results to the user. The complete process can be divided into a number of stages, namely Query Preprocessing, Relational Rule Generation, Rule Mapping, and Relational Schematic Score Computation. Each of the stage is explained in detail in this section.

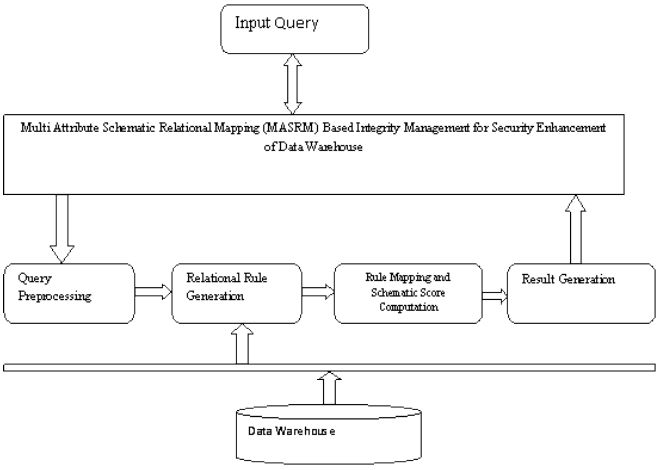


Figure 1: Architecture of the proposed method

A). *Query Preprocessing*

At this stage, the input query is split into terms and the data warehouse meta data have been read and from the input term set, the set of databases specified are identified. From the number of databases identified, then the numbers of functions mentioned are identified. Based on identified functions, the method identifies the number of schematic relations that has been used is identified. The identified features are used as the input to the next stage of query processing.

Algorithm:
Input: Data warehouse Metadata Md, Query Q
Output: Relational sets Rs, Functions set Fs.
Start
Read metadata Md.
Read input query Q.
Term set Ts = $\int Split(Q,")$
For each term Ti from Ts
Identify the relational database name.
If $\int_{i=1}^{size(md)} Md(i).name == Ti$ then
Add to relational set Rs = $\sum Relations(Rs) + Ti$
End
If $\int_{i=1}^{size(functions)} Function.name == Ti$ Then
Add to function set Fs = $\sum Functions(Md) + Ti$ End.
Stop.

The above displayed algorithm performs preprocessing of input query and identifies the set of relations and functions which have been mentioned in the input query submitted. The identified set will be used in schematic score computation later.

B). *Relational Rule Generation*
The relational rule generation is performed using the meta data of data bases present in the data warehouse. First the method identifies the number of relations a single database has with other databases. Then, for each relational database, the method identifies the set of sensitive and non sensitive members of the database. From identified relations and the sensitive, nonsensitive attributes, the method generates the rule accordingly.

Pseudo Code of Relational Rule Generation:
Input: Meta data Md.
Output: Relational Rule set Rs.
Start
Read Metadata Md.
For each relational database Rd
Identify the number of attributes.
NA = $\sum Attr \in Rd$
Identify number of relations it has with other database.
NR = $\int_{i=1}^{size(Md)} \sum Relation \in Md(i)$
For each relation R from NR
Compute number of sensitive attributes.
NSA = $\sum Attr(R) == Sensitive$
Compute number of non sensitive attributes.
NNSA = $\sum Attr(R) == NonSensitive$
Generate Rule Ru = {NA, NR, NSA, NNSA}
Add to rule set Rs = $\sum(Ru \in Rs) \cup Ru$

End
End
Stop.

The above displayed algorithm computes the schematic relational rules using the meta data of data warehouse. The generated rule set will be used to compute the schematic relational score to perform query processing.

C). *Rule Mapping and Schematic Score computation*
At this stage, using the preprocessed results and the generated rules, the method performs mapping of rules. First the method reads the user profile which has information about the access permission the user has and from the user profile for the user, the method identifies the set of all schematic relation the user has access is identified. Then from the preprocessed result, the method identifies the presence of the identified relations. With the identified results, the method performs mapping with the rule generated and identifies the set of rule gets matched. Using the rule match, the method identifies the presence of access for the sensitive and nonsensitive attributes from the term set obtained at the preprocessing stage. Using all these values the method computes the schematic score.

Pseudo Code of Schematic Score Computation:
Input: Rule Set Rs, Term Set Ts, Functions Set Fs, User Profile UP.
Output: Boolean.
Start
Read User Profile Up.
Identify set of all relation the user has access.
Ua = $\sum_{i=1}^{Up} \sum Relation(Up).access == User$
For each relation Ri from relational set Rs
If Ua$\in Ri$ *Then*
Count = count+1;
Else
Add to Negotiation Set Ns.
End
If count>RThreshold Then
Add rule to the match set.
Rm = $\sum Rule(Rm) \cup Ri$
End
End
For each rule Ri from Mr
Compute schematic relation score SRS.
SRS = $\frac{\sum Relation(ra \in Ri))}{size(ra)} + \frac{(NSA - NSA(Ra))}{NA} + \frac{(NNSA - NNSA(ra))}{NA}$
End
If SRS>STh then
Return true
Else
Return false
End
Stop

The above discussed algorithm computes the schematic relational score for the input query using the set of relations being generated. The computed value will be used to perform query processing.

D). *Result Generation*
At this stage, the method performs all the operations above mentioned by using the procedures. First, the method

performs preprocessing and then generates rule using the meta data. Second, the method computes the schematic relation score, and based on the value returned, the method executes the query for the user and returns the results to the user. The result to the user are fully based on the values of schematic relational score, obtained on the query submitted which are computed depending on the user profile and relational rules.

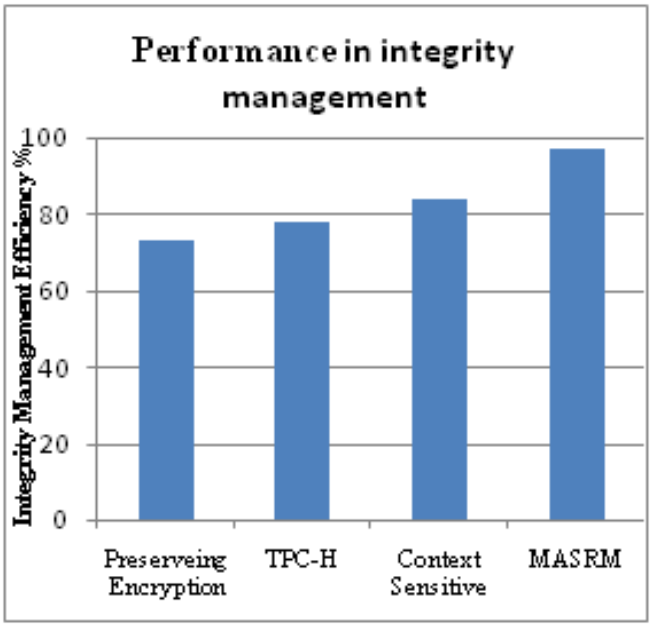
IV. RESULTS AND DISCUSSION

The proposed method has been designed and implemented using the SQL data base which has a number of relational databases. The warehouse has been created with thousands of relational databases and has been evaluated from the user query in lacks. The details of evaluation have been listed below:

Table 1: Details of simulation parameters

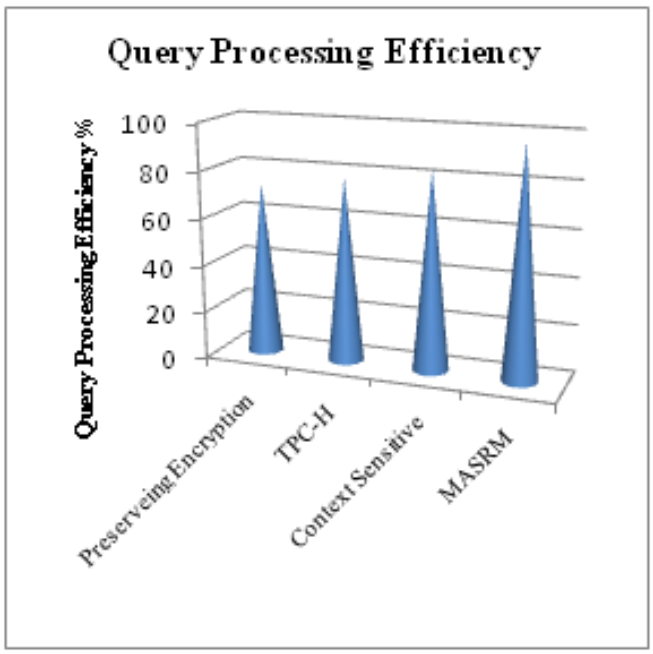
Parameter	Name
Data warehouse Tool	SQL
Number of database	5000
Number of queries	20000
Number of users	500

Table 1 shows the details of simulation parameters being used to perform evaluation of the proposed approach.



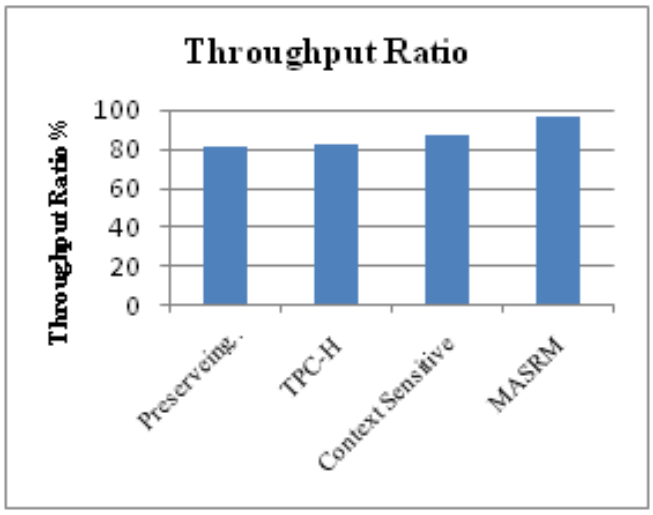
Graph 1: Comparison of integrity management efficiency

Graph 1, shows the efficiency of integrity management produced by different methods and it shows clearly that the proposed method has produced efficient integrity management than other methods.



Graph 2: Comparison of query processing efficiency

Graph 2, shows the comparative analysis of query processing produced by different methods and it shows clearly that the proposed method has produced more efficiency than other methods.



Graph 3: comparison of throughput ratio

The Graph 3, shows the comparison of throughput ratio achieved by different methods and it shows clearly that the proposed method has produced more efficient throughput ratio than other methods.

V. CONCLUSION

To improve the security in data warehouse systems, a multi attribute schematic relational mapping scheme has been

discussed in this paper. The method first preprocesses the input query to identify the set of relational objects present and required. Then the method generates the relational rule set from the metadata and using that the method computes the schematic relational score by performing rule mapping. Finally, based on the result of schematic relations score, the method approves the query processing and returns the result. The proposed method generates efficient throughput in query processing and reduces the rate of intrusion and unauthorized access.

VI. REFERENCES

1. M. Sreedhar Reddy, Prof. M. Rajitha Reddy, Prof R. Viswanath, Prof. G. V. Chalam, Prof. Rajya Laxmi, Prof. Md. Arif Rizwan, A Schematic Technique Using Data type Preserving Encryption to Boost Data Warehouse Security, IJCSI International Journal of Computer Science Issues, Vol. 8, Issue 1, January 2011
2. H. Hirsh. Integrating mulitple sources of information in text classi_cation using whril. In Snowbird Learning Conference, April 2000.
3. M. Barbosa and P. Farshim, "Randomness Reuse: Extensions and Improvements", 12th Institute of Mathematics and its Applications (IMA) Int. Conference on Cryptography and Coding, 2009.
4. J. Kim, Y. Lee, and S. Lee, "DES with any reduced masked rounds is not secure against side-channel attacks", Int. Journal Computers and Mathematics with App., 60, 2010
5. J. Kobielus, "The Forrester Wave: Enterprise Data Warehousing Platforms", Forrester Research Report, Q1, 2009.
6. Oracle Corporation, "Oracle Advanced Security Transparent Data Encryption Best Practices", Oracle White Paper, July 2010.
7. E. Lo, N. Cheng, and W. Hon, "Generating Databases for Query Workloads", Int. Conf. on Very Large DataBases (VLDB), 2010.
8. S. Amritpal, Nitin Umesh "Implementing Log Based Security in Data Warehouse", International Journal of Advanced Computer, 2013
9. Edgar R. Weippl, Security in Data Warehouses, IGI Global, Data Ware housing Design and Advanced Engineering Applications, Ch 015, 2010.
10. Santos, R. J., Bernardino J., Vieria, "Balancing Security and Performance for Enhancing Data Privacy in Data Warehouses", International Joint Conference Of IEEE TrustCom-11/IEEE ICESS-11/FCST -11, 2011.
11. Bockermann, C., Apel, M., and Meier, M., "Learning SQL for Database Intrusion Detection using Context Sensitive Modeling", Int. Conference on Knowledge Discovery and Machine Learning (KDML), 2009.
12. Swapnila S Mirajkar, IISantoshkumar Biradar, Using Secret Sharing Algorithm for Improving Security in Cloud Computing, International Journal of Advanced Research in Computer Science & Technology (IJARCST 2014)
13. K. D. Bowers, A. Juels and A. Oprea, "HAIL: A highavailability and integrity layer for cloud storage", CCS'09: Proc. 16th ACM Conf. on Computer and communications security, 2009, pp. 187-198.
14. G. R. Goodson, J. J. Wylie, G. R. Ganger and M. K. Reiter, "Efficient Byzantine-tolerant erasure-coded storage", DSN'04: Proc. Intl. Conf. on Dependable Systems and Networks, 2004, pp. 1-22.

AUTHOR INFORMATION

Mr. G. Thangaraju is working as Assistant Professor in the Department of Computer Applications, Thanthai Hans Roever College, Perambalur, Tamilnad India. He has 17 years of experience in teaching. He has published many research articles in the National / International Conferences and journals. He is currently pursuing doctor of programme in Computer Science at Karpagam University Coimbatore, Tamilnadu, India. His current area of research interests Data warehousing, Distributed Data Base and software metrics.

Dr. X. Agnise Kala Rani is working as Professor in the Department of Computer Applications at Karpagam University, Coimbatore. She received her Ph. D. in 2011 from Mother Teresa University. She holds the Master of Engineering degree from VMKV University and Master of Computer Applications degree from Madras University. She has several publications including scientific journals and top-tier networking conferences to her credit.