Int. J. Chem. Sci.: 14(4), 2016, 3073-3082
ISSN 0972-768X
www.sadgurupublications.com

COST EFFECTIVE PRIVACY PRESERVING OF INTERMEDIATE DATA SET IN CLOUD STORAGE

V. SARALA* and P. SHANMUGA PRIYA

Department of Information Technology, SCSVMV University, and Working in Department of Computer Science, Meenakshi Collage of Engineering, CHENNAI (T.N.) INDIA

ABSTRACT

Data intensive applications store their valuable intermediate datasets in cloud in order to save the cost of re-computing. This poses a risk on data privacy protection because malicious parties may deduce the private information of the original datasets by analyzing multiple intermediate datasets. This system is implemented based on the least frequent pattern mining algorithm to identify the least frequent table and thereby encrypting it. From the least frequent table the reference attribute between the data tables are found out and a privacy leakage constraint is applied to the intermediate datasets by calculating the severity of the data to identify the sensitive information. As the result in the most frequent table only the privacy sensitive column alone is encrypted. In addition to this, an automatic scheduling algorithm is proposed to maintain a log based tracking for frequent and infrequent usage of data under the time criteria.

Key words: Privacy preserving, Intermediate datasets, Automatic scheduling.

INTRODUCTION

Cloud users can store their valuable intermediate data sets selectively when processing original data sets in a data intensive application in order to curtail the overall expenses by avoiding frequent re-computation to obtain these data sets. Data users often reanalyze results, conduct new analysis, or share some intermediate results with others for collaboration. The secure encryption of privacy preserving of dynamic data sets are used to identify which intermediate data sets need to be encrypted and which do not, so that privacy-preserving cost can be saved.

The technical approaches for preserving the privacy of data sets stored in cloud mainly include encryption and anonymization. On one hand, encrypting all data sets, an effective approach, is widely adopted in current research. However, processing on encrypted

*Author for correspondence; E-mail: saralapurush@yahoo.co.in

data sets efficiently is a challenging task, because most of the applications run on unencrypted data sets. Although homomorphic encryption which theoretically allows performing computation on encrypted data sets, applying algorithms are rather expensive due to their inefficiency. On the other hand, partial information of data sets, e.g. aggregate information, is required to expose to data users in most cloud applications like data mining and analytics. In such cases, data sets are anonymized rather than encrypted to ensure both data utility and privacy preserving. Current privacy-preserving techniques like generalization can withstand most privacy attacks on one single data set, while preserving privacy for multiple data sets is still a challenging problem. Thus, for preserving privacy of multiple data sets, it is promising to anonymize all data sets first and then encrypt them before storing or sharing them in cloud. Usually, the volume of intermediate data sets is huge. Hence, encrypting all intermediate data sets will lead to high overhead and low efficiency when they are frequently accessed or processed. To address this issue, the system proposes to encrypt a part of intermediate data sets rather than all for reducing privacy-preserving cost.

Section 2 discusses about the literature survey emphasizing the research activities in cloud computing and also overviews the drawbacks of existing system. Section 3 presents the privacy preserving of intermediate dataset module description. Section 4 mentions the concluding remarks and future enhancements about the project.

EXPERIMENTAL

Related work

H. Takabi et al.[4] discussed the major challenges of security and privacy issues are discussed. As the cloud computing is a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction. Many organizations aren't comfortable storing their data and applications on systems that reside outside of their on-premise datacenters. This might be the single greatest fear of cloud clients. By migrating workloads to a shared infrastructure, customer's private information faces increased risk of potential unauthorized access and exposure. Cloud service providers must assure their customers and provide a high degree of transparency into their operations and privacy assurance. Privacy-protection mechanisms must be embedded in all security solutions. In a related issue, it's becoming important to know who created a piece of data, who modified it and how, and so on. Provenance information could be used for various purposes such as traceback, auditing, and history-based access control. Balancing between data provenance

and privacy is a significant challenge in clouds where physical perimeters are abandoned. It discusses about the Authentication and Identity Management and Access Control Needs.

H. Lin et al.[3] proposed a general encryption schemes to protect data confidentiality, but also limit the functionality of the storage system because a few operations are supported over encrypted data. Constructing a secure storage system that supports multiple functions is challenging when the storage system is distributed and has no central authority. This system proposed a threshold proxy re-encryption scheme and integrates it with a decentralized erasure code such that a secure distributed storage system is formulated. The distributed storage system not only supports secure and robust data storage and retrieval, but also lets a user forward his data in the storage servers to another user without retrieving the data back. The main technical contribution is that the proxy re-encryption scheme supports encoding operations over encrypted messages as well as forwarding operations over encoded and encrypted messages.

D. Yuan et al.[6] proposed many of scientific workflows are data intensive large volumes of intermediate datasets are generated during their execution. Some valuable intermediate datasets need to be stored for sharing or reuse. Traditionally, they are selectively stored according to the system storage capacity, determined manually. The system builds an intermediate data dependency graph (IDG) from the data provenances in scientific workflows. With the IDG, deleted intermediate datasets can be regenerated, and as such they developed a novel algorithm that can find a minimum cost storage strategy for the intermediate datasets in scientific cloud workflow systems. The strategy achieves the best trade-off of computation cost and storage cost by automatically storing the most appropriate intermediate datasets in the cloud storage. This strategy can be utilized on demand as a minimum cost benchmark for all other intermediate dataset storage strategies in the cloud. Then they decided whether an intermediate dataset should be stored or deleted in order to reduce the system cost. However, the cloud computing environment is very dynamic, and the usages of intermediate datasets may change from time to time. The cost transitive tournament shortest path (CTT-SP) based algorithm can find the minimum cost storage strategy of the intermediate datasets on demand in scientific cloud workflow systems.

Xuyun Zhang et al.[1] proposed a novel approach to identify, which intermediate data sets need to be encrypted while others do not, in order to satisfy privacy requirements given by data holders. A tree structure is modeled from generation relationships of intermediate data sets to analyze privacy propagation of data sets. As quantifying joint privacy leakage of multiple data sets efficiently is challenging, an upper bound constraint is exploited to confine the privacy disclosure. Based on such a constraint, the problem of saving privacy-

preserving cost as a constrained optimization problem is modeled. This problem is then divided into a series of subproblems by decomposing privacy leakage constraints. Finally, a practical heuristic algorithm is designed accordingly to identify the data sets that need to be encrypted. This approach integrates anonymization with encryption to achieve privacy preserving of multiple data sets.

Proposed work

This system is designed to identify only the important and critical intermediate datasets that needs to be encrypted for security purposes hence reducing encryption/ decryption cost and thus maintaining data privacy. It is based on identifying the least frequent table using least frequent pattern mining algorithm and thereby encrypting it by advanced encryption algorithm. From the least frequent table, the reference attribute between the data tables are found out and imposing a privacy leakage constraint to it in order to identify the sensitive information.

For each constraint, the maximal possible value for any of these values is an upper bound and may recover privacy-sensitive partial column level encryption. Hence a column wise encryption to the unencrypted table of the intermediate datasets is proposed. Additional feature of encrypting on the basis of reference attribute between the data tables are achieved to reduce the cost complexity when accessing the data. An automatic scheduling strategy is involved to maintain a log report of the frequent and infrequent usage of intermediate dataset under time conditions as the data in cloud are dynamic in nature. Based on the frequency of accessing, the tables are scheduled according to it and segregated on the least and most frequent table. Therefore this process is repeated to handle the data in cloud in a dynamic manner. When scheduling is done to the datasets the tables are modified and updated to the current situation to handle the dynamic nature of cloud.

System architecture

The data owner can store valuable intermediate data sets selectively when processing original data sets in data intensive applications, in order to curtail the overall expenses by avoiding frequent re-computation to obtain these data sets. Such scenarios are quite common because data users often reanalyze results, conduct new analysis on intermediate data sets, or share some intermediate results with others for collaboration. Usually, intermediate data sets in cloud are accessed and processed by multiple parties, but rarely controlled by original data set holders. In the proposed system an intermediate datasets are created for a Government application where all the people related information is present. When an original dataset is being processed, the intermediate datasets are created such as General

Table, Industry Table, Location Table and Personal table. When these intermediate datasets are collected together by an adversary it can menace the privacy-sensitive information from them, bringing considerable economic loss. Therefore an inference analysis can be made from these datasets.

In order to avoid an inference analysis from these intermediate datasets, the system uses a Least Frequent Pattern Matching algorithm to identify the least frequent tables. The reason for identifying the least frequent table is due to less encryption/decryption computational cost. As a result, the least frequent table will have least frequency of access to the intermediate datasets and therefore it incurs less computation cost rather than the most frequent table. Therefore the least frequent tables will be encrypted using Advanced Encryption Standard algorithm.

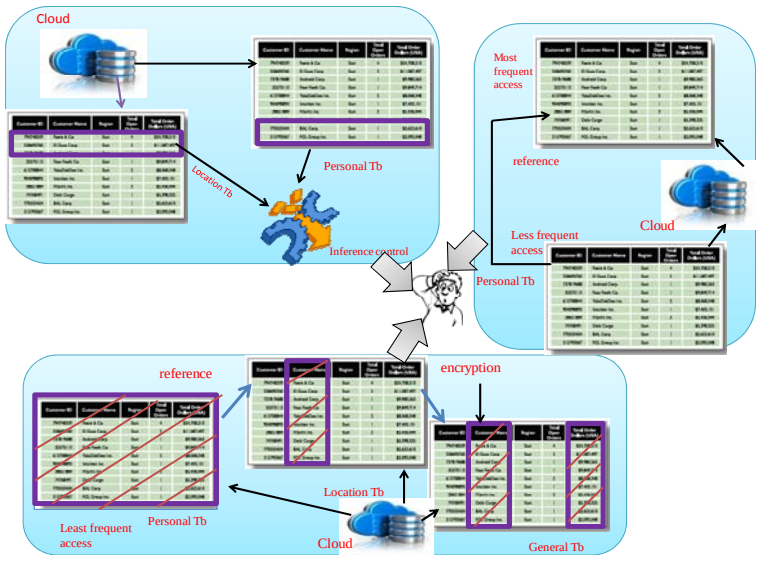

Fig. 1: Architecture of Privacy preserving Intermediate Datasets

There would be many intermediate datasets connected via an entity relationship from the least frequent table. All those tables have to be secure from revealing of privacy sensitive information. An inference analysis must not be made even from the most frequent tables. In order to achieve this only the privacy sensitive information alone have to be encrypted and the rest, leaving unencrypted to reduce the computation cost of encryption/decryption process. A privacy leakage constraint is applied to the intermediate datasets by calculating the severity and global usage of the data to the particular column. When the severity and global usage values exceeds the threshold value 4 then those data are the high sensitive privacy data and the remaining are the low sensitive data. Therefore the high sensitive

column alone is encrypted rather than encrypting all the datasets in the particular table. The encryption is done by the Advanced Encryption Standard algorithm (AES) as the computation cost for encryption and decryption is faster than the other symmetric encryption algorithm. As the result in the most frequent table only the privacy sensitive column alone is encrypted leaving the remaining unchanged in order to reduce the computation cost of the encryption process. Thus an adversary cannot make an inference analysis from any of the intermediate datasets and provides atmost security to the system.

Algorithm description

In this section the representative pattern frequent mining algorithm and advanced encryption standard is used to find the least and most frequent data and for the encryption process respectively.

Representative pattern frequent mining algorithm

Frequent pattern tree (FP-tree) structure, which is an extended prefix-tree structure for storing compressed, crucial information about frequent patterns, and develop an efficient FP-tree- based mining method, FP-growth, for mining the complete set of frequent patterns by pattern fragment growth. Efficiency of mining is achieved with three techniques:

(i) A large database is compressed into a highly condensed, much smaller data structure, which avoids costly, repeated database scans

(ii) FP-tree-based mining adopts a pattern fragment growth method to avoid the costly generation of a large number of candidate sets

(iii) A partitioning-based, divide-and-conquer method is used to decompose the mining task into a set of smaller tasks for mining confined patterns in conditional databases, which dramatically reduces the search space.

Two steps:

1. Scan the transaction DB for the first time, find frequent items (single item patterns) and order them into a list L in frequency descending order.

   e.g., L={f:4, c:4, a:3, b:3, m:3, p:3}.

   In the format of (item-name, support)

2. For each transaction, order its frequent items according to the order in L; Scan DB the second time, construct FP-tree by putting each frequency ordered transaction onto it.

Advanced Encryption Standard Algorithm (AES)

- Encryption on the basis of reference attribute between the data tables are identified.
- Impose a Privacy Leakage constraint to quantify the privacy data.
- Perform column level encryption to the privacy sensitive information alone to the intermediate datasets Advanced Encryption Standard algorithm.

AES is a non-Feistel cipher that encrypts and decrypts a data block of 128 bits. It uses 10, 12, or 14 rounds. The key size, which can be 128, 192, or 256 bits, depends on the number of rounds.

To provide security, AES uses four types of transformations: substitution, permutation, mixing, and key-adding.

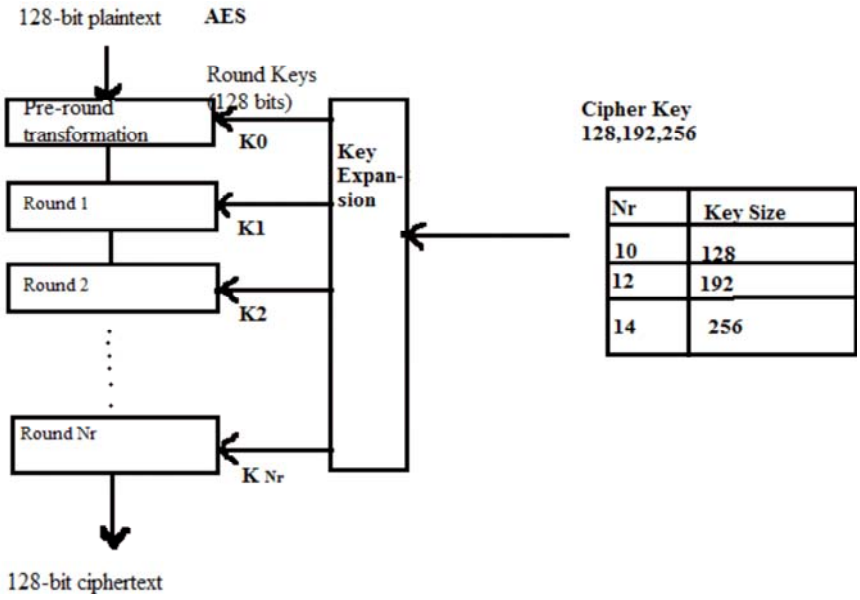

Fig. 2: General design of AES Encryption Cipher

- Processes data as 4 groups of 4 bytes (state).
- It has 9/11/13 rounds in which state undergoes:
  - Byte substitution (1 S-box used on every byte)
  - Shift rows (permute bytes between groups/columns)

  - Mix columns (subs using matrix multiply of groups)
  - Add round key (XOR state with key material)
- Initial XOR key material & incomplete last round.

All operations can be combined into XOR and table lookups - hence very fast & efficient

Byte substitution

- A simple substitution of each byte.
- Uses one table of 16x16 bytes containing a permutation of all 256 8-bit values.
- Each byte of state is replaced by byte in row (left 4-bits) & column (right 4-bits).
  - Eg. byte {95} is replaced by row 9 col 5 byte
  - Which is the value {2A}
- S-box is constructed using a defined transformation of the values in GF(2^8).

Shift rows

- A circular byte shift in each row.
  - 1^{st} row is unchanged
  - 2^{nd} row does 1 byte circular shift to left
  - 3rd row does 2 byte circular shift to left
  - 4th row does 3 byte circular shift to left
- Decryption does the shifts to right.
- Since state is processed by columns, this step permutes bytes between the columns.

Mixcolumns

- Each column is processed separately.
- Each byte is replaced by a value dependent on all 4 bytes in the column.
- Effectively a matrix multiplication in GF(2^8) using prime poly $m(x) = x^8+x^4+x^3+x+1$.

Add round key

- XOR state with 128-bits of the round key.
- Again processed by column (though effectively a series of byte operations).
- Inverse for decryption is identical since XOR is own inverse, just with correct round key.

AES Key expansion

- Takes 128-bit (16-byte) key and expands into array of 44/52/60 32-bit words.
- Start by copying the key into first 4 words.
- Then loop creating words that depend on values in previous & 4 places back.
  - In 3 of 4 cases just XOR these together.
  - Every 4^{th} has S-box + rotate + XOR constant of previous before XOR together.
- Designed to resist known attacks.

RESULTS AND DISCUSSION

The privacy preserving cost is reduced when compared to encrypt all the intermediate datasets. A part of intermediate datasets are encrypted to reduce the frequent computation of encryption/decryption process. Therefore the computation cost of encryption and decryption process is reduced.

CONCLUSION

The secure encryption of privacy preserving intermediate datasets enables to encrypt a part of intermediate datasets rather than encrypting all the datasets in order to reduce the privacy preserving cost. The least frequent table is encrypted and the reference attribute between the data tables are identified to impose a privacy leakage constraint. This identifies the privacy sensitive information and therefore column level encryption is done to the intermediate datasets to prevent the revealing of privacy sensitive information. Therefore the privacy preserving cost for the encryption/decryption process is reduced. With the contributions of this paper, we are planning to further preserve privacy and cost optimization of datasets that are accessible through cloud by considering many other factors such time span of usage, availability of servers and so on.

REFERENCES

1. X. Zhang, C. Liu, S. Nepal and S. Pandey, A Privacy Leakage Upper Bound Constraint-Based Approach for Cost-Effective Privacy Preserving of Intermediate Data Sets in Cloud, IEEE Transactions on Distributed Systems, **24(6)** (2013).

2. W. Du, Z. Teng and Z. Zhu, Privacy-Maxent: Integrating Background Knowledge in Privacy Quantification, Proc. ACM Int'l Conf. Management of Data (SIGMOD'08) (2008).

3. H. Lin and W. Tzeng, A Secure Erasure Code-Based Cloud Storage System with Secure Data Forwarding, IEEE Trans. Distributed Systems, **23(6)**, 995-1003 (2012).

4. H. Takabi, J. B. D. Joshi and G. Ahn, Security and Privacy Challenges in Cloud Computing, IEEE Security & Privacy, Nov./Dec., 24-31 (2010).

5. G. Wang, Z. Zutao, D. Wenliang and T. Zhouxuan, Inference Analysis in Privacy-Preserving Data Re-Publishing, Proc. Eighth IEEE Int'l Conf. Data Mining (ICDM '08), 1079-1084 (2008).

6. D. Yuan, Y. Yang, X. Liu and J. Chen, On-Demand Minimum Cost Benchmarking for Intermediate Data Set Storage in Scientific Cloud Workflow Systems, J. Parallel Computing, **71(2)**, 316-332 (2011).

Accepted : 31.10.2016