COST-EFFECTIVE PRIVACY PRESERVING OF INTERMEDIATE DATA SETS IN CLOUD

M. Manikandan, V. Meena, J. DineshPriyadarshan
Kumaraguru College of Technology,
Saravanampatty, Coimbatore - 641049, India

Abstract- **A massive computation power and storage capacity which enable users to deploy computation and data-intensive applications without infrastructure investment is being provided by cloud computing. Applications such as, a large volume of intermediate data sets will be generated, and often stored to save the cost of recomputing them. The privacy preserving of intermediate data sets becomes a challenging problem because may recover privacy-sensitive information by analyzing multiple intermediate data sets. Encrypting all data sets in cloud is widely adopted in existing approaches to address this challenge. But encrypting all intermediate data sets are neither efficient nor cost-effective because it is very time consuming and costly for data-intensive applications to encrypt/decrypt data sets frequently while performing any operation on them. Here a novel upper bound privacy leakage constraint-based approach is used to identify which intermediate data sets need to be encrypted and which do not, so that privacy-preserving cost can be saved while the privacy requirements of data holders can still be satisfied. Evaluation results demonstrate that the privacy-preserving cost of intermediate data sets can be significantly reduced with our approach over existing ones where all data sets are encrypted**

I.INTRODUCTION

Preserving the privacy of intermediate datasets becomes a challenging problem because adversaries may recover privacy-sensitive information by analyzing multiple intermediate datasets. Encrypting ALL datasets in cloud is widely adopted in existing approaches to address this challenge. But we argue that encrypting all intermediate datasets are neither efficient nor cost-effective because it is very time consuming and costly for data-intensive applications to en/decrypt datasets frequently while performing any operation on them.

Although the techniques used by spammers vary constantly, there is still one enduring feature: spams with identical or similar content are sent in large quantities and successively. Since only a small amount of e-mail users will order products or visit websites advertised in spams, spammers have no choice but to send a great quantity of spams to make profits. It means that even with developing and employing unexpected new tricks, spammers still have to send out large quantities of identical or similar spams simultaneously and in succession. This specific feature of spams can be designated as the near-duplicate phenomenon, which is a significant key in the spam detection problem.

Existing technical approaches for preserving the privacy of datasets stored in cloud mainly include encryption and anonymization. On one hand, encrypting all datasets, a straightforward and effective approach, is widely adopted in current research. However, processing on encrypted datasets efficiently is quite a challenging task, because most existing applications only run on unencrypted datasets.

The privacy concerns caused by retaining intermediate datasets in cloud are important but they are paid little attention. A motivating scenario is illustrated where an on-line health service provider, e.g., Microsoft Health Vault has moved data storage into cloud for economical benefits. Original datasets are encrypted for confidentiality. Data users like governments or research centres access or process part of original datasets after anonymization. Intermediate datasets generated during data access or process are retained for data reuse and cost saving

II. LITERATURE REVIEW

Masaru Takesue's, "Cloud Computing and Emerging It Platforms: Vision, Hype, and Reality for Delivering Computing as the Fifth Utility", 2009 deals with spam detection by use of user's interests. The author describes that if user wants to receive email only in his/her interests, the spam email can be eliminated easily. The technique is that Fingerprints (FPs) of about k portions of each spam's content are stored in the email filter and examine the polymorphic spams devised with intent to thwart the detection. For a smaller memory size of the filter, we exploit two Bloom filters merged into a single one to reduce cache miss to replace the least recently matched spams by recently matched ones. We use as the metrics the number N_t (< k) of FPs in the filter matching with those of an incoming email, but also of the N_t FPs, the greatest number N_d of FPs stored for a single spam. We plot spams and legitimate emails in the N_d - N_t space and detect spams by a piecewise linear function. The experiments with about 4,000 real world emails show that this filter achieves the false negative rate of about 0.36 with no false positive.

Masaru Takesue's next paper on "Cascaded Simple Filters for Accurate and Lightweight Email-Spam Detection", 2011 is based on multiple levels of filters. The author states that accurate spam filters, such as the Bayesian filter, need a large cost for off-line training (or learning) based on the analysis of a large corpus of email. This paper presents cascaded simple, i.e., rule-based, filters for accurate and lightweight detection of email spam. We cascade three filters that classify email based on respectively the fingerprints of message bodies, the white and black lists of email addresses in the From header, and the words specific to spam and legitimate email in the Subject header. Proposed filters need no training, but collect by themselves the information above when they are working, and especially when the user notifies them of their false negative decision (classifying spam as legitimate). Experiment with about 20,000 real world emails that the cascaded simple filters achieve the false negative rate of about 0.025 with no false positive (deciding legal email as spam) and the high performance of about 90 emails per second.

Mehrnoush Famil Saeedian and Humid Beigy's, "On-Demand Minimum Cost Benchmarking for Intermediate Data Set Storage in Scientific Cloud Workflow Systems", 2012 is based on clustering of emails to filter spam emails. The authors proposed a new dynamic weighted voting method based on the combination of clustering and weighted voting, and apply it to the task of spam filtering. In order to classify a new sample, it first compares with all cluster centroid and its similarity to each cluster is identified; Classifiers in the vicinity of the input sample obtain greater weight for the final decision of the ensemble. The evaluation shows that the algorithm outperforms pure SVM. The motivation for using this clustering and weighting methods for spam filtering is that Email is not uniform, but rather consists of messages on different topics and in different genres. This suggests that a classifier which works on a local level can achieve good result on this data. Most of the DWV methods kept training set or validation set and compute nearest neighbors, but algorithm do not keep training data here and just keep cluster centroid vectors, the other improvement is that computations are less than finding nearest neighbors.

Minoru Sasaki and Hiroyuki Shannon's, "Security and Privacy Challenges in Cloud Computing Environments", 2010 is based traditional content analysis but uses clustering to classify spam and non-spam mails. This method computes disjoint clusters automatically using a spherical k-means algorithm for all spam/non-spam mails and obtains centroid vectors of the clusters for extracting the cluster description. For each centroid vectors, the label ('spam' or 'non-spam') is assigned by calculating the number of spam email in the cluster. When new mail arrives, the cosine similarity between the new mail vector and centroid vector is calculated. Finally, the label of the most relevant cluster is assigned to the new mail. By using this method, we can extract many kinds of topics in spam/non-spam email and detect the spam email efficiently. In this paper, we de-scribe the proposed spam detection system and show the result of our experiments using the Ling-Spam test collection. This system provides the high-performance for both spam and non-spam messages. The spam precision is more than about 90% and the non-spam precision is more than 96% for all collections tested with higher accuracy.

III. EXISTING SYSTEM

Existing technical approaches for preserving the privacy of datasets stored in cloud mainly include encryption and anonymization. On one hand, encrypting all datasets, a straightforward and effective approach, is widely adopted in current research. However, processing on encrypted datasets efficiently is quite a challenging task, because most existing applications only run on unencrypted datasets. However, preserving the privacy of intermediate datasets becomes a challenging problem because adversaries may recover privacy-sensitive information by analyzing multiple intermediate datasets. Encrypting ALL datasets in cloud is widely adopted in existing approaches to address this challenge. But we argue that encrypting all intermediate datasets are neither efficient nor cost-effective because it is very time consuming and costly for data-intensive applications to en/decrypt datasets frequently while performing any operation on them.

3.1 PROPOSED SYSTEM

In this paper, we propose a novel approach to identify which intermediate datasets 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 datasets to analyze privacy propagation of datasets. As quantifying joint privacy leakage of multiple datasets efficiently is challenging, we exploit an upper-bound constraint to confine privacy disclosure. Based on such a constraint, we model the problem of saving privacy-preserving cost as a con-strained optimization problem. This problem is then divided into a series of sub-problems by decomposing privacy leakage constraints. Finally, we design a practical heuristic algorithm accordingly to identify the datasets that need to be encrypted. Experimental results on real-world and extensive datasets demonstrate that privacy-preserving cost of intermediate datasets can be significantly reduced with our approach over existing ones where all datasets are encrypted.

IV. MODULES

Privacy leakage is breaking the project into different smaller units. It helps in debugging of modules involved and also enables code reusability. This encourages rapid development, implementation and maintenance.

4.1.1 Data Storage Privacy Module.
4.1.2 Privacy Preserving Module.
4.1.3 Intermediate Dataset Module.
4.1.4 Privacy UpperBoundModule.

4.1.1 Data Storage Privacy Module:

The privacy concerns caused by retaining intermediate datasets in cloud are important but they are paid little attention. A motivating scenario is illustrated where an on-line health service provider, e.g., Microsoft Health Vault has moved data storage into cloud for economical benefits. Original datasets are encrypted for confidentiality. Data users like governments or research centres access or process part of original datasets after anonymization. Intermediate datasets generated during data access or process are retained for data reuse and cost saving. We proposed an approach that combines encryption and data fragmentation to achieve privacy protection for distributed data storage with encrypting only part of datasets.

4.1.2 Privacy Preserving Module:

Privacy-preserving techniques like generalization can with-stand most privacy attacks on one single dataset, while preserving privacy for multiple datasets is still a challenging problem. Thus, for preserving privacy of multiple datasets, it is promising to anonymize all datasets first and then encrypt them before storing or sharing them in cloud. Privacy-preserving cost of intermediate datasets stems from frequent en/decryption with charged cloud services.

4.1.3 Intermediate Dataset Module:

An intermediate dataset is assumed to have been anonym zed to satisfy certain privacy requirements. However, putting multiple datasets together may still invoke a high risk of revealing privacy-sensitive information, resulting in violating the privacy requirements. Data provenance is employed to manage intermediate datasets in our research. Provenance is commonly defined as the origin, source or history of derivation of some objects and data, which can be reckoned as the information upon how data was generated. Re-produce ability of data provenance can help to regenerate a dataset from its nearest existing predecessor datasets rather than from scratch

4.1.4 Privacy Upper Bound Module:

Privacy quantification of a single data-set is stated. We point out the challenge of privacy quantification of multiple datasets and then derive a privacy leakage upper-bound constraint correspondingly. We propose an upper-bound constraint based approach to select the necessary subset of intermediate datasets that needs to be encrypted for minimizing privacy-preserving cost. The privacy leakage upper-bound constraint is decomposed layer by layer.

V. SYSTEM TESTING

Implementation is the process of converting a new or revised system design into an operational one. The implementation is the final and important phase. It involves ser training, system testing and successfully running of developed proposed system. The user tests the developed system and changes are made according to their needs. The testing phase involves the testing of developed system using various kinds of data. An elaborate testing of data is prepared and the system is tested using that test data. The corrections are also noted for future use. The users are trained to operate the developed system. Both the hardware and software securities are made to run the developed system successfully in future.

Implementation is the process of converting a new or revised system design in to an operational one. Education of user should really have taken place much earlier in the project when they were being involved in the investigation and design work. Training has to be given to the user regarding the new system. Once the user has been trained, the system can be tested hardware and software securities are to run the developed system successfully in the future.

The first phase of software project is to gather requirements. Gathering software requirements begins as a creative brainstorming process in which the goal is to develop an idea for a new product that no other software vendor has thought. New software product ideas normally materialize as a result of analyzing market data and interviewing customers about their product needs.

The main function of the requirements gathering phase is to take an abstract idea that fills a particular need or that solves a particular problem and create a real world project with a particular set of objectives, a budget, a timeline and a team.

VI. CONCLUSION

The **"A Privacy Leakage Upper Bound Constraint-Based Approach for Cost-Effective Privacy Preserving of Intermediate Data Sets in Cloud"** has been developed to satisfy all proposed requirements. In this paper, we have proposed an approach that identifies which part of intermediate data sets needs to be encrypted while the rest does not, in order to save the privacy preserving cost. A tree structure has been modeled from

the generation relationships of intermediate data sets to analyze privacy propagation among data sets. We have modeled the problem of saving privacy-preserving cost as a constrained optimization problem which is addressed by decomposing the privacy leakage constraints.

The software executes successfully by fulfilling the objectives of the project. This system can be made required with minor modifications. The invention can be implemented in digital electronic circuitry, or in computer hardware, firmware, software, or in combinations of them. Apparatus of the invention can be implemented in a computer program product tangibly embodied in a machine-readable storage device for execution by a programmable processor; and method steps of the invention can be performed by a programmable processor executing a program of instructions to perform functions of the invention by operating on input data and generating output.

REFERENCES

[1] M. Armbrust, A. Fox, R. Griffith, A.D. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, and M. Zaharia, "A View of Cloud Computing," Comm. ACM, vol. 53,no. 4, pp. 50-58, 2010.

[2] R. Buyya, C.S. Yeo, S. Venugopal, J. Broberg, and I. Brandic, "Cloud Computing and Emerging It Platforms: Vision, Hype, and Reality for Delivering Computing as the Fifth Utility," Future Generation Computer Systems, vol. 25, no. 6, pp. 599-616, 2009.

[3] L. Wang, J. Zhan, W. Shi, and Y. Liang, "In Cloud, Can Scientific Communities Benefit from the Economies of Scale?," IEEE Trans. Parallel and Distributed Systems, vol. 23, no. 2, pp. 296-303, Feb. 2012.

[4] H. Takabi, J.B.D. Joshi, and G. Ahn, "Security and Privacy Challenges in Cloud Computing Environments," IEEE Security & Privacy, vol. 8, no. 6, pp. 24-31, Nov./Dec. 2010.

[5] D. Zissis and D. Lekkas, "Addressing Cloud Computing Security Issues," Future Generation Computer Systems, vol. 28, no. 3, pp. 583- 592, 2011.

[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 Distributed Computing, vol. 71, no. 2, pp. 316-332, 2011.

[7] S.Y. Ko, I. Hoque, B. Cho, and I. Gupta, "Making Cloud Intermediate Data Fault-Tolerant," Proc. First ACM Symp. Cloud Computing (SoCC '10), pp. 181-192, 2010.

[8] H. Lin and W. Tzeng, "A Secure Erasure Code-Based Cloud Storage System with Secure Data Forwarding," IEEE Trans. Parallel and Distributed Systems, vol. 23, no. 6, pp. 995-1003, June 2012.

[9] N. Cao, C. Wang, M. Li, K. Ren, and W. Lou, "Privacy-Preserving Multi-Keyword Ranked Search over Encrypted Cloud Data," Proc. IEEE INFOCOM '11, pp. 829-837, 2011.

[10] M. Li, S. Yu, N. Cao, and W. Lou, "Authorized Private Keyword Search over Encrypted Data in Cloud Computing," Proc. 31st Int'l Conf. Distributed Computing Systems (ICDCS '11), pp. 383-392, 2011.

Books Referred:

1. Professional Java Network Programming
2. Java Complete Reference
3. Data Communications and Networking, by Behrouz A Frozen.
4. Computer Networking: A Top-Down Approach, by James F. Kurose.
5. Basics of Cloud Computing by Scheldt.