ISSN (Online): 2349-7084
GLOBAL IMPACT FACTOR 0.238
ISRA JIF 0.351

INTERNATIONAL JOURNAL OF COMPUTER ENGINEERING IN RESEARCH TRENDS
VOLUME 1, ISSUE 6, DECEMBER 2014, PP 447-452

Privacy Preserving Transitional Data Set Using Heuristic Approach

[1] K.SANDHYA, [2] K.SUMALATHA

[1] M.Tech Research Scholar, Priyadarshini Institute of Technology and Science for Women
[2] Assistant Professor, Priyadarshini Institute of Technology and Science for Women

Abstract: In this paper, we propose a upper-bound privacy leakage constraint based approach to identify which intermediate datasets 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. To identify and encrypt all functionally encrypt able data, sensitive data that can be encrypted without limiting the functionality of the application on the cloud. 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 encrypt/decrypt datasets frequently while performing any operation on them.

Index Terms- Data Storage Privacy, Encryption and Decryption, Privacy Preserving, Intermediate Dataset, Privacy Upper Bound, Economics of scale.

I. INTRODUCTION

Cloud Computing refers to both the applications delivered as services over the Internet and the hardware and systems software in the datacenters that provide those services. 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.

A powerful underlying and enabling concept is computing through service-oriented architectures (SOA) – delivery of an integrated and orchestrated suite of functions to an end-user through composition of both loosely and tightly coupled functions, or services – often network based. Related concepts are component-based system engineering, orchestration of different services through workflows, and virtualization. The key to a SOA framework that supports workflows is componentization of its services, an ability to support a range of couplings among workflow building blocks, fault tolerance in its data- and process-aware service-based delivery, and an ability to audit processes, data and results, i.e., collect and use provenance information.

Cloud computing [1] should provide massive computation power and storage space for the users. The users can use these resources in pay as you go manner [2], instead of buying the required hard-disk or processors for their business. Because of this the business persons can reduce their investment cost and concentrate on their business development. Due to this so many users are very interested to use this cloud computing technology. But some of the users are very hesitant to store their data into the cloud according to security. So to provide security [3] for the data we are encrypting the entire data and allowing only authenticated users. At the time of executing any data intensive applications some intermediate datasets [4] or resultant data sets are generated, these are stored in the cloud for future purpose, instead of re-computing each and every whenever they need.

For some data mining or analysis areas there is a need of revealing some aggregate information to the public.

Publishing some data by satisfying the privacy requirements of data holders can be done by Anonymization [6]. Anonymization is one of the privacy techniques like encryption. For a single dataset there is privacy, but multiple datasets are not secure. So, in our proposed system to provide privacy for multiple datasets we are using both Anonymization and encryption technologies. In the proposed system constructing a Sensitive Intermediate Dataset Tree(SIT) based on generation relationship among the intermediate datasets and finding privacy leakage for each and every intermediate dataset and then by using heuristic method we can identify which intermediate dataset we need to encrypt and find the minimum privacy preserving cost.
If any adversary should access these datasets then there is a chance of analyzing the information, so we need to provide privacy for these datasets. But the computations are performed only on the readable data, so to perform any operations each and every time we need to decrypt the data set, perform the computation and then encrypt and store the dataset. For this purpose we need some extra storage space and also it is time consuming. There is a technology homomorphic encryption [5] by using theoretically proved not implemented practically.

II. RELATED WORK

The importance of retaining intermediate data sets in cloud has been widely recognized [6], but the research on privacy issues incurred by such data sets just commences. Davidson et al. [2] studied the privacy issues in workflow provenance, and proposed to achieve module privacy preserving and high utility of provenance information via carefully hiding a subset of intermediate data. This general idea is similar to ours, yet our research mainly focuses on data privacy preserving from an economical cost perspective while theirs concentrates majorly on functionality privacy of workflow modules rather than data privacy. Our research also differs from theirs in several aspects such as data hiding techniques, privacy quantification and cost models. But, our approach can be complementarily used for selection of hidden data items in their research if economical cost is considered.

However, processing on encrypted data sets efficiently is quite a challenging task, because most existing applications only run on unencrypted data sets. Although recent progress has been made in homomorphic encryption which theoretically allows performing computation on encrypted data sets, applying current algorithms are rather expensive due to their inefficiency [4].

This work provides the various approaches for privacy preserving in cloud computing. Encryption is the technique to preserve the privacy of data. Storing data in a third arty's cloud system causes serious concern on data confidentiality. In order to provide strong confidentiality for messages in storage servers, user can encrypt data by cryptographic method. Encrypting all the data sets, a straight forward and effective approach is widely adopted in [1], [2], [3].

This approach differs from the traditional cryptography technology for preserving user's privacy as it gets rid of key management and thus it stands simple and flexible. While anonymising is easier, the attributes that has to be made anonymous varies and it depends on the cloud service provider. This approach will be suitable only for limited number of services. Thus, the method has to be bettered by automating the anonymisation.
We briefly review the research on privacy protection and consider the economical aspect of privacy preserving, adhering to the pay-as-you-go feature of cloud computing. Once we identify the data to be encrypted, we must choose how many keys to use for encryption, and the granularity of encryption. In the simplest case, we can encrypt all such data using a single key, and share the key with all users of the service. Unfortunately, this has the problem that a malicious or compromised cloud could obtain access to the encryption key, e.g. by posing as a legitimate user, or by compromising or clouding with an existing user. In these cases, confidentiality of the entire dataset would be compromised.

Airavat [6] is a Map Reduce-based system which provides strong security and privacy guarantees for distributed computations on sensitive data. Airavat is a novel integration of mandatory access control and differential privacy. It enables many privacy-preserving Map Reduce computations without the need to audit untrusted code. Its objective is to prevent information leakage beyond the data provider's policy.

Silverline [7] is a set of tools that automatically identifies all functionally encryptable data in a cloud application, assigns encryption keys to specific data subsets to minimize key management complexity while ensuring robustness to key compromise, and provides transparent data access at the user device while preventing key compromise even from malicious clouds. Silverline provides a substantial first step towards simplifying the complex process of incorporating data confidentiality into these storage-intensive cloud applications. Its aim is to improve the confidentiality of application data stored on third-party computing clouds. But there are several disadvantages of Silverline. Sedic [8] provides a solution to the privacy threat that is to split a task, keeping the computation on the private data within an organization's private cloud while moving the rest to the public commercial cloud. Sedic leverages the special features of MapReduce to automatically partition a computing job according to the security levels of the data it works on, and arrange the computation across a hybrid cloud.

Encryption and fragmentation approach [9] couples encryption together with data fragmentation. Encryption will be applied only when explicitly demanded by the privacy requirements. Privacy requirements are enforced by splitting information over two independent database servers in order to break associations of sensitive information and by encrypting information whenever necessary. The information to be protected is first split into different fragments in such a way to break the sensitive associations represented through confidentiality constraints and to minimize the amount of information represented only in encrypted format. The resulting fragments may be stored at the same server or at different servers. Finally, the encryption key is given to the authorized users needing to access the information. Users that do not know the encryption key as well as the storing server(s) are able neither to access sensitive information nor to reconstruct the sensitive associations. But the protection of fragmented data when the information stored in the fragments may change over time is difficult.

III. PROPOSED SYSTEM

In this section we are finding the effective privacy preserving cost of intermediate datasets in the cloud by using the SIT, privacy representation and construction of compressed tree, minimum privacy preserving cost and heuristic method as follows.

1) Process Original data set:

Storage and computation services in cloud are equivalent from an economical perspective because they are charged in proportion to their usage. Thus cloud customers 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 recompilation to obtain these data sets.

The data holder will store the data into cloud after encryption. The original dataset is encrypted for confidentiality. The data users have to register themselves by giving the username and password. Then only they can able to decrypt the data that the data holder has stored in cloud. DES algorithm is used for encryption. Only the authenticated users can process the dataset. Storage and computation services in cloud are equivalent from an economical perspective because they are charged in proportion to their usage. Thus cloud customers 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 recompilation to obtain these data sets.

2) Privacy leakage quantification:

The privacy sensitive information is generally regarded as the association between sensitive data and individuals. Privacy leakage of the intermediate data set is quantified. And a threshold value is given by the data holder. Threshold value should not exceed the maximum privacy leakage of the single data set. If the privacy leakage threshold is minimum more data sets need to be encrypted. If it is maximum more data sets may remain unencrypted. The sum of the privacy leakage of the unencrypted data sets should not exceed the threshold value given by the data holder.

$$PL_s(d^*) \triangleq H(S,Q) - H^*(S,Q)$$

Where $H(S,Q) = \log(|QI| \,.\, |SD|)$ and

$$H^*(S,Q) = -\sum_{q \epsilon QI, s \epsilon SA} p(s,q).\log(p(s,q))$$

3) Privacy Leakage Constraint Decomposition:

The privacy leakage constraint is decomposed into different layers. So there is different threshold value for each layer. The privacy leakage incurred by the unencrypted data set in the layer can never be larger than the threshold value in that layer.

A local encryption solution in the layer is feasible if it satisfies the privacy leakage constraint. A set of feasible solutions exists in a layer which constitutes global solution. A compressed tree is created from layer 1 to H where H is the height of the tree.

The construction is achieved via three steps.

1. The data sets in EDi are compressed into one encrypted node.
2. All offspring data sets of the data sets in UDi are omitted.
3. The data sets in UDi are compressed into one node.

$$\sum_{d \in UDi} PLs(d) \leq \varepsilon i\,, 1 \leq i \leq H$$

The threshold εi,1 ≤ i ≤ H, is calculated by

$$\left\{\begin{array}{l} \epsilon_i = \epsilon_{i-1} - \sum_{d \in UD_{i-1}} PL_s(d) \\ \qquad \epsilon_1 = \epsilon \end{array}\right.$$

4) Cost Calculation:

Cost of storing the intermediate data set is calculated by the size of the intermediate data set, frequency of accessing that data set and the price set up by cloud service vendors. If the frequency of accessing the intermediate data set is larger than more cost will be incurred if the intermediate data set is encrypted.

The privacy preserving cost rate is denoted as

$$CR_{pp} \triangleq \sum_{d_i \epsilon\, D^{enc}} S_i\,.PR.f_i$$

Where Si is the size of the intermediate data set, fi is the frequency of accessing the stored intermediate data set, and PR is the price for encryption and decryption. The cost of privacy preserving should be minimum in order to get the optimal result. Data holder will give privacy requirements that is the privacy leakage threshold allowed by a data holder, the privacy leakage caused by the unencrypted data sets should be under a given threshold.

$$PL_m(D^{une}) \leq \varepsilon, D^{une} \in D.$$

Where PLm(Dune) is the privacy leakage of the multiple data sets and (Dune) is the unencrypted data sets.

5) Cost Effective Solution:

Usually, more than one feasible global encryption solution exists under the PLC, because there are many alternative local solutions in each layer. Further, each intermediate data set has various size and frequency of usage, leading to different overall cost with different solutions. Therefore it is desired to find a feasible solution with minimum privacy-preserving cost under privacy leakage constraints. Heuristic approach is used to reduce privacy-preserving cost. It prefers to encrypt the data sets which incur less cost but disclose more privacy sensitive information. Data sets with higher privacy-preserving cost and lower privacy leakage are expected to remain unencrypted. Thus cost is reduced in this technique instead of encrypting all data sets.

IV.PERFORMANCE EVALUTION AND RESULTS

Data holders store their data into cloud. Only the authenticated users can decrypt and download the data. While processing the data intermediated data sets are generated. Privacy leakage of the intermediate data sets is calculated. Based on the privacy requirement of the data holder intermediate data sets are encrypted selectively. Cost of encrypting the data sets is also calculated. The data set which incurs less cost for encryption and leaks more privacy is selected for encryption and others remain unencrypted. The privacy leakage of the unencrypted data set is lesser than the threshold value given by the data holder. When adversary sees the data set he cannot infer any information from them.

V. CONCLUSIONS

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. Based on such a constraint, we model the problem of saving privacy-preserving cost as a constrained optimization problem.

The privacy preserving cost of intermediate datasets in cloud can be reduced over existing approaches instead of encrypting all the intermediate datasets by encrypting only part of intermediate datasets in our approach by using SIT, compressed tree and heuristic algorithms.

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.

[11] C. Gentry, "Fully Homomorphic Encryption Using Ideal Lattices,"
Proc. 41st Ann. ACM Symp. Theory of Computing (STOC '09), pp. 169-178, 2009.

[12] B.C.M. Fung, K. Wang, and P.S. Yu, "Anonymizing Classification
Data for Privacy Preservation," IEEE Trans. Knowledge and Data Eng., vol. 19, no. 5, pp. 711-725, May 2007.

[13] B.C.M. Fung, K. Wang, R. Chen, and P.S. Yu, "Privacy-Preserving Data Publishing: A Survey of Recent Developments," ACM Computing Survey, vol. 42, no. 4, pp. 1-53, 2010.

[14] X. Zhang, C. Liu, J. Chen, and W. Dou, "An Upper-Bound Control Approach for Cost-Effective Privacy Protection of Intermediate Data Set Storage in Cloud," Proc. Ninth IEEE Int'l Conf. Dependable,
Autonomic and Secure Computing (DASC '11), pp. 518-525, 2011.

[15] I. Roy, S.T.V. Setty, A. Kilzer, V. Shmatikov, and E. Witchel, "Airavat: Security and Privacy for Mapreduce," Proc. Seventh USENIX Conf. Networked Systems Design and Implementation (NSDI'10), p. 20, 2010.

[16] K.P.N. Puttaswamy, C. Kruegel, and B.Y. Zhao, "Silverline:Toward Data Confidentiality in Storage-Intensive Cloud Applications," Proc. Second ACM Symp. Cloud Computing (SoCC '11), 2011.

[17] K. Zhang, X. Zhou, Y. Chen, X. Wang, and Y. Ruan, "Sedic:Privacy-Aware Data Intensive Computing on Hybrid Clouds,"
Proc. 18th ACM Conf. Computer and Comm. Security (CCS '11),pp. 515-526, 2011.

[18] V. Ciriani, S.D.C.D. Vimercati, S. Foresti, S. Jajodia, S. Paraboschi, and P. Samarati, "Combining Fragmentation and Encryption to Protect Privacy in Data Storage," ACM Trans. Information and
System Security, vol. 13, no. 3, pp. 1-33, 2010.

[19] S.B. Davidson, S. Khanna, T. Milo, D. Panigrahi, and S. Roy, "Provenance Views for Module Privacy," Proc. 30th ACM
SIGMOD-SIGACT-SIGART Symp. Principles of Database Systems (PODS '11), pp. 175-186, 2011.

[20] S.B. Davidson, S. Khanna, S. Roy, J. Stoyanovich, V. Tannen, and Y. Chen, "On Provenance and Privacy," Proc. 14th Int'l Conf. Database Theory, pp. 3-10, 2011.

[21] S.B. Davidson, S. Khanna, V. Tannen, S. Roy, Y. Chen, T. Milo, and J.Stoyanovich, “Enabling Privacy in Provenance-Aware Workflow Systems,” Proc. Fifth Biennial Conf. Innovative Data Systems Research (CIDR ’11), pp. 215-218, 2011.

[22] P. Samarati, “Protecting Respondents’ Identities in Microdata Release,” IEEE Trans. Knowledge and Data Eng., vol. 13, no. 6, pp. 1010-1027, Nov. 2001.

[23] A. Machanavajjhala, D. Kifer, J.ehrke, and M. Venkitasubramaniam, “L-Diversity: Privacy Beyond K-Anonymity,” ACM Trans. Knowledge Discovery from Data, vol. 1, no. 1, article 3, 2007.

[24] 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), pp. 1079-1084, 2008.

[25] W. Du, Z. Teng, and Z. Zhu, “Privacy-Maxent: Integrating Background Knowledge in Privacy Quantification,” Proc. ACM
SIGMOD Int’l Conf. Management of Data (SIGMOD ’08), pp. 459-472, 2008.

[26] E.T. Jaynes, “Information Theory and Statistical Mechanics,”Physical Rev., vol. 106, no. 4, pp. 620-630, 1957.

[27] Microsoft HealthVault, http://www.microsoft.com/health/ww/
products/Pages/healthvault.aspx, July 2012.

[28] K.-K. Muniswamy-Reddy, P. Macko, and M. Seltzer, “Provenance for the Cloud,” Proc. Eighth USENIX Conf. File and Storage Technologies (FAST ’10), pp. 197-210, 2010.

[29] Amazon Web Services, “Aws Service Pricing Overview,” http://aws.amazon.com/pricing/, July 2012.

[30] KVM, http://www.linux-kvm.org/page/Main_Page, July 2012.

[31] OpenStack, http://openstack.org/, July 2012.

[32] Hadoop, http://hadoop.apache.org, June 2012.