A Novel Approach For Ensuring Cost Effective Privacy In Cloud

Dr.R.Saravanan
Director,
RVS Educational Trust's Group of Institutions
Dindigul, India

Mrs. R.Sivakami,M.E,(Ph.D)
Associate Professor, IT Department
PSNA College of Engineering and Technology
Dindigul, India

Feby Cherian
M.E Computer and Communication
PSNA College of Engineering and Technology,
Dindigul, India

***Abstract* — Cloud computing helps users to store massive data and allows them to process or retrieve the data whenever required without the aid of any infrastructure investment. But in the cloud, the privacy of the data is a major concern. For processing the data, data is classified into multiple data sets. So in order to maintain the privacy, the common approach used is to encrypt all intermediate data sets and store in the cloud. But this approach will increase the cost and also will consume more time which would be a big challenge. In this paper, we propose an approach which will calculate the sensitive information of all intermediate data sets stored in a queue in the order of the priority of sensitive information. Based on a particular threshold value, user can decide how much is the degree of encryption which will encrypt the most sensitive data sets and keep insensitive data sets without encryption. This will reduce the cost and also save the time without losing the privacy of data.**

Keywords-cloud computing;data privacy; threshold; intermediate data sets

I. INTRODUCTION

Cloud computing refers to delivery of computer services which lets the users to use the software and hardware which are managed in remote locations by the Cloud Service Providers. This helps the cloud customers to save their money by investing in IT servers and thereby concentrate in their own primary businesses.

Security and privacy are the major concerns in cloud computing but they are paid little attention. The price for storing the data in the cloud is also directly proportional to the volume of data. So the cloud users may store intermediate data sets for computation purposes so that they can reduce expenses by computing the same data sets again and again[1]. The storage of these data sets increases the risk of privacy being violated. Multiple vendors are able to access the same data sets. This increases the chances for adversary to get the sensitive information by accessing the multiple intermediate data sets. This might lead to severe economic losses or it might also affect the social status of the data owner.

Data encryption [10],[2] and anonymization [3],[4] techniques having been extensively studied recently, are promising and widely-adopted ways to combat privacy breach and violation on cloud. Encryption, access control, and differential privacy are the mechanisms that are used to protect the data privacy. These are well-known pillars of privacy protection and still have open questions in the context of cloud computing. The data sets that are uploaded into cloud are not only for storage purpose, but also for doing online cloud applications. In such cases, encryption or access control mechanisms alone fail to ensure privacy preservation and data utility exposure[13]. However, most existing security algorithms lack scalability over the data. To address the privacy issues current approaches deserve considerable attention. Still the data sets scattered on cloud probably compromise data privacy if they are not properly managed. The adversaries are able to collect these data sets from cloud and infer certain privacy from them even if each data set individually satisfies a privacy requirement. For preserving privacy of multiple data sets, it is advisable to anonymize all data sets first and then encrypt them before storing them in cloud.

Privacy-preserving techniques like generalization [14] can have most privacy attacks on one single data set, but for preserving privacy for multiple data sets is still a challenging issue. In order to solve the issue, it is recommended to anonymize all data sets first and then encrypt them before storing or sharing them in the cloud.

As the volume of intermediate data sets is large, encrypting all intermediate data is neither efficient nor cost effective method as it is frequently accessed or processed. So we suggest to encrypting data sets which are more sensitive rather than encrypting all data sets for reducing the privacy-preserving cost.

In this paper, we propose an approach to identify the intermediate data sets to encrypt so that any sensitive

information given by data holders will not leak at any cost. The relationships between the data sets are represented as a tree structure model. Using an upper-bound constraint, we define privacy leakage of multiple data sets by decomposing the privacy leakage constraints. Finally, we design an algorithm which identifies the data sets that needs to be encrypted. With experimental results on real world, it is clear that privacy preserving cost of intermediate data sets can be significantly reduced with our approach as compared to the existing techniques where all data sets are encrypted.

We have classified our research into three. First, we illustrate how we can ensure privacy leakage requirements without encrypting all intermediate data sets. Second speaks about the algorithm which identifies the data sets needs to be encrypted for preserving privacy. Third, experiment results demonstrate how privacy-preserving costs get reduced by our approach over existing approaches.

II. RELATED WORK

Nowadays, popular scientific workflows are often deployed in data privacy preservation and privacy protection in cloud computing environments. These privacy and security related issues have been extensively studied in the research area and they have made fruitful progresses with a variety of privacy models and privacy preserving methods. But most of the existing security algorithms lack scalability over the data. In the year of 2007, the concept of cloud computing was proposed [5] and it is deemed to be as the next generation of IT platforms that can deliver computing as a kind of utility.

The economical aspect of privacy preserving is adhering to the pay-as-you-go feature of cloud computing. The security and privacy topics were discussed and comprehensive technical review of security issues were included in the study, in which , integrity, availability, accountability and privacy were identified as the major attributes[19]. In each property, a few security issues are described, followed by corresponding defense solutions. In the end, it was claimed that the study might help shaping the future of research directions in the security and privacy contexts in terms of clouds.

There are numerous security challenges that are enumerated in the topic cloud security. The challenges are related with resource allocation, multi-tenancy, authentication and authorization, system monitoring and logging, computer forensics, virtualization, availability, and cloud standards. The study focused on introducing the Service Level Agreements (SLAs), trust and accountability topics with regard to cloud security. Our goal is to automatically get the right granularity for data encryption that provides the best trade-off between robustness and management complexity. For this we partition the data into subsets, where each subset of data is accessed by the same group of users. Thus, cloud users can store valuable intermediate data sets selectively when processing original data sets in data-intensive applications like medical diagnosis, in order to limit the overall expenses by avoiding frequent re-computation to obtain these data sets. Even though encryption works well for data privacy in these approaches, it is obligatory to encrypt and decrypt data sets frequently in many applications.

Encryption is usually combined with other methods to attain cost reduction, high data handiness and privacy protection. Roy et al. [6] proposed the data privacy problem that was reasoned by Map Reduce and presented a system named Airavat which incorporates mandatory access control with differential privacy. Puttaswamy et al. [7] described a set of tools called Silverline that recognizes all functionally encryptable data and then encrypts them to protect privacy. Zhang et al.[8] suggested a work named Sedic which partitions Map Reduce computing jobs in terms of the security labels of data they work on and then assigns the computation without sensitive data to a public cloud. The sensitivity of data in cloud is required to be labeled in advance to make the above approaches. Ciriani et al.[9] put forward a method that combines encryption and data fragmentation to achieve privacy protection for distributed data storage with encrypting only part of data sets. We follow this theory, but integrating data anonymization and encryption together to fulfill cost-effective privacy preserving. The importance of storing intermediate data sets in cloud has been widely recognized [20], but the privacy issues incurred by such data sets just commences. Davidson et al.[17],[18]discussed the privacy issues in workflow provenance, and proposed to gain module privacy preserving and high utility of provenance information via carefully hiding a subset of intermediate data. This idea is similar to ours, yet our research mainly concentrates on data privacy preserving from an economical cost perspective while theirs focuses mainly on functionality privacy of workflow modules rather than data privacy. Our research also differs in several aspects such as cost models, privacy quantification and data hiding techniques. But our approach can be used for selection of hidden data items in their research if economical cost is considered[15]. The PPDP research community has extensively investigated on privacy-preserving issues and made fruitful progress with a variety of privacy models and preserving methods. Many anonymization techniques like generalization have been used to preserve privacy of data, but these techniques alone fail to resolve the problem of preserving privacy for multiple data sets. Our approach combines anonymization with encryption to achieve privacy preserving of multiple data sets.

III. MOTIVATING EXAMPLE

For example let's consider the scenario, Microsoft Health Vault [12], has moved data storage into cloud for economic benefits. Then the original data sets are encrypted for maintaining confidentiality. Data users like government or pharmaceutical company access or process part of original data sets after anonymization. Intermediate data sets that are

generated during data access or process are stored in cloud database for data reuse and cost saving. Two independently created intermediate data sets in Fig.1 are anonymized

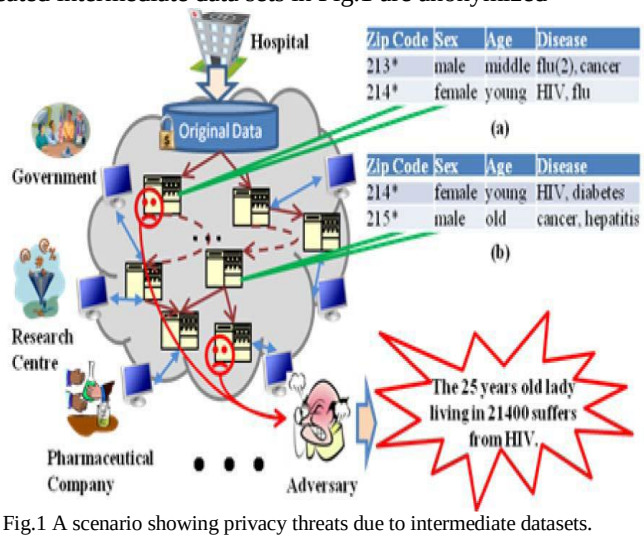

Fig.1 A scenario showing privacy threats due to intermediate datasets.

satisfy 2 diversity that means at least two individuals own the same quasi-identifier and each quasi-identifier corresponds to at least two sensitive values[16]. Knowing that a lady who has 25 years age living in 21400 (corresponding quasi-identifier is (214 * female, young) is in both data sets, an adversary can conclude that this individual suffers from HIV with high confidence if (a) and (b) are collected together. Hiding (a) or (b) by encryption is a promising way to prevent.Such a privacy leakage. Assume (a) and (b) are of the same size, the frequency of accessing (a) is 100 and that of (b) is 1000. We hide (a) to preserve privacy because this can be done in less cost than hiding (b). In all real-world applications, a great number of intermediate data sets are involved. Hence, it is really challenging to identify which data sets should be encrypted to ensure that privacy leakage requirements are satisfied and also by keeping the hiding overheads as low as possible.

IV. PROBLEM ANALYSIS

A. *Sensitive Intermediate Data Set Representation*

Let ds0 be a privacy-sensitive original data set. We use DS = {ds1,ds2,…,dsn} to denote a group of intermediate data sets generated from ds0 where n is the number of intermediate data sets. Directed Acyclic Graph (DAG) is used to capture the topological structure of generation relationships among these data sets. Sensitive Intermediate data set Graph, denoted as SIG (fig.2) is defined as DAG representing the generation relationships of intermediate data sets DS from ds0.

Sensitive Intermediate data set Tree (SIT) is nothing but SIG in a tree structure. The root of the tree is ds0. An SIG or SIT not only denotes the generation relationships of an original data set and its intermediate data sets, but also captures how privacy-sensitive information moves among such data sets. Usually, the privacy-sensitive information in ds0 is spread into its offspring data sets. Hence, an SIG or SIT can be used to analyze privacy disclosure of multiple data sets.

B. *Privacy-Preserving Cost Analysis*

Privacy-preserving cost of intermediate data sets comes from frequent en/decryption with charged cloud services. Cloud service providers have set up various pricing models

Practically, en/decryption requires data storage computation power, and other cloud services. To avoid pricing details and concentrate on the discussion of our core ideas, we combine the prices of several services required by en/decryption into one. This combined price is denoted as PRG. PRG indicates the overhead of en/decryption on per GB data per execution. The term Si represents the size of any data set (dsi). The term Flagi, a dichotomy tag, represents whether dsi is hidden. The term fi signifies the frequency of accessing or processing dsi. If dsi is labeled as hidden, it will be en/decrypted whenever it is processed or accessed. Thus, the larger the fi is, the more cost will be charged if dsi is hidden. The term PrLei is the privacy leakage through dsi, and is computed by PrLes(dsi). Data Sets in DS can be divided into two sets. One is encrypted data sets, denoted as DS_{en}. The second is for unencrypted data sets, denoted as DSun. Then, the equations DSen U DSun = DS and DSen Ω DSun= Ø hold. The privacy-preserving cost rate (Cpp) formula is

$$Cpp\ (DS_{en},DS_{un}))= (\int\ t\text{T}0(Si.PRG.fi.t)dt. \qquad (1)$$

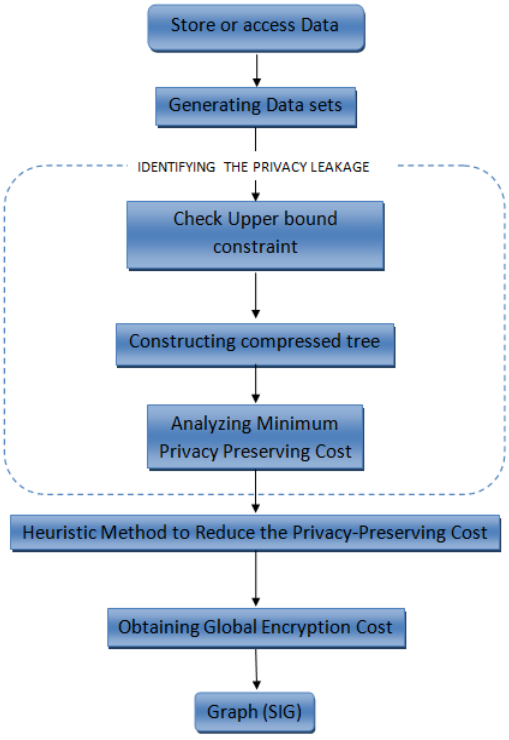

Fig.2. The flow chart showing the construction of SIG graph

C. Privacy Leakage Threshold

Privacy Leakage threshold is provided by the data holder. If privacy leakage of unencrypted data set is less than threshold value, then that data set need not be encrypted. This is defined as Privacy Leakage Constraint also known as PLC. We represent original sensitive data sets as ds0, anonymized intermediate data sets as ds*. The group of sensitive data is denoted as SD and the set of anonymized data is denoted as Quasi-identifiers as QI. The probability for an adversary to get the sensitive information from the sensitive data and Quasi-identifier data is denoted as p(s,q). The privacy leakage of data set ds* is defined as

$$PrLe(ds^*=H(S,Q)-H^*(S,Q). \quad (2)$$

Where H(S,Q) is the entropy of random variable[11] and is defined as

$$H(S,Q)=\log QI.SD. \quad (3)$$

H*(S,Q) is calculated as ,

$$H^*(S,Q)=\sum p(s,q).\log(p(s,q)) \quad (4)$$

In order to satisfy the PLC we divide the PLC recursively into different layers in a SIT. Let the privacy leakage for each layer be (€i).Then the privacy leakage for that layer is less than that of threshold. Threshold for each layer can be calculated as

$$€i = €_{i-1} - \sum PrLe(ds). \quad (5)$$

Minimum privacy cost(CMi) can be calculated using the recursive formula:

$$CMi(€i)=\min\{ \sum (S_k.PRG.f_k) + CM_{i+1} (€i - \sum PrLe(ds_k)) \}, \quad (6)$$

$$CM_{H+1}(€_{H+1})=0$$

D. Heuristic Algorithm

Heuristic value is obtained by heuristic function. Heuristc function is used to calculate the heuristic value of state node (SNi) of a particular layer.The Heuristic value of one layer can be calculated by the following formula:

$$F(SNi)=C_{cur}/(€-€i+1+(€i+1.C_{des}.BF_{AVG})/PrLe_{AVG} \quad (7)$$

Where C_{cur} denotes the privacy preserving cost C_{des} represents total cost of the data sets and BF_{AVG} represents average brand factor. This algorithm works by selecting a state node having highest heuristic value and goes to its child state nodes until it reaches the goal. So in this algorithm a priority queue is used to keep the nodes which add the qualified state nodes. While adding the child nodes into priority nodes the algorithm generates a local encryption solution. Based on the cost and privacy leakage algorithm it sorts the data sets. Thus the data sets with lower privacy leakage are expected to remain not encrypted.

V. Experiment Results and Analysis

The experimental result on real-world data sets is described from which we can see that C-HEU is much lower than C-ALL with various privacy leakage degrees. Even the minimum cost saving of C-HEU over C-ALL at the left side of Fig.3 is more than 40%. Further, we can see that the difference C-SAV between C-ALL and C-HEU increases when the privacy leakage degree increases. This is because looser privacy leakage bounds imply more data sets can remain unencrypted. The reason about the difference between C-HEU and C-ALL with different privacy leakage degree, Fig.3 points out how the difference changes with various numbers of extensive data sets while $€d$ is certain. In most of the real-world cases, data owners would like the privacy leakage of data to be very low. The range of these specific values is rather random and does not affect our analysis because what we want to see is the trend of C-HEU against C-ALL. In most real-world cases, data owners would like the data privacy leakage to be much low. As a result we choose four low privacy leakage degrees of 0.01, 0.05, 0.1, and 0.2 to carry out our experiments. The choice of these specific values is quite random and does not affect our study because what we want is trend of C-HEU against C-ALL. We can see that both C-ALL and C-HEU go up as the number of intermediate data sets is getting larger. That is, the greater the number of intermediate data sets is the more the privacy-preserving cost will be acquired. Most importantly, we can see from that the difference C-SAV between C-ALL and C-HEU becomes bigger and bigger when the number of intermediate datasets increases. That is, more overhead can be reduced when datasets becomes larger. This is the result of the histrionic rise in C-ALL and relatively slower increase in C-HEU when the number of datasets is getting larger.

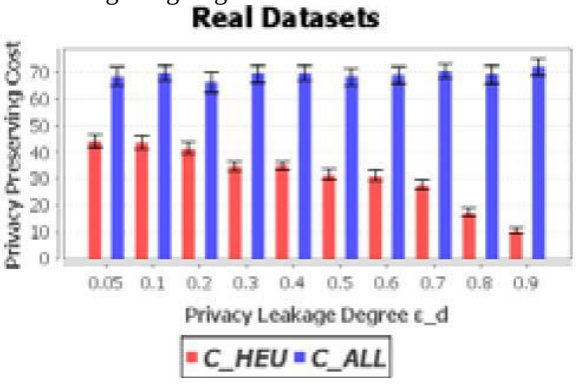

Fig.3. Experiment results about real-world data sets: change in privacy preserving cost in relation to privacy leakage degree.

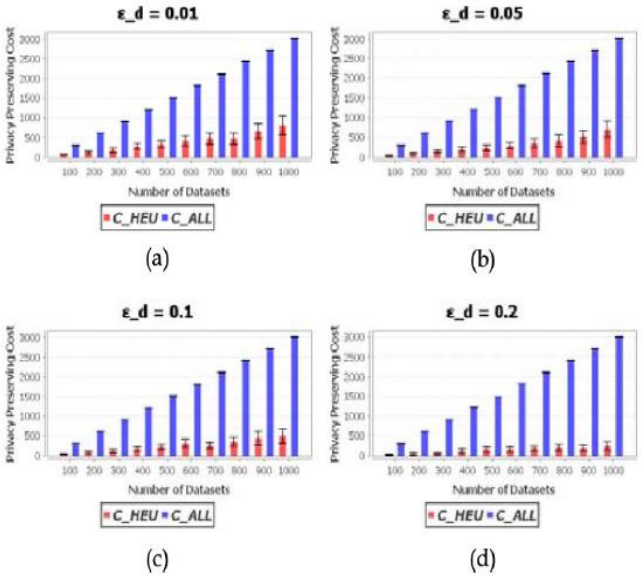

Fig.4 Experiment results in large no of data sets: change in privacy-preserving cost in relation to the number of data sets and the privacy leakage degree.

From the Fig. 4 we can see that the difference between C-ALL and C-HEU becomes bigger and bigger when the number of intermediate data sets increases. The more expense can be reduced when the number of data sets becomes larger. This is the result of the dramatic rise in C-ALL and relatively slower increase in C-HEU when the number of data sets is getting larger. In the view of Big Data, the size and number of datasets and their intermediate data sets are relatively large in cloud. Thus, this trend means our method can reduce the privacy preserving cost significantly in real world scenarios. As a wrap-up both the experimental results expound that privacy-preserving cost intermediate datasets can be saved drastically through our approach over existing ones where all data sets are encrypted.

VI. CONCLUSION AND FUTURE WORK

In this paper, we put forward a unique approach to identify which intermediate data sets need to be encrypted while others do not, in order to fulfill privacy requirements given by data holders. A tree structure is sported from generation relationships of intermediate data sets to evaluate privacy propagation of data sets. Based on such a constraint, we design the problem of saving privacy-preserving cost as a constrained optimization problem. This problem is then separated into a series of sub-problems by decomposing privacy leakage constraints. Finally, we propose a practical heuristic algorithm accordingly to identify the datasets that want to be encrypted. The experimental results on real-world and extensive datasets demonstrate that privacy preserving cost of intermediate data sets can be significantly reduced with our approach over existing ones where all data sets are encrypted.

With our research, we are planning to further investigate the management of intermediate data sets in cloud by using some load balanced scheduling techniques.

REFERENCES

[1] 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.

[2] Liu C, Zhang X, Yang C, Chen J. Ccbke—session key negotiation for fast and secure scheduling of scientific applications in cloud computing. Future Generation Computer Systems 2013; 29(5):1300–1308.

[3] Zhang X, Yang LT, Liu C, Chen J. A scalable two-phase top-down specialization approach for data anonymization using map reduce on cloud. IEEE Transactions on Parallel and Distributed Systems 2013; in press. DOI: 10.1109/TPDS.2013.48

[4] Fung BCM, Wang K, Chen R, Yu PS. Privacy-preserving data publishing: a survey of recent developments. ACM Computer Survey 2010; 42(4):1–53. DOI: 10.1145/1749603.1749605

[5] A. Weiss, Computing in the cloud, ACM Networker 11 (2007) 18–25.

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

[7] Puttaswamy KPN, Kruegel C, Zhao BY. Silverline: Toward Data Confidentiality in Storage-Intensive Cloud Applications. Proceedings of the 2nd ACM Symposium on Cloud Computing (SoCC'11), 2011; Article 10.

[8] 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.

[9] 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.

[10] Cao N, Wang C, Li M, Ren K, Lou W. Privacy-preserving Multi-Keyword Ranked Search Over Encrypted Cloud Data. Proceedings of the 31st Annual IEEE International Conference on Computer Communications (INFOCOM'11), 2011; 829–837.

[11] 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.

[12] Microsoft Health Vault, http://www.microsoft.com health products/Pages/healthvault.aspx, July 2012.

[13] 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.

[14] 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 2010.

[15] M.D.d. Assuncao, A.d. Costanzo, R. Buyya, Evaluating the cost-benefit of using cloud computing to the capacity of clusters, in: 18th ACM International Symposium on High Performance Distributed Computing, HPDC'09, Garching, Germany, 2009, pp. 1–10.

[16] A. Machanavajjhala, D. Kifer, J. Gehrke, and M. Venkitasubramaniam,"L-Diversity:Privacy Beyond K-Anonymity," ACM Trans.Knowledge Discovery from Data, vol. 1, no. 1, article 3, 2010.

[17] 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.

[18] 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.

[19] 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.

[20] 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.