Enabling for Cost-Effective Privacy Preserving of Intermediate Data Sets in Cloud

S.Hemalatha*, S.Alaudeen Basha**

* M.E-Student, Department of CSE, Kongunadu College of Engineering and Technology, Tamil Nadu, India.
** Asst. Professor, Department of CSE, Kongunadu College of Engineering and Technology, Tamil Nadu, India.

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. In order to preserve privacy, the clients will encrypt their data when they out- source it to the cloud. However, the encrypted form of data greatly impedes the utilization due to its randomness. Such data would be stored on the cloud only in an encrypted form, accessible only to users with the correct keys, thus protecting its confidentiality against unintentional errors and attacks.

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. However, processing on encrypted datasets efficiently is quite a challenging task , because most existing applications only run on unencrypted datasets. Although recent progress has been made in homomorphic encryption which theoretically allows performing computation on encrypted datasets, applying current algorithms are rather expensive due to their inefficiency. On the other hand, partial information of datasets, e.g., aggregate information, is required to expose to data users in most cloud applications like data mining and analytics. In such cases, datasets 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 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. Usually, the volume of intermediate datasets is huge .Hence, we argue that encrypting all intermediate datasets will lead to high overhead and low efficiency when they are frequently accessed or processed. As such, we propose to encrypt part of intermediate datasets rather than all for reducing privacy-preserving cost. 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 constrained 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 realworld 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.

II. RELATED WORK

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 coluding with an existing user. In these cases, confidentiality of the entire dataset would be compromised. In the other extreme, we could encrypt each data object with a different key. This increases robustness to key compromise, but drastically increases key management complexity. Our goal is to automatically infer the right granularity for data encryption that provides the best tradeoff between robustness and management complexity. To this end, we

partition the data into subsets, where each data subset is accessed by the same group of users. We then encrypt each data subset using a different key, and distribute keys to groups of users that should have access (based on the desired access control policies).Thus, a malicious or buggy cloud that compromises a key can only access the data that is encrypted by that key, minimizing its negative impact. We introduce a dynamic access analysis technique that identifies user groups who can access different objects in data set. To provide data robustness is to replicate a message such that each Storage server stores a copy of the message. It is very robust because the message can be retrieved as long as one storage server survives. The privacy concerns caused by retaining intermediate datasets in cloud are important but they are paid little attention. Storage and computation services in cloud are equivalent from an economical perspective because they are charged in proportion to their usage. Tight integration of encoding, encryption, and forwarding makes the storage system efficiently meet the requirements of data robustness, data confidentiality, and data forwarding. Thus, cloud users can store valuable intermediate datasets selectively when processing original datasets in data-intensive applications like medical diagnosis, in order to curtail the overall expenses by avoiding frequent re-computation to obtain these datasets. Such scenarios are quite common because data users often reanalyze results, conduct new analysis on intermediate datasets, or share some intermediate results with others for collaboration in cloud that deals with privacy preserving protection for data storage and usage. Although encryption works well for data privacy in these approaches, it is necessary to encrypt and decrypt datasets frequently in many applications. Encryption is usually integrated with other methods to achieve cost reduction, high data usability and privacy protection. Royet al. investigated the data privacy problem caused by Map Reduce and presented a system named Airavat which incorporates mandatory access

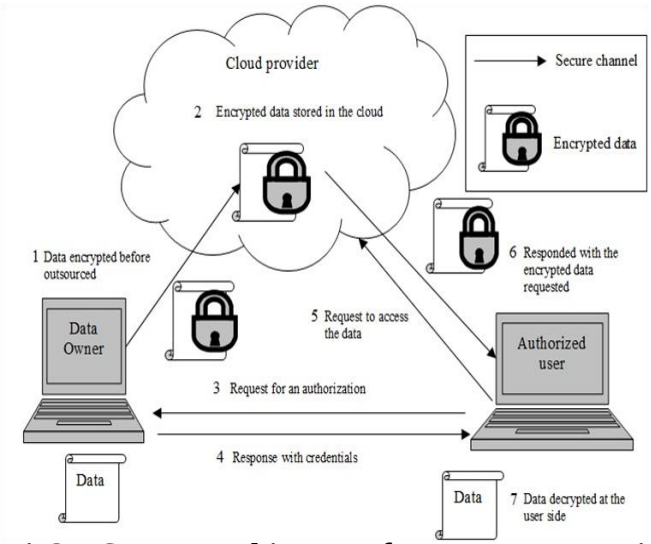

Fig2.1:System architecture for secure transaction using the cloud

control with differential privacy. Puttaswamy et al described a set of tools called Silverline that identifies all functionally encryptable data and then encrypts them to protect privacy. Zhang et al. proposed a system 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 is required to be labeled in advance to make the above approaches available. Ciriani et al. proposed an approach that combines encryption and data fragmentation to achieve privacy protection for distributed data storage with encrypting only part of datasets. We follow this line, but integrate data anonymization and encryption together to fulfill cost-effective privacy preserving. The importance of retaining intermediate datasets in cloud has been widely recognized , but the research on privacy issues incurred by such datasets just commences. Davidson et al. 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 dataitems in their research if economical cost is considered. The PPDP research community has investigated extensively on privacy-preserving issues and made fruitful progress with a variety of privacy models and preserving methods . Privacy principles such as k-anonymity and l-diversity are put forth to model and quantify privacy, yet most of them are only applied to one single dataset. Privacy principles for multiple datasets are also proposed, but they aim at specific scenarios such as continuous data publishing or sequential data releasing. The research in exploits information theory to quantify the privacy via utilizing the maximum entropy principle. The privacy quantification herein is based on the work . Many anonymization techniques like generalization have been proposed to preserve privacy, but these methods alone fail to solve the problem of preserving privacy for multiple datasets. Our approach integrates anonymization with encryption to achieve privacy preserving of multiple datasets.

III. APPROACHES

Our approach works by automatically identifying subsets of an application's data that are not directly used in computation, and exposing them to the cloud only in encrypted form.

• We present a technique to partition encrypted data into parts that are accessed by different sets of users (groups). Intelligent key assignment limits the damage possible from a given key compromise, and strikes a good tradeoff between robustness and key management complexity.

• We present a technique that enables clients to store and use their keys safely while preventing cloud-based service from stealing the keys. Our solution works today on unmodified web browsers.

IV. Motivating Example

A motivating scenario is illustrated in where an online health service provider, e.g., Microsoft HealthVaule , 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 in cloud database. Two independently generated intermediate datasets in Fig.1 are anonymized to satisfy 2-diversity, i.e., at least two individuals own the same quasi-identifier and each quasi-identifier corresponds to at least two sensitive values. Knowing that a lady aged 25 living in 21400 (corresponding quasi-identifier is ⟨214 ∗, female, young⟩) is in both datasets, an adversary can infer that this individual suffers from HIV with high confidence if (a) and (b) are collected together. Hiding (a) or (b) by encryption

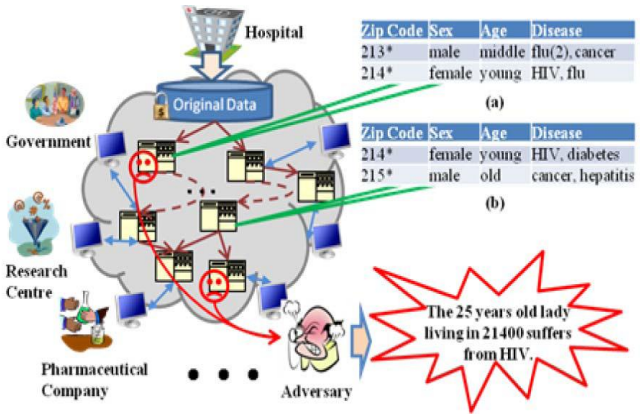

Fig. 4.1. A scenario showing privacy threats due to intermediate datasets.

is a promising way to prevent such a privacy breach. Assume (a) and (b) are of the same size, the frequency of accessing(a) is 10 and that of (b) is 100. We hide (a) to preserve privacy because this can incur less expense than hiding (b). In most real-world applications, a large number of intermediate datasets are involved. Hence, it is challenging to identify which datasets should be encrypted to ensure that privacy leakage requirements are satisfied while keeping the hiding expenses as low as possible.

V. Problem Analysis

5.1Datacentric security and privacy:

Data in the cloud typically resides in a shared environment, but the data owner should have full control over who has the right to use the data and what they are allowed to do with it once they gain access.To provide this data control in the cloud, a standard based heterogeneous data-centric security approach is an essential element that shifts data protection from systems and applications. In this approach, documents must be self-describing and defending regardless of their environments. Cryptographic approaches and usage policy rules must be considered. When someone wants to access data, the system should check its policy rules and reveal it only if the policies are satisfied. Existing cryptographic techniques can be utilized for data security,but privacy protection and outsourced computation need significant attention—both are relatively new research directions.

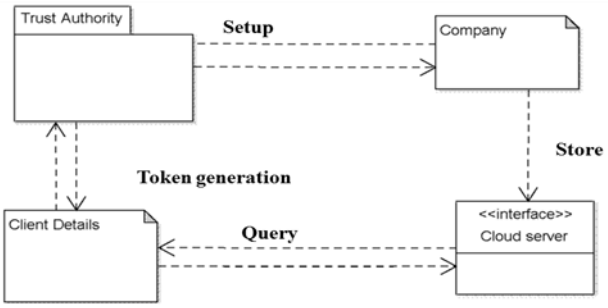

Fig5.1.1:An architecture for privacy and

9/3/2013 16

Data provenance issues have just begun to be addressed in the literature. In some cases,information related to a particular hardware component (storage, processing, or communication) must be associated with a piece of data.

5.2 Privacy-Preserving Cost Problem

Privacy-preserving cost of intermediate datasets from frequent en/decryption with charged cloud services.Cloud service venders have set up various pricing models to support the pay-as-you-go model, e.g., Amazon Web Services pricing model.Practically, en/decryption needs computation power, data storage and other cloud services. To avoid pricing details and focus on the discussion of our core ideas, we combine the prices of various services required by en/decryption into one. This combined price is denoted as $PPRR$. $PPRR$ indicates the overhead of en/decryption on per GB data per execution.Datasets in DD can be divided into two sets. One is for encrypted datasets, denoted as . The other is for unencrypted datasets, denoted as Then, the equations $DD\ eenncc \cup DD\ uunnee = DD$ and $DD\ eenncc \cap DD\ uunnee = \emptyset$ hold. We define the pair $\langle\ , DD\ uunnee\ \rangle$ as a global privacy-preserving solution. The privacy-preserving cost incurred by a solution $\langle\ , DD\ uunnee\ \rangle$ is denoted as $CCpppp\ (\langle DD\ eenncc\ , DD\ uunnee\ \rangle)$. With the notations framed above, the cost $CCpppp\ (\langle\ , DD\ uunnee\ \rangle)$ in a given period [$TT0$, T], can be deduced by the following formula:

$$CCpppp\ (\langle DD\ eenn\ cc\ , DD\ uunnee\ \rangle) = \int_{tt=TT0}^{TT} (\sum dd\ ii \in DD\ eenncc\ SSii \cdot PPRR \cdot ffii \cdot tt\ ) \cdot ddtt. \quad (1)$$

The privacy-preserving cost rate for $CCpppp\ (\langle\ , DD\ uunnee\ \rangle)$,denoted as $CCRRpppp$, is defined as follows.

$$CCRRpppp \triangleq \sum dd\ ii \in DD\ eenncc\ SSii \cdot PPRR \cdot \quad (2)$$

In the real world, $SSii$ and $ffii$ possibly vary over time, but we assume herein that they are static so that we can concisely present the core ideas of our approach. The dynamic case will be explored in our future work. With this assumption, $CCRRpppp$

determines $CCpppp$ (⟨ , $DD\ uunnee$ ⟩ in a given period. Thus, we blur their meanings subsequently. The problem of how to make privacy-preserving cost as low as possible given a SIT can be modeled as an optimization problem on :

Minimize $CCRRpppp = \sum dd\ ii \in DD\ eenncc\ SSii \cdot PPRR \cdot ffii, \subseteq DD.$ (3)

Meanwhile, the privacy leakage caused by unencrypted datasets in $DD\ uunnee$ must be under a given threshold.

Definition :(Privacy Leakage Constraint)

Let $\varepsilon\varepsilon$ be the privacy leakage threshold allowed by a data holder, then a privacy requirement can be represented as $PPPPmm$ ( ) ≤ ,

$DD\ uunnee \subseteq DD$. This privacy requirement is defined as a Privacy Leakage Constraint, denoted as PLC.With a PLC, the problem defined in (3) becomes a constrained optimization problem. So, we can save privacy-preserving cost by minimizing it. As it is challenging to obtain the exact value of $PPPPmm$ ( ), which is formulated our approach is to address the problem via substituting the PLC with one of its sufficient conditions.

5.3 Privacy leakage upper-bound constraint based approach for privacy preserving:

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. we specify relevant basic notations and elaborate two useful properties on a SIT. The privacy leakage upper-bound constraint is decomposed layer by layer . A constrained optimization problem with the PLC is then transformed into a recursive form.and, a heuristic algorithm is de-signed for our approach. We extend our approach to a SIG .

5.4 Recursive Privacy Leakage Constraint Decomposition

To satisfy the PLC, we decompose the PLC recursively into different layers in a SIT. Then, the problem stated can be addressed via tackling a series of small-scale optimization problems. Let the privacy leakage threshold required in the layer $PPii$ be , $1 \le ii \le HH$. The privacy leakage incurred by $UUDDii$ in the solution $\pi\pi ii$ can never be larger than , i.e., $PPPPmm$ ($UUDDii$) ≤ $\varepsilon\varepsilon ii$. The threshold $\varepsilon\varepsilon ii$ can be regarded as the privacy leakage threshold of the remainder part of a SIT after the layer $PPii-1$. In terms of the basic idea of our approach, the privacy leakage constraint $PPPPmm$ ( ) ≤ $\varepsilon\varepsilon ii$ is substituted by one of its sufficient conditions.

5.5 Privacy-Preserving Cost Reducing Heuristic Algorithm

In this section, we design a heuristic algorithm to reduce privacy-preserving cost. In the state-search space for a SIT, a state node $SSNNii$ in the layer herein refers to a vector of partial local solutions,

i.e., $SSNNii$ corresponds to⟨$\pi\pi 1rr$ 1 , … , $\pi\pi iirr\ ii$ ⟩, where $\pi\pi kkrr\ kk \in \Lambda\Lambda kk$, $1 \le kk \le ii$.

Note that the state-search tree generated according to a SIT is different from the SIT itself, but the height is the same. Appropriate heuristic information is quite vital to guide the search path to the goal state. The goal state in our algorithm is to find a near-optimal solution in a limited search space Heuristic values are obtained via heuristic functions. A heuristic function, denoted as ( ), is defined to compute the heuristic value of SSN . Generally, ( ) consists of two parts of heuristic information, i.e., $ff(SSNNii)$ =$FF(SSNNii)$ + $h(SSNNii)$, where the information $FF(SSNNii)$ is gained from the start state to the current state node $SSNNii$ and the information $h(SSNNii)$ is estimated from the current state node to the goal state, respectively.

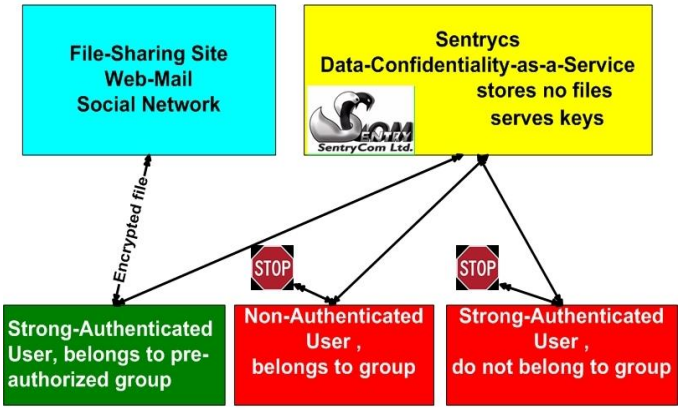

Fig 5.5.1:A scenario in implementation for privacy production

.

Intuitively, the heuristic function is expected to guide the algorithm to select the datasets with small cost but high privacy leakage to encrypt. Based on this, $FF(SSNNii)$ is defined as $FF(SSNNii) \triangleq CCccuurr / (\varepsilon\varepsilon - \varepsilon\varepsilon ii + 1)$, where $CCccuurr$ is the privacy-preserving cost that has been incurred so far, $\varepsilon\varepsilon$ is the initial privacy leakage threshold and $\varepsilon\varepsilon ii$+1 is the privacy leakage threshold for the layers after $PPii$. Specifically, $CCccuurr$ is calculated by $CCccuurr = \sum dd\ rr \in \cup ii\ EEDDkk\ (SSrr \cdot PPRR)$.The smaller total privacy-preserving cost will be. Larger ($\varepsilon\varepsilon - \varepsilon\varepsilon ii$+1) means more datasets before $PPii$+1 remain unencrypted in terms of the

RPC property, i.e., more privacy-preserving expense can saved.

VI. Experiment Results and Analysis

The experimental result on real-world datasets is depicted , from which we can see that $CCHHEEUU$ is much lower than $CCSSPPPP$ with different privacy leakage degree.Even the smallest cost saving of $CCHHEEUU$ over $CCSSPPPP$ at the left side of is more than 40%.Further, we can see that the difference $CCSSSSVV$ between $CCSSPPPP$ and $CCHHEEUU$ increases when the privacy leakage degree increases. This is because looser privacy leakage constraints imply more datasets can remain unencrypted. we reason about the difference between $CCHHEEUU$ and $CCSSPPPP$ with different privacy leakage degree, Fig.6 illustrates how the difference changes with different numbers of extensive datasets while $\varepsilon\varepsilon dd$ is certain. In most real-world cases, data owners would like the data privacy leakage to be much low. The selection of these specific values is rather random and does not affect our analysis because what we want to see is the trend of $CCHHEEUU$ against. At the same

time, we woud like to informatively conduct the experiments. Hence, we select four values. Interested readers can try 3, 5 or other number of values. The conclusions will be similar. we can see that both $CCSSPPPP$ and $CCHHEEUU$ go up when the number of intermediate datasets is getting larger. That is, the larger the number of intermediate datasets is, the more privacy-preserving cost will be incurred. Most importantly, we can see from that the difference $CCSSSSVV$ between $CCSSPPPP$ and $CCHHEEUU$ becomes bigger and bigger when the number of intermediate datasets increases. That is, more expense can be reduced when the of datasets becomes larger. This trend is the result of the dramatic rise in $CCSSPPPP$ and relatively slower increase in $CCHHEEUU$ when the number

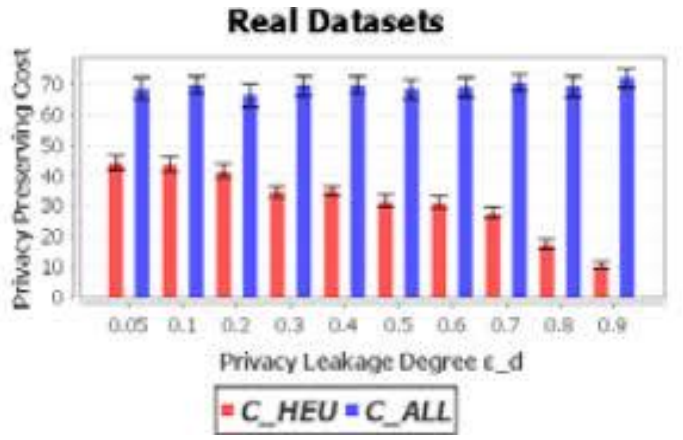

Fig.6.1. Experiment results about real-world datasets: Change in privacy-preserving cost in relation to privacy leakage degree.

of datasets is getting larger. In the context of Big Data, the number and sizes of datasets and their intermediate datasets are quite large in cloud. Thus, this trend means our approach can reduce the privacy preserving cost significantly in real world scenarios. As a conclusion , both the experimental results demonstrate that privacy-preserving cost intermediate datasets can be saved significantly through our approach over existing ones where all datasets are encrypted.

VII. Conclusions and Future Work

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. 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.

References

[1] E. Bertino, F. Paci, and R. Ferrini, "Privacy-Preserving Digital Identity Management for Cloud Computing,"IEEE Computer Society Data Engineering Bulletin, Mar.2009, pp. 1–4.

[2] 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.

[3] 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.H.

[4] Lin and W. Tzeng, "A Secure Erasure Code-Based Cloud Storage System with Secure Data Forwarding," IEEE Trans. Parallel Distrib. Syst., vol. 23, no. 6, pp. 995-1003, 2012. N. Cao, C. Wang, M. Li,

[5] 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 Communications Security (CCS'11), pp. 515-526, 2011.

[6] 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

[7] 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.

[8] 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.

[9] S.B. Davidson, S. Khanna, V. Tannen, S. Roy, Y. Chen, T. Miloand J. Stoyanovich, "Enabling Privacy in Provenance-Aware Workflow Systems," Proc. 5th Biennial Conf. Innovative Data Systems Research (CIDR'11), pp. 215-218, 2011.

Authors

First Author – S.Hemalatha, M.E-Student, Department of CSE, Kongunadu College of Engineering and Technology, Tamil Nadu, India., Email: hemalatha185@gmail.com , Phone: 7502886887

Second Author – S.Alaudeen Basha, Asst. Professor, Department of CSE, Kongunadu College of Engineering and Technology, Tamil Nadu, India., Email: alaudeen.me@gmail.com, Phone: 9944511786