A Secured Cost-effective Multi-Cloud Storage in Cloud Computing

Yashaswi Singh, Farah Kandah, Weiyi Zhang
Department of Computer Science, North Dakota State University, Fargo, ND 58105

Abstract— **The end of this decade is marked by a paradigm shift of the industrial information technology towards a pay-per-use service business model known as** cloud computing. Cloud data storage **redefines the security issues targeted on customer's out-sourced data (data that is not stored/retrieved from the costumers own servers). In this work we observed that, from a customer's point of view, relying upon a solo** SP **for his outsourced data is not very promising. In addition, providing better privacy as well as ensuring data availability, can be achieved by dividing the user's data block into data pieces and distributing them among the available** SP **s in such a way that no less than a threshold number of** SP **s can take part in successful retrieval of the whole data block. In this paper, we propose a secured cost-effective multi-cloud storage (SCMCS) model in cloud computing which holds an economical distribution of data among the available** SP **s in the market, to provide customers with data availability as well as secure storage. Our results show that, our proposed model provides a better decision for customers according to their available budgets.**

Keywords: **Cloud computing, security, storage, cost-effective, cloud service provider, customer.**

I. Introduction

The end of this decade is marked by a paradigm shift of the industrial information technology towards a subscription based or pay-per-use service business model known as cloud computing. This paradigm provides users with a long list of advantages, such as provision computing capabilities; broad, heterogeneous network access; resource pooling and rapid elasticity with measured services [15]. Huge amounts of data being retrieved from geographically distributed data sources, and non-localized data-handling requirements, creates such a change in technological as well as business model. One of the prominent services offered in cloud computing is the cloud data storage, in which, subscribers do not have to store their data on their own servers, where instead their data will be stored on the cloud service provider's servers. In cloud computing, subscribers have to pay the service providers for this storage service. This service does not only provides flexibility and scalability for the data storage, it also provide customers with the benefit of paying only for the amount of data they need to store for a particular period of time, without any concerns for efficient storage mechanisms and maintainability issues with large amounts of data storage. In addition to these benefits, customers can easily access their data from any geographical region where the Cloud Service Provider's network or Internet can be accessed. An example of the cloud computing is shown in Fig. 1. Along with these unprecedented advantages, cloud data storage also redefines the security issues targeted on customer's outsourced data (data that is not stored/retreived from the costumers own servers).

Since cloud service providers (SP) are separate market entities, data integrity and privacy are the most critical issues that need to be addressed in cloud computing. Even though the cloud service providers have standard regulations and powerful infrastructure to ensure customer's data privacy and provide a better availability, the reports of privacy breach and service outage have been apparent in last few years [1] [3] [12] and [13]. Also the political influence might become an issue with the availability of services [8].

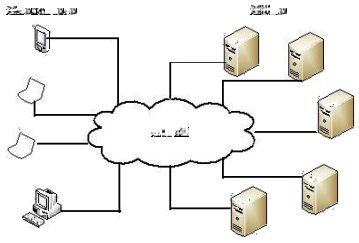

Fig. 1. Cloud computing architecture example

In this work we observed that, from a customer's point of view, relying upon a solo SP for his outsourced data is not very promising. In addition, providing better privacy as well as ensure data availability, can be achieved by dividing the user's data block into data pieces and distributing them among the available SP s in such a way that no less than a threshold number of SP s can take part in successful retrieval of the whole data block.

To address these issues in this paper, we proposed an economical distribution of data among the available SP s in the market, to provide customers with data availability as well as secure storage. In our model, the customer divides his data among several SP s available in the market, based on his available budget. Also we provide a decision for the customer, to which SP s he must chose to access data, with respect to data access quality of service offered by the SP s at the location of data retrieval. This not only rules out the possibility of a SP misusing the customers' data, breaching the privacy of data, but can easily ensure the data availability with a better quality of service.

Our proposed approach will provide the cloud computing users a decision model, that provides a better security by distributing the data over multiple cloud service providers in

978-1-4244-9920-5/11/$26.00 ©2015 IEEE

such a way that, none of the SP can successfully retrieve meaningful information from the data pieces allocated at their servers. Also, in addition, we provide the user with better assurance of availability of data, by maintaining redundancy in data distribution. In this case, if a service provider suffers service outage [1] [12] or goes bankrupt, the user still can access his data by retrieving it from other service providers.

From the business point of view, since cloud data storage is a subscription service, the higher the data redundancy, the higher will be the cost to be paid by the user. Thus, we provide an optimization scheme to handle the tradeoff between the cost that a cloud computing user is willing to pay to achieve a particular level of security for his data. In other words, we provide a scheme to maximize the security for a given budget for the cloud data.

The rest of the paper is organized as follows. The related work is discussed in Section II, followed by the system model and the threat model discussed in Section III. In section IV we discussed the Linear Programming model we propose as a part of our cost-effective security model. A statistical model is implemented using our approach in section V. We conclude the paper in Section VI.

II. Related Work

Privacy preservation and data integrity are two of the most critical security issues related to user data [4]. In conventional paradigm, the organizations had the physical possession of their data and hence have an ease of implementing better data security policies. But in case of cloud computing, the data is stored on an autonomous business party, that provides data storage as a subscription service. The users have to trust the cloud service provider (SP) with security of their data. In [7], the author discussed the criticality of the privacy issues in cloud computing, and pointed out that obtaining an information from a third party is much more easier than from the creator himself.

Following the pattern of paradigm shift, the security policies also evolved from the conventional cryptographic schemes applied in centralized and distributed data storage, for enabling the data privacy. Many of the cryptographic approaches have been proposed for hiding the data from the storage provider and hence preserving data privacy [18] [19] [5]. In [19], the authors proposed a scheme in which, the user's identity is also detached from the data, and claim to provide public auditing of data. These approaches concentrate on one single cloud service provider, that can easily become a bottleneck for such services. In [14], the authors studied and proved that sole cryptographic measures are insufficient for ensuring data privacy in cloud computing. They also argued that the security in cloud storage needs a hybrid model of privacy enforcement, distributed computing and complex trust ecosystems.

One more bigger concern that arises in such schemes of cloud storage services, is that, there is no full-proof way to be certain that the service provider does not retain the user data, even after the user opts out of the subscription. With enormous amount of time, such data can be decrypted and meaningful information can be retrieved and user privacy can easily be breached. Since, the user might not be availing the storage services from that service provider, he will have no clue of such a passive attack.

The better the cryptographic scheme, the more complex will be it's implementation and hence the service provider will ask for higher cost. This could also lead to a monopoly over cloud services in the market. To provide users with better and fair chances to avail efficient security services for their cloud storage at affordable costs, our model distributes the data pieces among more than one service providers, in such a way that no one of the SP s can retrieve any meaningful information from the pieces of data stored on its servers, without getting some more pieces of data from other service providers. Therefore, the conventional single service provider based cryptographic techniques does not seem too much promising.

In [16], the authors discussed distributing the data over multiple clouds or networks in such a way that if an adversary is able to intrude in one network, still he can not retrieve any meaningful data, because its complementary pieces are stored in the other network. Our approach is similar to this approach, because both aim to remove the centralized distribution of cloud data. Although, in their approach, if the adversary causes a service outage even in one of the data networks, the user data can not be retrieved at all. This is why in our model, we propose to use a redundant distribution scheme, such as in [17], in which at least a threshold number of pieces of the data are required out of the entire distribution range, for successful retrieval.

III. Models

First in this section, we will describe our system model and the threat model. Then, formally we will describe our problem statement we are going to study in this paper. Note that, in this work the terms cloud service provider and service provider are interchangeable, the terms cloud storage and cloud data storage are interchangeable, also the terms user and customer are interchangeable.

A. System Overview

We consider the storage services for cloud data storage between two entities, cloud users (U) and cloud service providers (SP). The cloud storage service is generally priced on two factors, how much data is to be stored on the cloud servers and for how long the data is to be stored. In our model, we assume that all the data is to be stored for same period of time.

We consider p number of cloud service providers (SP), each available cloud service provider is associated with a QoS factor, along with its cost of providing storage service per unit of stored data (C). Every SP has a different level of quality of service (QoS) offered as well as a different cost associated with it. Hence, the cloud user can store his data on more than one SP s according to the required level of security and their affordable budgets.

B. Threat Model

Customers' stored data at cloud service providers is vulnerable to various threats. Previous studies in [9], [11] discussed in detail that a cloud service provider can be a victim to Denial of service attacks or its variants.

In our work, we consider two types of threat models. First is the single point of failure [9], [11], which will affect the data availability, that could occur if a server at the cloud service provider failed or crashed, which makes it harder for the costumer to retrieve his stored data from the server. Availability of data is also an important issue which could be affected, if the cloud service provider (SP) runs out of business. Such worries are no more hypothetical issues, therefore, a cloud service customer can not entirely rely upon a solo cloud service provider to ensure the storage of his vital data.

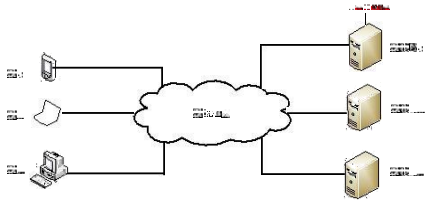

Fig. 2. CSP failure

To illustrate this threat we use an example in Fig. 2. Let us assume that three customers (C1, C2 and C3) stored their data on three different service providers (CSP1, CSP2 and CSP3) respectively. Each customer can retrieve his own data from the cloud service provider who it has a contract with. If a failure occur at CSP1, due to internal problem with the server or some issues with the cloud service provider, all C1's data which was stored on CSP1's servers will be lost and cannot be retrieved. One solution for this threat is that, the user will seek to store his data at multiple service providers to ensure better availability of his data.

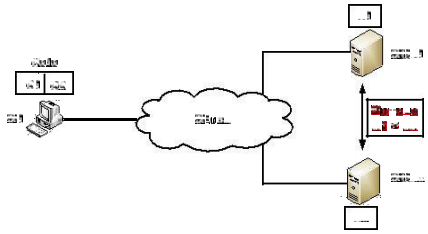

Fig. 3. Colluding cloud service providers

Our second threat discussed in this paper is the colluding service providers [6], in which the cloud service providers might collude together to reconstruct and access the user stored data.

In [16] the authors provide the idea for distributing the data among two storage clouds such that, an adversary can not retrieve the contents of the data without having access to both the storage clouds. Relying entirely upon a couple of service providers for the storage and retrieval of data might not be secured against colluding service providers. Such an attack scenario is entirely passive, because the cloud user can not detect that his information has been collectively retrieved from the service providers without his consent. We illustrate the colluding service providers' threat in Fig. 3. Let us assume that two cloud service providers are available for customer (C1), who want to store his own data securely. In here he will divide his data into two parts (D1 and D2) and distribute these parts on the two available CSP s (CSP1 and CSP2) respectively. The two cloud service providers might collude with each other, and exchange the parts of data that the customer has stored on their server and reconstruct the whole data without being detected by the user.

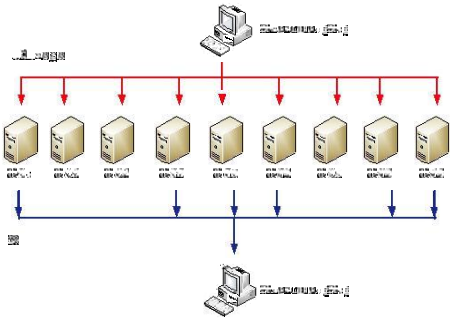

Fig. 4. Data Storage and Retrieval

C. Problem Statement

In this section, we will formally state the definition of our problem that we are going to study.

Given p number of cloud service providers ($SP_i : i \in 1, 2, .., p$). Each SP associated with a QoS factor ($QS_i \in (0, 1)$) along with the cost of storing data units (C_i). Our **S**ecured **C**ost-effective **M**ulti-**C**loud **S**torage Model (**SCMCS**) seeks a distribution of customer's data pieces among the available SP s in such a way that, at least q number of SP s must take part in data retrieval, while minimizing the total cost of storing the data on SP s as well as maximizing the quality of service provided by the SP s.

IV. Secured Cost-effective Multi-Cloud Storage

In this work, to mitigate the threats facing cloud storage, we extended the cloud data storage to include multiple service providers, where each cloud storage represents a different service provider. Our motivation behind such an extension is that, the adversary, similar to any other cloud user, is abstracted from the actual clouds of servers implemented by different cloud service provider.

A. Setup

In this section we describe the setup for the linear programming assignment problem (LP-Assignment) that describes our proposed model. Each cloud customer is provided with p cloud service providers, where each of them offers a QoS level for storage services and required a cost C to be paid by the customer per storage unit of data. Previous studies in [16] [17] proposed a dividing scheme for user's data in such a way that, the user will divide his data into N data pieces where at-least k data pieces out of N data pieces are required to recover any meaningful information of the data. In addition to this (k, N) threshold, we propose another threshold of (q, p); which states that, at least q number of cloud service providers out of p number of cloud service providers must take part in retrieving users data to provide a successful information retrieval. Note

TABLE I
NOTATION AND DESCRIPTION

Notations	Descriptions
N	Total number of data units
k	minimum number of data units required for data retrieval
p	Total number of available cloud service providers
q	Minimum number of service providers required for data retrieval
i	$i = 1,2,.., p$
SP_i	Cloud service provider
QS_i	Quality of Service factor for each service provider
QS_{net}	The QoS achieved at the time of retrieval
C_i	Cost of storing per unit data for i^{th} service provider
C_{tot}	Total cost of storing the distributed customer data on p service providers
a_i	Number of data units assigned to i^{th} service provider
j	$j = 1,2,.., a_i$
$x_{i,j}$	j^{th} data unit on i^{th} service provider

that, each SP_i is assigned a_i pieces of data. Our notations and their corresponding descriptions are listed in table I.

We use an example in Fig. 4 to illustrate our proposed threshold. In this example we assume that we have 9 cloud service providers (SP1, SP2, ..., SP9). Let us assume that a customer (C1) has divided his own data he wish to store on some SP 's servers into 9 data pieces. A customer required to retrieve at least 6 data pieces from different SP s to reconstruct his own data to get the full information, where in our example, six SP s will participate in the data retrieval (SP1, SP4, SP5, SP6, SP8 and SP9).

B. LP-Assignment Problem

One of the obvious objectives in this scenario is to minimize the cost of storage of the data pieces over $p_t$service providers. If a_i is the number of data pieces stored on i^{th} SP which has a per unit cost of storing the data as c_i, then the total cost the customer has to pay is given in Eq. (4.1).

$$C_{tot} = \sum_{i}^{p} a_i c_i \tag{4.1}$$

In our model, we consider $x_{i,j}$ as a binary variable, which is set to 1 if the j^{th} data piece on SP_i becomes a candidate in the current data retrieval. Since the QoS factor depends on the physical location of information retrieval as discussed earlier, the QoS achieved in retrieving the data can be computed as given in Eq. (4.2).

$$QS_{net} = \sum_{i}^{p} \sum_{j}^{a_i} x_{i,j} * QS_i \tag{4.2}$$

Therefore our objective function for our proposed LP-Assignment is to minimize the total cost of storing the distributed customer data on p number of service provides, while maximizing the QoS achieved at the time of retrieval. Our objective is given in Eq. (4.3) and Eq. (4.4).

$$\text{Minimize } [C_{tot}] \text{ and Maximize } [QS_{net}] \tag{4.3}$$

$$\text{Maximize } [QS_{net} - C_{tot}] \tag{4.4}$$

Costraints: Since a_i is the data pieces allocated to be stored at SP_i, this implies:

$$\sum_{i=1}^{p} a_i = N \tag{4.5}$$

Referring to the (k, N) threshold and the (q, p) threshold discussed before, the minimum number of pieces that must be chosen for data retrieval is k, for which at least q service providers are required. Thus, we have:

$$\sum_{j=1}^{p} x_{i,j} \geq q \tag{4.6}$$

and

$$\sum_{j=1}^{p} \sum_{i=1}^{a_j} x_{i,j} \geq k \tag{4.7}$$

where, $N \geq k$ and $p \geq q$. Now, to make sure that a single SP can not retrieve any meaningful information, the number of data pieces allotted to each SP must be less than k:

$$0 < a_i < k \tag{4.8}$$

C. Solution

Since we have multiple optimization objectives as well as a set of variables a_i with non-definitive bounds, it seems to be very complex Linear programming problem. On a more closer perception to the requirements and the objective, we found that this model can be simplified with the help of lemma 1.

Lemma 1. Given N data pieces to be distributed among p service providers such that, at least q service providers must take part in retrieval of data using at least k pieces from the distribution. This implies $q_{max} = k$ and each $a_i = 1$. □

Proof : Let us assume that $q_{max} \neq k$ and there exists at least one service provider, SP_m, that has two data pieces. Since any (at least) k number of data pieces can retrieve the data, we have two situations. First is that, the m^{th} SP does not take part in data retrieval. In this case, the maximum number of service providers that can be used for successful data retrieval is k, where each SP has exactly one data piece. Second is, the m^{th} SP takes part in the data retrieval, here it only needs $k - 2$ data pieces to retrieve the data successfully. Hence, the maximum number of service providers needed to retrieve these data pieces is $k - 2$. So along with m^{th} SP , we needed only $k - 1$ service providers. Here, if we state $q = k - 1$, then it would not be true for any $k - 1$ SP s each having exactly one piece of data. Hence, we can say that, for maximum q, all $a_i = 1$ and this implies that $q_{max} = k$. ■

Using lemma 1, we simplify our Linear programming model to include two binary variables, (s_i) as storage variable and (r_i) as retrieval variable, such that:

$$s_i = \begin{cases} 1 & \text{if } SP_i \text{ is allotted a data piece,} \\ 0 & \text{otherwise.} \end{cases} \tag{4.9}$$

$$r_i = \begin{cases} 1 & \text{if } s_i = 1 \text{ and this piece takes part in data retrieval,} \\ 0 & \text{otherwise.} \end{cases} \tag{4.10}$$

TABLE II
FIRST SCENARIO RESULTS

Service provider	Cost	QoS	Storeage	Retrieve
SP 1	95	0.58	-	-
SP 2	23	0.18	X	X
SP 3	60	0.55	X	X
SP 4	48	0.35	X	X
SP 5	89	0.36	-	-
SP 6	76	0.69	X	X
SP 7	45	0.40	X	X
SP 8	10	0.03	X	-
SP 9	82	0.66	-	-
SP 10	44	0.26	X	X

TABLE III
SECOND SCENARIO RESULTS

Service provider	Cost	QoS	s_i	r_i
SP 1	95	0.58	1	1
SP 2	23	0.18	1	1
SP 3	60	0.55	1	1
SP 4	48	0.35	0	0
SP 5	89	0.36	0	0
SP 6	76	0.69	1	1
SP 7	45	0.40	1	1
SP 8	10	0.03	1	0
SP 9	82	0.66	1	1
SP 10	44	0.26	0	0

where $i = 1,2,..,p$.

Now since, our objective function comprises of multiple objectives, we use goal programming phenomenon to statistically provide weights to each of individual objectives and unite them into a single objective. Hence, our simplified LP problem can be described as bellow:

$$\text{maximize} \quad [w_1 * \sum_{i=1}^{N} (r_i * QS_i) - w_2 * \sum_{i=1}^{p} (s_i * C_i)]$$

$$\text{where} \quad w_1 + w_2 = 1.$$

$$\text{Subject to} \quad \sum_{i=1}^{N} r_i = q,$$

$$\sum_{i=1}^{p} s_i = N,$$

$$\text{and} \quad q = k \leq N \leq p.$$

Owing to the non-negativity principle of linear programming, we have:

$$r_i, s_i, w_1, w_2, C_i, \text{ and } QS_i \geq 0.$$

while,

$$k > 0.$$

Since the basic constraints are mostly equalities, these can easily be disintegrated into two inequalities and can be easily solved in ampl [2]. In the following section we present our results based on our proposed linear programming model.

V. NUMERICAL RESULTS

In this section, to illustrate the performance of our secured cost effective model in providing secure storage scheme as well as availability of data for users we provide our results in two different scenarios.

Our proposed secured cost effective model has to choose the optimal storage allocation based upon the costs of each ser-vice provider, while maximizing the overall quality of service offered.

In our first scenario, we set that the total number of available cloud service providers to 10. Each service provider is associated with randomly generated cost per unit data and quality of service factor. We set the threshold value $k = q = 6$, which specifies that, at least 6 data pieces are needed to retrieve the whole data block. In other words, from lemma 1, at least 6 service providers must take part in data retrieval. We assume that, there is no upper bound on the budget of the customer. Such a model can be used to provide a lower bound for pre-planing estimate for cloud data storage for a customer. Table II represent the results for our first scenario. User's data block was divided into 7 data pieces, where from Table II, it can be seen that by applying our proposed model the data pieces were stored at (SP 2, SP 3, SP 4, SP 6, SP 7, SP 8 and SP 10). To retrieve the whole data block with maintaining a maximized QoS from different SP s, our model will retrieve the 6 data pieces that are required to reconstruct the whole data block from all service providers but SP 8, since it has the least offered QoS. When implemented in ampl [2], the value achieved for the objective function after 18 simplex iterations was -120.942.

In our second scenario, we provide an initial budget for the user, which plays a role as an upperbound to the expenditure of storage and retrieving from SP s. Our proposed model will try to maximize the quality of service it can achieve within the available budget. In this scenario we set the available budget to 400.4 calculated as $k + 1$ times the mean of available costs. In this case, the cost part of our objective function becomes a constraint as given in Eq. (5.1).

$$\sum_{i=1}^{p} s_i * C_i \leq 400.4 \tag{5.1}$$

Our objective function becomes :

$$\text{maximize} \quad \sum_{i=1}^{N} (r_i * QS_i)$$

Table III show the results of our second scenario after 168 total simplex iterations and 25 branch and bound nodes (depth = 7), ampl gives the optimal value for this model as 3.06. The results in Table III show that by taking the customer's budget into consideration, our proposed model will choose the service providers which are affordable for the construer. The chosen service providers are indicated by a value of (1) under the storage variable (S_i) column in the table. Also from the results in Table III, the service provider that have been chosen by our model to provide the customer with a better QoS are indicated by (1) under the retrieval variable (R_i).

VI. CONCLUSION

In this paper, we proposed a secured cost-effective multi-cloud storage (SCMCS) in cloud computing, which seeks to provide each customer with a better cloud data storage decision, taking into consideration the user budget as well as providing

him with the best quality of service (Security and availability of data) offered by available cloud service providers. By dividing and distributing customers data, our model has shown its ability of providing a customer with a secured storage under his affordable budget.

REFERENCES

[1] Amazon.com, "Amazon s3 availablity event: July 20, 2008", Online at http://status.aws.amazon.com/s3-20080720.html, 2008.

[2] "A Mordern Language for Mathematical Programming", Online at http://www.ampl.com.

[3] M. Arrington, "Gmail Disaster: Reports of mass email deletions", Online at http://www.techcrunch.com/2006/12/28/gmail-disasterreports-of-mass-email-deletions/, December 2006.

[4] P. S. Browne, "Data privacy and integrity: an overview", In Proceeding of SIGFIDET '71 Proceedings of the ACM SIGFIDET (now SIGMOD), 1971.

[5] A. Cavoukian, "Privacy in clouds", Identity in the Information Society, Dec 2008.

[6] J. Du, W. Wei, X. Gu, T. Yu, "RunTest: assuring integrity of dataflow processing in cloud computing infrastructures", In Proceedings of the 5th ACM Symposium on Information, Computer and Communications Security (ASIACCS '10), ACM, New York, NY, USA, 293-304.

[7] R. Gellman, "Privacy in the clouds: Risks to privacy and confidentiality from cloud computing", Prepared for the World Privacy Forum, online at http://www.worldprivacyforum.org/pdf/WPF_Cloud_Privacy_Report.pdf, Feb 2009.

[8] The Official Google Blog, "A new approach to China: an update", online at http://googleblog.blogspot.com/2010/03/new-approach-to-china-update.html, March 2010.

[9] N. Gruschka, M. Jensen, "Attack surfaces: A taxonomy for attacks on cloud services", Cloud Computing (CLOUD), 2010 IEEE 3rd International Conference on, 5-10 July 2010.

[10] W. Itani, A. Kayssi, A. Chehab, "Privacy as a Service: Privacy-Aware Data Storage and Processing in Cloud Computing Architectures," Eighth IEEE International Conference on Dependable, Autonomic and Secure Computing, Dec 2009.

[11] M. Jensen, J. Schwenk, N. Gruschka, L.L. Iacono, "On Technical Security Issues in Cloud Computing", IEEE International Conference on Cloud Computing, (CLOUD II 2009), Banglore, India, September 2009, 109-116.

[12] J. Kincaid, "MediaMax/TheLinkup Closes Its Doors", Online at http://www.techcrunch.com/2008/-7/10/mediamaxthelinkup-closes-its-dorrs/, July 2008.

[13] B. Krebs, "Payment Processor Breach May Be Largest Ever", Online at http://voices.washingtonpost.com/securityfix/2009/01/payment_processor_breach_may_b.html, Jan, 2009.

[14] M. Dijk, A. Juels, "On the Impossibility of Cryptography Alone for Privacy-Preserving Cloud Computing", HotSec 2010.

[15] P. Mell, T. Grance, "Draft NIST working definition of cloud computing", Referenced on June. 3rd, 2009, Online at http://csrc.nist.gov/groups/SNS/cloud-computing/index.html, 2009.

[16] P. F. Oliveira, L. Lima, T. T. V. Vinhoza, J. Barros, M. Medard,´ "Trusted storage over untrusted networks", IEEE GLOBECOM 2010, Miami, FL. USA.

[17] A. Shamir, "How to share a secret", Commun. ACM 22, 11(November 1979).

[18] S. H. Shin, K. Kobara, "Towards secure cloud storage", Demo for CloudCom2010, Dec 2010.

[19] C. Wang, Sherman S.-M. Chow, Q. Wang, K. Ren, W. Lou, "Privacy-preserving public auditing for secure cloud storage", in InfoCom2010, IEEE, March 2010.