A New Framework for Secure Cloud Data Storage

Win Win Naing, Thinn Thu Naing
University Of Computer Studies, Yangon, Myanmar
winwinnaing89@gmail.com,ucsy21@most.gov.mm

Abstract

Nowadays, the classic technology called cloud computing is very hot topic. It is widely developed in various environments such as information technology, business and medical, etc. Cloud computing platform offers on-demand services with elasticity, scalability on a simple pay-per-use manner. However, there are many challenges about security and privacy for cloud infrastructure. Because security and privacy preserving is very important task in cloud computing and still implemented with various ways. It is impossible to make complete cloud-based infrastructure without implementation of security policies. In this paper, we propose a new secure cloud data storage with new DateTime-based auditing system that implement for data integrity. We use Trusted-third Party (TTP) and Cloud Service Provider (CSP).

Keywords- Cloud Service Provider; Trusted-third Party; Encryption.

1. Introduction

Cloud computing has recently emerge as a platform for managing, organizing and provisioning large-scale services through an Internet-based infrastructure. It is a new computing paradigm and is being driven by many well known IT and vendors.

Cloud computing builds on top of several other technologies, i.e. distributed computing, grid computing, utility computing and autonomic computing, and it can be envisaged as a natural step forward from the grid-utility model. In the heart of cloud computing infrastructure we find a group of reliable services delivered through powerful *data computing centers* that are based on modern *virtualization* technologies and related concepts such as component-based system engineering, orchestration of different services through *workflows* and *service-oriented architectures* (SOAs) [7].

Using cloud storage, every user can remotely access and store their valuable data, applications and services from a shared pool of configurable computing resources without the burden of local data storage and maintenance.

A Working Definition of Cloud Computing from Mell [4] of NIST is as follows: Cloud computing is a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction. This cloud model promotes availability and is defined in terms of 1) essential characteristics, 2) service models and 3) deployment models. The Essential Cloud Characteristics are: on-demand self-service, broad network access, resource pooling, location independence, rapid elasticity and measured service.

The Cloud Service Models are: Software as a Service (SaaS) – use provider's applications over a network, Platform as a Service (PaaS) – deploys customer-created applications to a cloud and Infrastructure as a Service (IaaS) – Rent

processing, storage, network capacity, and other fundamental computing resources.

The Cloud Deployment Models are:

- *Private cloud:* enterprise owned or leased.
- *Community cloud*: shared infrastructure for specific community.
- *Public cloud*: sold to the public, mega-scale infrastructure.
- *Hybrid cloud*: composition of two or more cloud types.

Cloud computing can be implemented completely within an organizational computing environment as a private cloud. However, the main thrust of cloud computing is to provide a means to outsource parts of that environment to an outside party. As with any outsourcing of information technology services, concerns exist about the implications for computer security and privacy, particularly with moving vital applications or data from the organization's computing center to their consumers or computing center of another organization.

While reducing cost is a primary motivation for moving towards a cloud provider, reducing responsibility for security should not be. Eventually, the organization is accountable for the overall state of the outsourced service. Monitoring and addressing security and privacy issues remain in the purview of the organization, just as other important issues, such as performance, availability, and recovery [3].

From the perspective of data security, which has always been an important aspect of quality of service, Cloud Computing inevitably poses new challenging security threats for number of reasons. Firstly, traditional cryptographic primitives for the purpose of data security protection cannot be directly adopted due to the users' loss control of data under Cloud Computing. Therefore, verification of correct data storage in the cloud must be conducted without explicit knowledge of the whole data. Considering various kinds of data for each user stored in the cloud and the demand of long term continuous assurance of their data safely, the problem of verifying correctness of data storage in the cloud becomes even more challenging. Secondly, Cloud Computing is not just a third party data warehouse. The data stored in the cloud may be frequently updated by the users, including insertion, deleting, modification, appending, reordering, etc. To ensure storage correctness under dynamic data update is hence of paramount importance. However, this dynamic feature also makes traditional integrity insurance techniques futile and entails new solutions. Last but not the least, the deployment of Cloud Computing is powered by data centers running in a simultaneous, cooperated and distributed manner, Individual user's data is redundantly stored in multiple physical locations to further reduce the data integrity threats. Therefore, distributed protocols for storage correctness assurance will be of most importance in achieving a robust and secure cloud data storage system in the real world. However, such important area remains to be fully explored in the literature [9].

The rest of this paper is organized as follows. In the next section, we discuss the related papers with our work. In section 3, we present security threats in cloud computing. And then we describe the proposed scheme in section 4. Finally, we conclude our paper in section 5.

2. Related Work

In this section, we review existing secure cloud data storage and auditing system. Firstly, M.R.Tribhuwan, V.A.Bhuyar, Shabana Pirsade [6] proposed data storage security in cloud computing through Two-way handshake based on Token Management. In the paper, they presented a new two way handshake scheme based on token management. They utilized the

homomorphic token with distributed verification of erasure-coded data to ensure the correctness of users' data in the cloud.Moreover, Cong Wang, Kui Ren, Wenjing Lou, and Jin Li [8] proposed auditing system for secure cloud data storage services. They presented that publicly auditable cloud data storage is able to help nascent cloud economy become fully established. But they had further challenges for accountability, multi-writer model, and performance.

Furthermore, SeongHan, Shin, Kazukuni Kobara [5] also proposed secure cloud storage system. In their system, they showed how the leakage-resilient authentication and data management system works for secure cloud storage. They introduced the LR-AKE client interface.

Some researchers also implemented other secure cloud infrastructure that provides a system for trusted data sharing through untrusted cloud providers. Gansen Zhao, Chunming Rong, Jin Li, Feng Zhang and Yong Tang [10] projected a system to ensure system integrity and confidentiality of data, as cloud service providers may have complete control on the computing infrastructure that underpins the services.

Furthermore, there are a number of security systems for cloud infrastructure. And many researchers have been presented in the literature to support system's security. In this study, we aim to introduce the concept of secure computation and data storage in the context of cloud computing.

3. Security Threats in Cloud Computing

In March 2010, CSA [2] announced the top threats in cloud computing which, if not properly secured, may cause devastating impacts on the mission-critical cloud services. Among those threats,

- Data Loss or Leakage: Some examples to compromise data are 1) operational failures (e.g., improper data deletion or alteration); 2) inconsistent use of encryption and software keys (or loss of encryption keys); 3) unreliable data storage; and 4) insufficient AAA controls to allow unauthorized parties to access sensitive data. The threat of data compromise is mainly due to the architectural or operational characteristics of the cloud environment.
- Account or Service Hijacking: Account and service hijacking, usually with stolen credentials, remains a top threat. With the stolen credentials, an attacker can access critical areas of the deployed cloud services to compromise confidentiality, integrity, and availability of data and those services. If cloud service providers also provide SSO (Single Sign-On) or ID management services, the account and service hijacking would cause the collateral damage on those services as well.

Some general security guidance to deal with the above threats can be found in [1], [2]:

- Encryption and Key Management: Encryption provides data protection while key management enables access to protected data. It is strongly recommended to encrypt data in transit over networks, at rest, and on backup media. In particular, data encryption at rest (e.g., for long-term archival storage) can avoid the risk of malicious cloud service providers or malicious multi-tenants abuse. At the same time, secure key stores (including key backup and recoverability) and access to key stores must be securely implemented since improper (or access to) key storage could lead to the compromise of all encrypted data.
- Identity and Access Management: Secure management of identity and access control is a critical factor to prevent account and service hijacking. It is strongly recommended to prohibit sharing of account credentials, to leverage strong (multi-factor) authentication if possible, and to consider delegated authentication and managing trust across all types of cloud services.

In order to guarantee security against 'Data Loss or Leakage' and 'Account or Service Hijacking' threats, this security guidance would be summarized with three important keywords: Credential Management, Strong Authentication, and Key Management.

4. The Proposed Scheme

In this section, we present a framework design of security protocol for cloud data storage service as shown in figure 1. Although the infrastructures under the cloud are much more powerful and reliable than personal computing devices, they still face a broad range of both internal and external threats to data integrity.

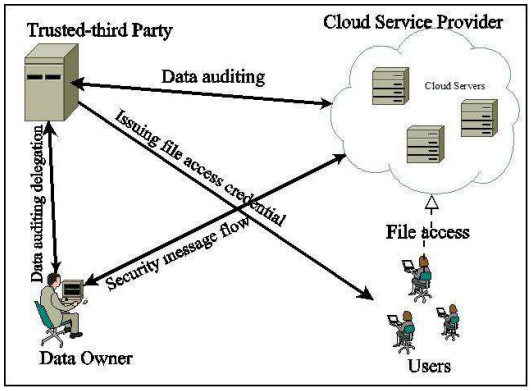


Figure 1. The architecture of cloud data storage service.

Outages and security breaches of noteworthy cloud services appear from time to time. Moreover, for benefits of their own, there are various motivations for CSPs, for monetary reasons, reclaiming storage by discarding data that has not been or is rarely accessed or even hiding data loss incidents to maintain a reputation. In short, although outsourcing data into the cloud is economically attractive for the cost and complexity of long-term large-scale data storage, it does not offer any guarantee on data integrity and availability.

Traditional cryptographic primitives for the purpose of data security protection cannot be directly adopted. In particular, simply downloading the data for its integrity verification is not a practical solution due to the high cost of input/output (I/O) and transmission across the network. Besides, it is often insufficient to detect data corruption only when accessing the data, as it does not give correctness assurance for unaccessed data and might be too late to recover the data loss or damage. Considering the large size of the outsourced data and the owner's constrained resource capability, the tasks of auditing the data correctness in a cloud environment can be formidable and expensive for data owners. Moreover, from the system usability point of view, data owners should be able to just use cloud storage as if it is local, without worrying about the need to verify its integrity [8]. Therefore, we propose DateTime-based auditing system to fully ensure data security and save data owners' computation resources. Trusted-third party provides a transparent method for establishing trust between data owner and cloud service provider (CSP).

The general idea of the proposed system is to encrypt the data before storing on the cloud. The specific security requirements can be summarized as follows.

a) The cloud service provider (CSP) that provides data storage service should not have the capability of cooperation the confidentiality of the data by any means.

b) Sharing of the data can be achieved by the authorization. With a given authorization, authorized users can then access the data kept on the cloud service provider (CSP). The authorization and the access of data should not give the CSP any right to access and to write the data.

In our system, we assume that a public/private key pair with each entity for basic

security mechanisms is already in place to provide basic communication security.

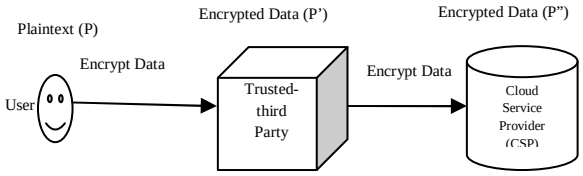
4.1. Data Storage

The system model assumed in this work is a typical cloud computing model with three main players:

1. A **Cloud Owner or Consumer** that employs the cloud storage and process data.
2. A **Cloud Service Provider (CSP)** which manages and operates a cloud infrastructure of storage and computing service.
3. A **Trusted-third Party (TTP)** which loads the authorization mechanisms, cryptographic data structures , data auditing and keying material of more than one cloud customer between the cloud consumer and cloud service provider.

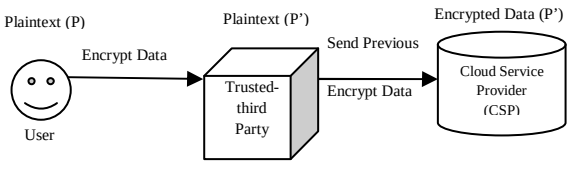
Exploiting data encryption before outsourcing is one way to mitigate the privacy concern. In our system, proposed system allows the user to choose encryption level for their data storage. Therefore, data or application transferring protocol will vary on the customer choosing encryption level. The cloud customers can choose three levels: (i) high, (ii) median (iii) low.

(1) High Encryption Level



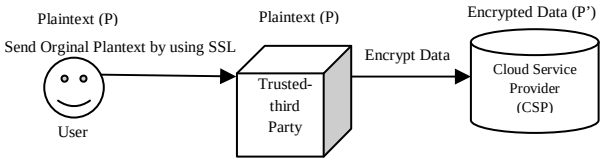
When the customers choose the high encryption level for their data storage, the system will encrypt the plaintext and send to the TTP. And then resulting encrypted data will encrypt next time and send to the CSP. Because we assumed that the Cloud Service Provider are non-trusted. Therefore, the customers' data can't disrupt by CSP.

(2) Median Encryption Level



In this scenario, firstly, the system will encrypt the plaintext and send to the TTP. And then only resulting data will send to the CSP without extra encryption process.

(3) Low Encryption Level



In this scenario, the system will send original plaintext into the TTP without encrypting the data. In this step, we want to use TLS or SSL. In next step, the resulting plaintext will encrypt and send to the cloud service provider (CSP) and store in the cloud servers. Firstly, every customer must get access agreement with TTP to process the secure data storage.

4.2 File Distribution Preparation

In cloud data storage system, many customers store their data and application in the cloud and no longer possess the data and control. Therefore, correctness, availability and integrity of the data files stored on the cloud service provider must be guaranteed.

We plan to use erasure-correcting code to tolerate multiple failures in distributed storage system. In erasure-coded storage systems, when some detectable errors (including disk failures and unrecoverable sector errors) occur, the lost data caused by them can be reconstructed from the redundant data encoded by erasure codes. We rely on this technique to disperse the data file F redundantly across a set of n=m+k distributed servers. A (m+k, k) Reed-Solomon erasure-correcting code is used to create k redundancy parity vectors from m data vectors in such a way that the original m data vectors can be reconstructed from any m out of the m + k data and parity vectors. By placing each of the m + k vectors on a different server, the original data file can survive the failure of any k of the m+k servers without any data loss, with a space overhead of k/m. For support of efficient

sequential I/O to the original file, the unmodified m data file vectors together with k parity vectors is distributed across m+k different servers [8].

4.3. Data Authentication

In proposed system, firstly, every customer must authenticate to use data from the cloud service provider (CSP). The following scenario is authentication steps between trusted-third party (TTP) and cloud users.

a) User firstly launches a mutual authentication between TTP and users in order to confirm that he (or she) is expecting to communicate with TTP.
b) By successful authentication, user submits his request of sharing data stored in CSP to TTP, which contains the information of data requested, user's ID and public key and so on.
c) TTP validates the request. If allows, TTP will send a token with a signature to user, which contains the information of data allowed to share, user's ID and public key, operation type to data and so on.
d) By successful authentication, user submits the token and the token's signature to CSP. After CSP validating the token and user's identity by verifying the token and the token's signature, the data requested by the user from TTP will be sent to user.

4.4. DateTime-Based Auditing System

In this section, we describe auditing approach that is based on date and time parameters. Our system intends to use date/time data as a parameter for data auditing because this method is simple way and it is not necessary to retrieve all file blocks from the server and the server has not access the entire file. The process can be briefly summarized as below. First of all, after file distribution, the user pre-computes some message digest based on date and time that the users' last existing date and time from the CSP. For a piece of data *m*, pre-computing and auditing phase are as follows.

1) User pre-computes a message digest based on date and time (dt) by combing with some random number *r.*

d = m (dt) || *r;* where '||'means concatenation and *m (dt)* means date and time parameters of data m.

2) And then hash the data.

h=H (d); where H (.) is cryptographic hash function.

3) Send this data to the TTP as a message digest and also store in a cloud service provider.
4) TTP stores message digest for every user after leaving their data from the CSP.
5) When the user wants to make sure the storage correctness of the data in the cloud, the user challenges the CSP via TTP.
6) If the two matches in that case the user is ensured that his data on the cloud is stored correctly, and if it does not match then the cloud server is misbehaving.

By using date and time parameters to audit data integrity, our approach only cost constant communication overhead for data auditing because a constant amount of metadata is stored.

5. Conclusions and Future Work

In this paper, we proposed a new secure data storage architecture for cloud computing. This system can be offered various encryption levels to the user. Therefore, users' control can be maximize and provided a more secure data storage protocol. For the file distribution, we will use erasure-coded data storage method. By using it, some detectable errors occur; the lost data caused by them can be reconstructed from the redundant data encoded by erasure codes. We also presented DateTime-based auditing system. This method provides data integrity and constant communication overhead because it is only necessary to extract the corresponding hashed date/time data from the cloud storage server to match it and it is not necessary to extract all file blocks. Moreover, confidentiality can be guaranteed. We plan to implement a fully functional protocol based on our design and evaluate it's perform with other system in the near future.

References

[1] CSA (Cloud Security Alliance), “Security Guidance for Critical Areas of Focus in Cloud Computing V2.1,” December 2009. Available at http://www.cloudsecurityalliance.org/ guidance/.
[2] CSA, “Top Threats to Cloud Computing V1.0,” March 2010.
[3] W. A. Jansen, “Cloud Hooks: Security and Privacy Issues in Cloud Computing,” NIST.
[4] P. Mell and T. Grance, “The NIST Definition of Cloud Computing,” National Institute of Standards and Technology, Information Technology Laboratory, Version 15, 2009.
[5] S. Shin, K. Kobara, “Towards Secure Cloud Storage.”
[6] M.R.Tribhuwan,V.A.Bhuyar, S. Pirsade, “Ensuring Data Storage Security in Cloud Computing through Two-way handshake based on Token Management,” in *IEEE computer society*, 2010.
[7] M. A. Vouk, “Cloud computing - Issues, Research and Implementations,” in *Int. Conf. on Information technology Interfaces*, 2008.
[8] C. Wang, K. Ren, W. Lou, J. Li, “Toward Publicly Auditable Secure Cloud Data Storage Services,” *IEEE Network*, July/August 2010.
[9] C. Wang, Q. Wang, K. Ren and W. Lou, “Ensuring Data Storage Security in Cloud Computing,”
[10] G. Zhao, C. Rong, J. Li, F. Zhang, Y. Tang, “Trusted Data Sharing over Untrusted Cloud Storage Providers,” *in IEEE computer society*, 2010.