IJCAT - International Journal of Computing and Technology, Volume 1, Issue 9, October 2014
ISSN : 2348 - 6090
www.IJCAT.org

A High Level Authentication Management for the Storage Systems in Cloud Computing

[1] Ramisetty Balaji, [2] N. Anand Reddy

[1, 2] M.Tech 2nd Year, Department of CSE, SEAT
Tirupati, AP, India

Abstract - Cloud computing is an emerging technology, which provides services over internet such as software, hardware, network and storage. The key role for cloud computing is virtualization which reduces the total cost and gives reliable, flexible and secured services. However compute service are chosen between the providers located in multiple data centres. One of the major security concerns related to the virtualization and the Storage where the outside attackers can use the files in the storage and the data owners are not capable of knowing attacks. Currently, the amount of sensitive data produced by many organizations is out pacing their storage ability. The management of such huge amount of data is quite expensive due to the requirements of high storage capacity and qualified personnel. Storage-as-a-Service (SaaS) offered by cloud service providers (CSPs) is a paid facility that enables organizations to outsource their data to be stored on remote servers. Thus, SaaS reduces the maintenance cost and mitigates the burden of large local data storage at the organization's end. A data owner pays for a desired level of security and must get some compensation in case of any misbehavior committed by the CSP.

On the other hand, the CSP needs a protection from any false accusation that may be claimed by the owner to get illegal compensations. In this paper, a cloud-based storage scheme is proposed that allows the data owner to benefit from the facilities offered by the CSP and enables indirect mutual trust between them. The proposed scheme has two important features: i) It allows the owner to outsource sensitive data to a CSP, and it ensures that only authorized users (i.e., Those who have the right to access the owner's file) receive the outsourced data i.e. It enforces the access control of the outsourced data can be done by sending a key through email to the registered users and ii) Enables indirect mutual trust between the owner and the CSP using Cheating detection module.

Keywords **- Outsourcing Data Storage, Dynamic Environment, Mutual Trust, CSP, Cheating Detection Module, Access Control, TTP.**

1. Introduction

Cloud storage can provide on-demand, scalable and Quality of Service (QoS) guaranteed storage resource, and users can operate their data anytime and anywhere. Cloud computing is a system whereby data storage, applications and to some degree processing power are freed from local constraints and are available via servers or computers elsewhere in the world. The need for huge local drives is negated, those space-hogging office suites no longer sit on local machines and the option of an extra power boost when needed is within our reach. Facing the powerful and appealing advantages of cloud storage, there are certain problems in existing cloud computing which is impending the fast growth of cloud computing technology, among them few causes are Data security, Data confidentiality and Data access control. So a lot of people and companies are hesitant to put their data in cloud. The main reason is that people and companies are afraid of loss of control on their data. And there are some incidents of data leakage and losing which verify people's fears. To protect stored data, it is not sufficient to use traditional network security techniques that are used for securing messages between pairs of users or between clients and servers. This paper presents a survey of techniques for securely storing data, including theoretical approaches, prototype systems, and existing systems currently available. This paper provides an overview of the prominent characteristics of several systems like Plutus, Sirius, PDP, CPDP, Auditing systems & protocols

I. Secure file systems like Plutus and SiRiUS, which strives to provide strong security even with an un-trusted server. Plutus maintains key distribution in decentralized manner and also data is stored in encryption format. Cryptographic schemes maintained by users rather than servers. The disadvantages are replication leads to over storage, complete key distribution is insecure, cryptographic operations maintained by user is not safe. . But placing complete key in user system may not be a secure one thus key distribution used.

II. Most existing secure storage solutions require the creators of data to trust the storage server to control all users' access to this data as well as return the data intact. So solution on this is provided in (Kan Yang *et al*, 2013) (Ayad Barsoum *et al*, 2013) by introducing a trusted third party or an auditor which will audit the data storage after a regular interval of time and maintain the log of data on its side for same purpose.

1.1. Cloud Computing *Characteristics*

A cloud service should have five essential characteristics:

- *On-demand Self-service*: Managers can obtain services as and when needed, without human intervention.
- *Broad Network Access*: Users can access services using a broad range of network client devices, such as browsers, mobile phones, tablets, laptops, and workstations.
- *Resource Pooling*: Resources are not dedicated to particular consumers but are pooled and allocated as required.
- *Rapid Elasticity*: The amount of resource allocated to a consumer can be expanded and contracted easily and quickly to meet demand.
- *Measured Service*: The amount of resource consumed is measured, to enable automatic expansion and contraction, and to give visibility of usage to providers and consumers.

1.2 Security Properties of Cloud

1.2.1 Data Integrity

Data integrity is defined as the accuracy and consistency of stored data, in absence of any alteration to the data between two updates of a file or record. Cloud services should ensure data integrity and provide trust to the user privacy. Cloud computing poses privacy concerns primarily, because the service provider at any point in time, may access the data that is on the cloud. The Cloud service provider could accidentally or deliberately alter or delete some information from the cloud server. Hence, the system must have some sort of mechanism to ensure the data integrity.

1.2.2 Confidentiality

The confidentiality of a system is guaranteed providing it prevents unauthorized gathering of information. In data secure systems, the "confidentiality" characteristic requires authorizations and checks to be defined, to ensure that information cannot be accessed by subjects who do not have corresponding rights. It must be possible to assign and withdraw the rights that are necessary to process this data, and checks must be implemented to enforce compliance. Cryptographic techniques and access controls based on strong authentication are normally used to protect confidentiality.

1.2.3 Authenticity

The authenticity of a subject or object is defined as its genuineness and credibility; these can be verified on the basis of its unique identity and characteristic features. Information is authentic if it can be reliably assigned to the sender, and if it can be proved that this information has not been changed since it was created and distributed. A secure technique for identifying the communication partners and mechanisms for ensuring authenticity are essential here. These mechanisms must be capable of confirming or disproving the authenticity of the protected information. None of the system participants can create or distribute messages and data on behalf of another subject.

1.2.4 Storage Correctness

It is to ensure users that their data are indeed stored appropriately and kept intact all the time in the cloud.

1.2.5 Newness Property

Receiving the most recent version of the outsourced. Data file is an imperative requirement of cloud-based storage systems. Newness property ensures that the authorized users receive the most recent version of the Data. There must be a detection mechanism if the cloud provider ignores any data-update requests issued by the owner.

1.2.6 Cloud Portability

Cloud portability means the ability to move applications and its associated data between one cloud provider and another or between public and private cloud environments.

2. Related Works

We introduce a model for provable data possession (PDP) that allows a client that has stored data at an untrusted server to verify that the server possesses the original data without retrieving it. The model generates probabilistic proofs of possession by sampling random sets of blocks

from the server, which drastically reduces I/O costs. The client maintains a constant amount of metadata to verify the proof. The challenge/response protocol transmits a small, constant amount of data, which minimizes network communication. Thus, the PDP model for remote data checking supports large data sets in widely-distributed storage system. We present two provably-secure PDP schemes that are more efficient than previous solutions, even when compared with schemes that achieve weaker guarantees. In particular, the overhead at the server is low (or even constant), as opposed to linear in the size of the data.

Experiments using our implementation verify the practicality of PDP and reveal that the performance of PDP is bounded by disk I/O and not by cryptographic computation. Checking data possession in networked information systems such as those related to critical infrastructures (power facilities, airports, data vaults, defense systems, etc.) is a matter of crucial importance. Remote data possession checking protocols permit to check that a remote server can access an uncorrupted file in such a way that the verifier does not need to know beforehand the entire file that is being verified. Unfortunately, current protocols only allow a limited number of successive verifications or are impractical from the computational point of view. In this paper, we present a new remote data possession checking protocol such that:

1) It allows an unlimited number of file integrity verifications;
2) Its maximum running time can be chosen at set-up time and traded off against storage at the verifier.

Storage outsourcing is a rising trend which prompts a number of interesting security issues, many of which have been extensively investigated in the past. However, Provable Data Possession (PDP) is a topic that has only recently appeared in the research literature. The main issue is how to frequently, efficiently and securely verify that a storage server is faithfully storing its client's (potentially very large) outsourced data.

The storage server is assumed to be untrusted in terms of both security and reliability. (In other words, it might maliciously or accidentally erase hosted data; it might also relegate it to slow or off-line storage.) The problem is exacerbated by the client being a small computing device with limited resources. Prior work has addressed this problem using either public key cryptography or requiring the client to outsource its data in encrypted form. In this paper, we construct a highly efficient and provably secure PDP technique based entirely on symmetric key cryptography, while not requiring any bulk encryption. Also, in contrast with its predecessors, our PDP technique allows outsourcing of dynamic data, i.e., it efficiently supports operations. Different approaches have been investigated that encourage the owner to outsource the data, and offer some sort of guarantee related to the confidentiality, integrity, and access control of the outsourced data. These approaches can prevent and detect malicious actions from the CSP side. On the other hand, the CSP needs to be safeguarded from a dishonest owner, who attempts to get illegal compensations by falsely claiming data corruption over cloud servers. This concern, if not properly handled, can cause the CSP to go out of business.

3. Existing System

A solution to detect cheating from owner side as well as CSP side is done through digital signatures. For each file owner attaches digital signature before outsourcing. The CSP first verifies digital signature of owner before storing data on cloud. In case of failed verification, the CSP rejects to store data and asks the owner to resend the correct signature. If the signature is valid, both the file and signature are stored on the cloud servers. The digital signature achieves non-repudiation from the owner side.

When an authorized user (or the owner) requests to retrieve the data file, the CSP sends file, owner's signature and CSP's signature on (file || owner's signature). The authorized user first verifies the CSP's signature. In case of failed verification, the user asks CSP to re-perform the transmission process. If CSP's signature is valid, the user then verifies owner's signature.

If verification fails, this indicates the corruption of data over the cloud servers. The CSP cannot repudiate such corruption for the owner's signature is previously verified and stored by the CSP along with file. Since CSP's signature is attached with the received data, a dishonest owner cannot falsely accuse the CSP regarding data integrity.

4. Our System and Implementation

4.1 System Model

4.1.1 Owner Model.

That can be an organization / individual generating sensitive data to be stored in the cloud and made available for controlled external use.

4.1.2 Cloud Service Provider (CSP)

Who manages cloud servers and provides paid storage space on its infrastructure to store the owner's files and make them available for authorized users.

4.1.3 Authorized Users

A set of owner's clients who have the right to access the remote data.

4.1.4 Trusted Third Party (TTP)

An entity that is trusted by all other system components, and has capabilities to detect/specify dishonest parties. The cloud computing storage model considered in this work consists of four main components as illustrated in Figure 1. The relations between different system components are represented by double-sided arrows, where solid and dashed arrows represent trust and distrust relations, respectively. For example, the data owner, the authorized users, and the CSP trust the TTP. On the other hand, the data owner and the authorized users have mutual distrust relations with the CSP. Thus, the TTP is used to enable indirect mutual trust between these three components. There is a direct trust relationship between the data owner and the authorized users.

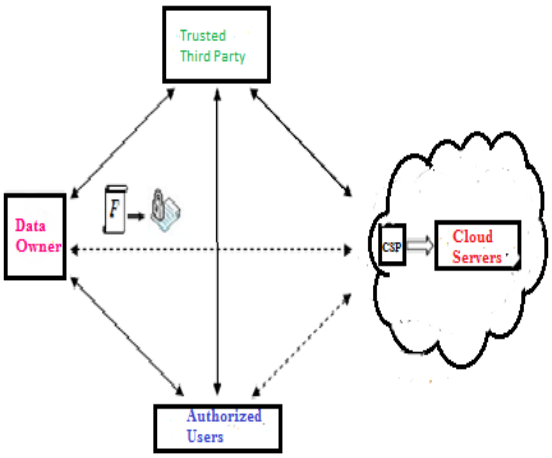

Fig 1: Cloud computing data storage system model

4.2 Updating and Access Control

The Outsourcing of the confidential data has been done by the data owner to the cloud storage servers in an encrypted form. When the authorized users request for data, they will get data in an encrypted form this data can be decrypted by them using the secret key shared among the authorized users. It is assumed that the interaction between the owner and the authorized users to authenticate their identities has already been completed, and it is not considered in this work.

The TTP and CSP must be always online, while the owner is intermittently online .The authorized users able to access data file from CSP even when the owner is offline.

4.3 Cheating Model

The CSP resides in an untrusted domain and thus the confidentiality and integrity of data in the cloud may be at risk. For economic incentives and maintaining a reputation, the CSP may hide data loss, or reclaim storage by discarding data that have not been or is rarely accessed. On the other hand, a data owner and authorized users may collude and falsely accuse the CSP to get a certain amount of reimbursement. They may dishonestly claim that data integrity over cloud servers has been violated.

4.4 Security Requirements

4.4.1 Confidentiality

Outsourced data must be protected from the TTP, the CSP, and users that are not granted access.

4.4.2 Integrity

Outsourced data are required to remain intact on cloud servers. The data owner and authorized users must be enabled to recognize data corruption over the CSP side.

4.4.3 Access control

Only authorized users are allowed to access the outsourced data.

4.4.4 CSP's Defence

The CSP must be safeguarded against false accusations that may be claimed by dishonest owner/users, and such a malicious behaviour is required to be revealed.

5. Framework

Framework consists of notations, setup and file preparation for data outsourcing, data, access and cheating detection of dishonest owner/user.

5.1 Notations

- F is a data file to be outsourced
- h is a cryptographic hash function
- k is a data encryption key/secret key
- Ek is a symmetric encryption algorithm under, e.g., AES (advanced encryption standard) K
- E-1K is a symmetric decryption algorithm under K
- F1 is an encrypted version of the file F
- F1HTTP is a hash value for F1 , and is computed and stored by the TTP
- F1Hu is a hash value for F1 , and is computed by the authorized user
- ENCs(K) is an encrypted version of secret key under S
- is a secret shared between owner and his authorized users. S

5.2 File Preparation for Data Outsourcing

The system setup has two parts: one is done on the Owner side, and the other is done on the TTP side and CSP will stores only encrypted file the data owner generates a secret key K for a file. To achieve privacy-preserving, the owner creates an encrypted file version F1=Ek(F) . For access control he creates encrypted secret key enables only authorized users to decrypt secret key and access the outsourced file. The owner sends F1 and ENCs(K) to the TTP, and deletes the data file from its local storage.

A small part of the owner's work is delegated to the TTP to reduce the storage overhead and lower the overall system computation. For the TTP to resolve disputes that may arise regarding data integrity it computes and locally stores hash value for the encrypted file F1HTTP. The TTP sends encrypted file F1 to the CSP. The TTP keeps only F1HTTP and ENCs(K) on its local storage.

5. 3 Data Access and Cheating Detection

An authorized user sends a data-access request to both the CSP and the TTP. The authorized user receives F1 from the CSP and (F1HTTP, ENCs (K)) from the TTP.

5.3.1 Verification of Encrypted Data File

The authorized user computes hash of encrypted file F1Hu received from the CSP and compare it with one received from the TTP F1HTTP. If F1HTTP ≠ F1Hu then invoke cheating detection procedure at TTP. And if F1HTTP = F1Hu then decrypts ENCs (K) to get Secret Key K and hence decrypts file. Data Access and Verification of encrypted data file.

5.3.2 Cheating Detection Procedure

TTP is invoked to determine the dishonest party. The TTP receives encrypted file F1 from the CSP and computes temporary hash value for encrypted file F1Htemp. If F1HTTP ≠ F1Htemp then reports "dishonest CSP and data is corrupted" to owner. If F1HTTP = F1Htemp then reports "dishonest owner/user and data is not corrupted". Cheating detection procedure is shown in figure 2.

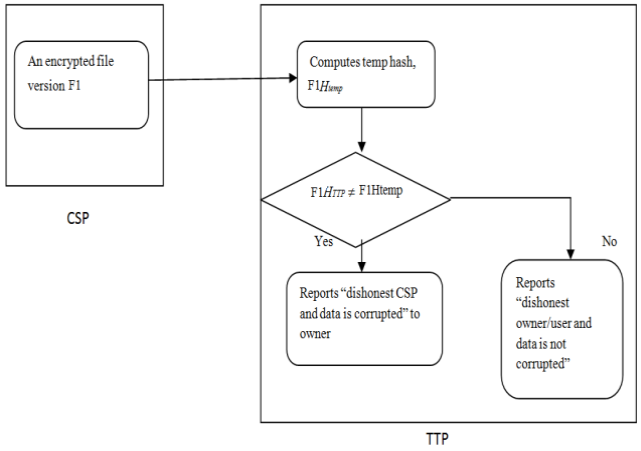

Fig 2: Cheating detection procedure

6. Conclusions and Future Work

The cloud based storage scheme is proposed that allows owner to benefit from facilities offered by the CSP and enables Detection of Dishonest entity (i.e. owner/CSP). It enables data owners to release their concerns regarding confidentiality, integrity, access control of the outsourced data. To resolve disputes that may occur regarding data integrity, a trusted third party is invoked to determine the dishonest party (owner or CSP). Also the security related issues are resolved they are:

(i) Access control is enabled using the Login Modules of each entity, while Login It will validate the credentials given by the each entity.
(ii) Data Confidentiality is achieved using the encryption algorithms.
(iii) Detection of Dishonest Owner/CSP Using the TTP alert module.

A number of future research directions stem from our current research. The area of cloud computing has attracted many researchers from diverse fields; however, much effort remains to achieve the wide acceptance and usage of cloud computing technology

References

[1] C. Erway, A. K¨upc¸ ¨ u, C. Papamanthou, and R. Tamassia, "Dynamic provable data possession," in Proceedings of the 16th ACM Conference on Computer and Communications Security, 2009, pp. 213–222.

[2] Q. Wang, C. Wang, J. Li, K. Ren, and W. Lou, "Enabling public verifiability and data dynamics for storage security in cloud computing," in Proceedings of the 14th European Conference on Research in Computer Security, 2009, pp. 355–370.

[3] A.Juels and B. S. Kaliski, "PORs: Proofs of Retrievability for large files," in CCS'07: Proceedings of the 14th ACM conference on Computer and communications security. ACM, 2007, pp. 584–597.

[4] H. Shacham and B. Waters, "Compact proofs of retrievability," Cryptology ePrint Archive, Report 2008/073, 2008, http:// eprint.iacr.org/.

[5] M. Kallahalla, E. Riedel, R. Swaminathan, Q. Wang, and K. Fu, "Plutus: Scalable secure file sharing on untrusted storage," in Proceedings of the FAST 03 Conference on File and Storage Technologies. USENIX, 2003.

[6] E.-J. Goh, H. Shacham, N. Modadugu, and D. Boneh, "Sirius: Securing remote untrusted storage," in Proceedings of the Network and Distributed System Security Symposium, NDSS. The Internet Society, 2003

[7] G. Ateniese, K. Fu, M. Green, and S. Hohenberger, "Improved proxy re-encryption schemes with applications to secure distributed storage," in Proceedings of the Network and Distributed System Security Symposium, NDSS. The Internet Society, 2005.