S. Janani Devi *et al*, International Journal of Computer Science and Mobile Applications,
Vol.1 Issue. 6, December- 2013, pg. 20-26 **ISSN: 2321-8363**

A Confidential-Secure Public Investigation for Cloud Storage

G.S. Sreetha Devi[1], devisreetha@gmail.com
S. Janani Devi[2], mail2jananis@gmail.com
PG Scholar
Bannari Amman Institute of Technology

Abstract

By cloud memory, consumers are storing large amount of data and using the on demand application and services of cloud computing .Protection of data integrity plays an leading role in cloud computing. The privateness and security is the most troubling problem of the customers. Thus we use public investigation for providing security and protecting the consumer's data from the intruders. Public Investigation is proposed to check the presence of the intruder's. Protection is provided against threats and secures cloud storage system supporting privacy-preserving public investigation. Privateness is maintained for the customer's data and reduces the online pressure to the customer. High Security is given on cloud storage and performance analysis which shows that the proposed scheme is provably highly secure and efficient in manner.

Keywords— cloud computing; data storage; confidentiality protection; public investigation; cryptographic protocols

1. Introduction

Cloud Computing is a common expression used to describe different types of computing concepts that involve a large number of computers connected through a real-time communication network such as Internet and in cloud computing, the word cloud is used as a metaphor for the internet ..It is transforming the very nature of how businesses use information technology. A simple example of cloud computing is Yahoo email, Gmail, or Hotmail etc. Cloud computing is a type of computing that relies on rather than having local servers or personal devices to handle applications in internet. All you need is just an internet connection and you can start sending emails. The server and email management software is all on the cloud and is totally managed by the cloud service provider Yahoo; Google etc. The consumer gets to use the software alone and enjoy the benefits In a cloud computing system, there's a significant workload shift. While Cloud Computing makes these advantages more appealing than ever, it also brings new and challenging security threats towards users Local computers no longer have to do all the heavy lifting when it comes to running applications. The cloud also focuses on maximizing the effectiveness of the shared resources. The network of computers that make up the cloud handles them instead. Hardware and software demands on the user's side decrease. As a result, the correctness of the data in the cloud is being put at risk due to the following reasons.

The goal of cloud computing is to apply high-performance computing power, which is normally used by research facilities, to perform trillions of computations per second, in consumer-oriented applications such as to deliver personalized information, financial portfolios immersive computer games to provide data storage. This shared IT infrastructure contains large pools of systems that are linked together. First of all, although the infrastructures under the cloud are much more powerful and reliable than personal computing devices, they are still facing the broad range of both internal and external threats for data integrity. Secondly, there do exist various motivations for CSP to behave unfaithful towards the cloud users regarding the status of their outsourced data. As users no longer physically possess the storage of their data, traditional cryptographic primitives for the purpose of

data security protection cannot be directly adopted .In particular, simply downloading all the data for its integrity verification is not a practical solution due to the expensiveness in I/O and transmission cost across the network. Besides, it is often insufficient to detect the data corruption only when accessing the data, as it does not give users correctness assurance for those unaccessed data and might be too late to recover the data loss or damage. Considering the large size of outsourced data and the user's constrained resource capability, the tasks of auditing the data correctness in a cloud environment can be formidable and expensive for the cloud users. Moreover, the overhead of using cloud storage should be minimized as much as possible, such that user does not need to perform too many operations to use the data. Users' computation resources as well as online burden, it is of critical importance to enable public auditing service for cloud data storage, so that users may resort to an independent third party Investigation (TPI) to investigate the outsourced data when needed. The TPI, who has expertise and capabilities that users do not, can periodically check the integrity of all the data stored in the cloud on behalf of the users, which provides a much more easier and affordable way for the users to ensure their storage correctness in the cloud. Moreover, in addition to help users to evaluate the risk of their subscribed cloud data services, the investigation result from TPI would also be beneficial for the cloud service providers to improve their cloud based service platform, and even serve for independent arbitration purposes. In a word, enabling public investigation services will play an important role for this nascent cloud economy to become fully established, where users will need ways to assess risk and gain
trust in the cloud. Recently, the notion of public investigation has been proposed in the context of ensuring remotely stored data integrity under different system and security models .Public investigation allows an external party, in addition to the user himself, to verify the correctness of remotely stored data.

This severe drawback greatly affects the security of these protocols in Cloud Computing. From the perspective of protecting data confidentiality, users who own the data and rely on TPI just for the storage security of their data, do not want this investigation process introducing new vulnerabilities of unauthorized information leakage towards their data. Without a properly designed investigation protocol, encryption itself cannot prevent data from "flowing away" towards external parties during the Investigation process. Thus, it does not completely solve the problem of protecting data privacy but just reduces it to the key management. Unauthorized data leakage still remains a problem due to the potential exposure of decryption keys. Therefore, how to enable a third-party investigation protocol, independent to data encryption, is the problem we are going to tackle in this paper. Our work is among the first few ones to support privacy-preserving public auditing in Cloud Computing, with a focus on data storage. As the individual Investigation of these growing tasks can be tedious and cumbersome, a natural demand is then how to enable the TPI to efficiently perform multiple investigation tasks in a batch manner simultaneously. TPA to perform the investigation without demanding the local copy of data and thus drastically reduces the communication and computation overhead as compared to the straightforward data investigation approaches. By integrating the HLA with random masking, our protocol guarantees that the TPI could not learn any knowledge about the data content stored in the cloud server during the efficient investigation process. The aggregation and algebraic properties of the investigator further benefit our design for the batch investing. Specifically, our contribution is of following three aspects:

1. We motivate the public investigation system of data storage security in Cloud Computing and provide a confidentiality protection investigation protocol .our scheme enables an external auditor to investigate user's outsourced data in the cloud without learning the data content.
2. Our scheme is the first to support scalable and efficient public investigation in the Cloud Computing. Specifically, our scheme achieves batch Investigation where multiple delegated Investigation tasks from different users can be performed simultaneously by the TPI.
3. We prove the security and justify the performance of our proposed schemes through concrete experiments and comparisons with the state-of-the-art

2. Obstacles statements

2.1 The structure and Hazard Model

We consider a cloud data storage service involving three different entities they are the *cloud user* (U), who has large amount of data files to be stored in the cloud; the *cloud server* (CS), which is managed by the *cloud service provider* (CSP) to provide data storage service and has significant storage space and computation resources (we will not differentiate CS and CSP hereafter; the *third party Investigator* (TPI),who has expertise and capabilities that cloud users do not have and is trusted to assess the cloud storage service reliability on behalf of the user upon request. Users rely on the CS for cloud data storage and maintenance. They may also dynamically interact with the CS to access and update their stored data for various application purposes. To save the computation resource as well as the online burden, cloud users may resort to TPI for ensuring the storage integrity of their outsourced data, while hoping to keep their data private from TPI. Moreover, the CS may decide to hide the data corruptions caused by server hacks or Byzantine failures to maintain reputation. We assume the TPI, who is in the business on investing, is reliable and independent, and thus has no incentive to collude with either the CS or the users during the investigation process. However, it harms the user if the TPI could learn the outsourced data after the investigation. To authorize the CS to respond to the audit delegated to TPI's, the user can sign a certificate granting investigate rights to the TPI's public key, and all investigate from the TPI are authenticated against such a certificates.

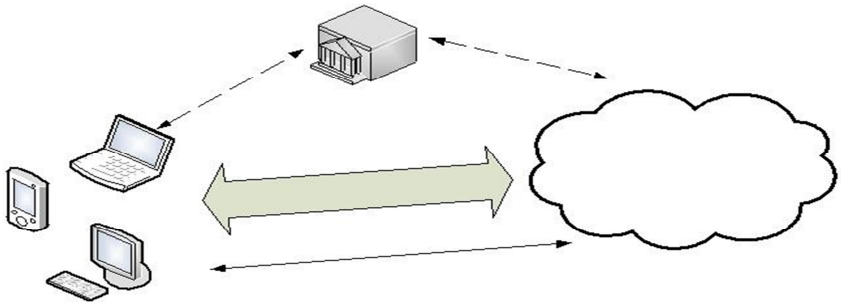

Fig. 1: The architecture of cloud data storage service

2.2 Design Goals

To enable confidential protection public investigation for cloud data storage under the aforementioned model, our protocol design should achieve the following security and performance guarantees.

- Public investigation: to allow TPI to verify the correctness of the cloud data on demand without the architecture of cloud data storage service retrieving a copy of the whole data or introducing additional online burden to the cloud users.
- Storage correctness: to ensure that there exists no cheating cloud server that can pass the TPI's investigated without indeed storing user's data.
- Confidential protection: to ensure that the TPI cannot derive users' data content from the information collected during the investigation process.
- Batch investigation: to enable TPI with secure and efficient investigation capability to cope with multiple investigation delegations from possibly large number of different users simultaneously.
- Lightweight: to allow TPI to perform efficient investigation with minimum communication and computation overhead.

3. The Proposed Design

This section presents our public investigation scheme which provides outsourcing solution of data not only the data itself, but also its integrity checking. We start from an overview of our public investigation system and discuss two straightforward schemes and their demerits. Then we present our main scheme and show how to extent our main scheme to support batch investigation for the TPI upon delegations from multiple users. Finally, we discuss how to generalize our confidentiality protection public investigation scheme and its support of data dynamics.

3.1 Definitions and Framework

We follow a similar definition of previously proposed schemes in the context of remote data integrity and adapt the framework for our confidentiality protection public investigation system.

A public investigation scheme consists of four algorithms .

1. **KeyGen** is a key generation algorithm that is run by the user to setup the scheme.
2. **SigGen** is used by the user to generate verification metadata, which may consist of MAC, signatures, or other related information that will be used for investigation.
3. **GenProof** is run by the cloud server to generate a proof of data storage correctness
4. **VerifyProof** is run by the TPI to investigate the proof from the cloud server.

Running a public investigation system consists of two phases:

1. Setup: The user initializes the public and secret parameters of the system by executing KeyGen, and pre-processes the data file F by using **SigGen** to generate the verification metadata. The user then stores the data file F and the verification metadata at the cloud server, and delete its local copy. As part of pre-processing, the user may alter the data file F by expanding it or including additional metadata to be stored at server.
2. Investigate: The TPI issues an audit message or challenge to the cloud server to make sure that the cloud server has retained the data file F properly at the time of the audit. The cloud server will derive a response message from a function of the stored data file F and its verification metadata by executing GenProof. The TPI then verifies the response via **VerifyProof**.

Our frame work assumes the TPI is stateless, which is a desirable property achieved by our proposed solution. It is easy to extend the framework above to capture a stateful investigation system, essentially by splitting the verification metadata into two parts which are stored by the TPI and the cloud server respectively. Our design does not assume any additional property on the data file. If the user wants to have more error-resiliency, he/she can always first redundantly encode the data file and then uses our system with the data file that has error-correcting codes integrated.

4. Evaluations

4.1Security Analysis

We evaluate the security of the proposed scheme by analyzing its fulfillment of the security guarantee described which is called namely, the storage correctness and confidential protection property. We start from the single user case, where our main result is originated. Then we show the security guarantee of batch investigating for the TPI in multi-user setting.

4.1.1 Storage Correctness Guarantee

We need to prove that the cloud server cannot generate valid response for the TPI without faithfully storing the data, as captured by

Theorem 1*:* If the cloud server passes the Audit phase, then it must indeed possess the specified data intact as it is.

Proof*:* The proof consists of two steps. First, we show that there exists an extractor of μ′ in the random oracle model. Once a valid response {_, μ′} is obtained, the correctness of this statement follows from the theorem. Now, the cloud server is treated as an novel. The extractor controls the random oracle h(·) and answers the hash query issued by the cloud server. For a challenge = h(R) returned by the extractor, the cloud server outputs {_, μ,R} such that the following equation holds.

R · e(_, g) = e((sc Yi=s1 H(Wi)_i) · uμ, v)

Suppose that an extractor can rewind a cloud server in the protocol to the point just before the challenge h(R) is given. Now the extractor sets h(R) to be □ 6= . The cloud server outputs {_, μ□,R}such that the following equation holds.

R · e(_□ , g) = e((sc Y i=s1 H(Wi)_i)□ · uμ□ , v)

The extractor then obtains {_, μ′ = (μ − μ□)/(− □)} as a valid response of the underlying proof of storage. Finally, we remark that this extraction argument and the random oracle paradigm are also used in the proof of the underlying scheme.

4.1.2 Confidential protection Guarantee

We want to make sure that the TPI cannot derive users' data content from the information collected during auditing process.

Theorem 2: From the server's response {_, μ,R}, TPI cannot recover μ′.

Proof: We show the existence of a simulator that can produce a valid response even without the knowledge of μ′, in the random oracle model. Now, the TPI is treated as an adversary. Given a valid _ from the cloud server, firstly, randomly pick, μ from Zp, set R ← e((Qsc i=s1 H(Wi)_i) · uμ, v)/e(_, g). Finally, backpatch = h(R) since the simulator is controlling the random oracle h(·).

4.1.3 Security Guarantee

Now we show that our way of extending our result to a multi-user setting will not affect the mentioned

Theorem 3: Our batch auditing protocol achieves the same storage correctness and confidentiality protection guarantee as in the single-user case.

Proof: The privacy-preserving guarantee in the multi-user setting is very similar to that of Theorem 2, and thus omitted here. For the storage correctness guarantee, we are going to reduce it to the single-user case. We use the forking technique as in the proof of Theorem 1. However, the verification equation for the batch investigates involves K challenges from the random oracle. This time we need to ensure that all

the other K − 1 challenges are determined before the forking of the concerned random oracle response. This can be done using the idea .As soon as the novel issues the very first random oracle query for i = h(R||vi||L) for any i □ [1,K], the simulator immediately determines the values j = h(R||vj ||L) for all j □ [1,K]. This is possible since they are all using the same R and L. Now, all but one of the ks is equal, so a valid response can be extracted similar to the single-user.

4.2 Performance Analysis

We now assess the performance of the proposed privacy-preserving public auditing schemes to show that they are indeed lightweight. We will focus on the cost of the efficiency of the privacy-preserving protocol and our proposed batch auditing technique. The experiment is conducted using C on a Linux system with an Intel Core 2 processor running at 1.86 GHz, 2048 MB of RAM, and a 7200 RPM Western Digital 250 GB Serial ATA drive with an 8 MB buffer. Our code uses the Pairing-Based Cryptography (PBC) library version 0.4.18. The elliptic curve utilized in the experiment is a MNT curve, with base field size of 159 bits and the embedding degree 6. The security level is chosen to be 80 bit, which means |_i| = 80 and |p| = 160. All experimental results represent the mean of 20 trials.

4.2.1 Cost of confidential protection Protocol

We begin by estimating the cost in terms of basic cryptographic operations. Suppose there are c random blocks specified in the challenge.

4.2.2 Batch investigation Efficiency

An asymptotic efficiency analysis on the batch auditing, by considering only the total number of pairing operations. However, on the practical side, there are additional less expensive operations required for batching, such as modular exponentiations and multiplications. Meanwhile, the different sampling strategies that the different number of sampled blocks c are also a variable factor that affects the batching efficiency. Thus, whether the benefits of removing pairings significantly outweigh the seadditional operations is remained to be verified. To get a complete view of batching efficiency, we conduct a timed batch investing test, where the number of investing tasks is increased from 1 to approximately200 with intervals of 8. The performance of the corresponding non-batched (individual) auditing is provided as a baseline for the measurement. Following the same experimental settings c = 300 and c = 460, the average per task investing time, which is computed by dividing total auditing time by the number of tasks, for both batch and individual investigation. It can be shown that compared to individual investigation, batch investing indeed helps reducing the TPI's computation cost, as more than 11% and 14% of per-task auditing time is saved, when c is set to be 460 and 300, respectively.

4.2.3 Sorting out Invalid Responses

Now we use experiment to justify the efficiency of our recursive binary search approach for the TPI to sort out the invalid responses when batch auditing fails, as discussed. This experiment is tightly pertained to the work , which evaluates the batch verification efficiency of various short signatures. To evaluate the feasibility of the recursive approach, we first generate a collection of 256 valid responses, which implies the TPI may concurrently handle 256 different auditing delegations. We then conduct the tests repeatedly while randomly corrupting a _-fraction, ranging from 0 to 18%, by replacing them with random values. The average auditing time per task against the individual auditing approach is presented the result shows that even the number of invalid responses exceeds 15% of the total batch size, the performance of batch investing can still be safely concluded as more preferable than the straightforward individual investigation. Note that the random distribution of invalid responses within the collection is nearly the worst-case for batch investing. If invalid responses are grouped together, it is possible to achieve even better results

5. Conclusion

In this paper, A security of data storage in cloud computing is done by the "confidential-protection public investigation method which is proposed in this paper. TPI is proposed in this paper which would not learn anything about the data stored on the cloud during the efficient investigation process. Considering TPI can simultaneously handle multiple investigate sessions from different users. Then confidential-protection public investigation protocol is further extended into a multi-user setting and the TPI can perform multiple auditing tasks in a good manner for greater efficiency. Thus the analysis process shows that our proposed method are very secure and greater in efficient. In this paper, we propose a privacy-preserving public auditing system for data storage security in Cloud Computing. We utilize the homomorphic linear authenticator and random masking to guarantee that the TPI would not learn any knowledge about the data content stored on the cloud server during the efficient auditing process, which not only eliminates the burden of cloud user from the tedious and possibly expensive auditing task, but also alleviates the users' fear of their outsourced data leakage. Considering TPI may concurrently handle multiple audit sessions from different users for their outsourced data files, we further extend our privacy-preserving public auditing protocol into a multi-user setting, where the TPI can perform multiple auditing tasks in a batch manner for better efficiency. Extensive analysis shows that our schemes are provably secure and highly efficient.

References

[1] Armbrust.M, Fox A, Griffith A, Joseph A.D, Katz R.H, Konwinski A, Lee G, Patterson D.A, Rabkin A, , "Above the clouds: A berkeley view of cloud computing," University of California, Berkeley, Tech. Rep. UCB-EECS-2009-28, Feb 2009.

[2] P. Mell and T. Grance, "Draft NIST working definition of cloud computing," Referenced on June. 3rd, . html, (2009).

[3] Arrington M, "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] S. Wilson, "Appengine outage," Online at http://cio-weblog.com/50226711/appengine outage.php, June 2008.
[5] Amazon.com, "Amazon s3 availability event: July 20, 2008," Online at http://status.aws.amazon.com/s3-20080720.html, 2008.
[6] Juels A and Burton J and Kaliski S, "Pors: Proofs of retrievability for large files," in *Proc. of CCS'07*, Alexandria, VA, October (2007).
[7] Ateniese G,Burns R, Curtmola R, Herring J, Kissner L, Peterson A and Song D, "Provable data possession at untrusted stores," in *Proc. of CCS'07*, Alexandria, VA, October 2007, pp. 598–609.
[8] Shah M.A , Swaminathan R, and Baker M, " Privacy preserving audit and extraction of digital contents," Cryptology ePrint
[9] Wang Q, Wang C, Li J, Ren K and W. Lou, "Enabling public verifiability and data dynamics for storage security in cloud computing," in *Proc. of ESORICS'09, volume 5789 of LNCS*. Springer-Verlag, pp. 355–370(2009).
[10] Kincaid J, "MediaMax/TheLinkup Closes Its Doors," Online at / mediamaxthelinkup-closes-its-doors/, (July 2008)
[11] 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)
[12] Cloud Security Alliance, "Security guidance for critical areas of focus in cloud computing," 2009, http://www. cloudsecurityalliance.org.(2009).
[13] Shah M.A , Baker M,Mogul J.C, and Swaminathan R, "Auditing to keep online storage services honest," in *Proc. Of HotOS'07*. Berkeley, CA, USA: USENIX Association, 2007, pp. 1–6.
[14] Boneh D, Lynn B, and Shacham H, "Short signatures from the Weil pairing," *J. Cryptology*, vol. 17, no. 4, pp. 297–319, (2004).
[15] Ferrara A.L ,Greeny A, S. Hohenberger, and Pedersen M, "Practical short signature batch verification," in *Proceedings of CT-RSA, volume 5473 of LNCS*. Springer-Verlag, 2009, pp. 309– 324.," in *ASIACRYPT*, (2009).
[16] S. Yu, C. Wang, K. Ren, and W. Lou, "Achieving secure, scalable, and fine-grained access control in cloud computing," in *Proc. of IEEE INFOCOM'10*, San Diego, CA, USA, March 2010.

Authors Biography

S. Janani Devi received the Bachelor's degree in Electronics and Communication from Indus college of Engineering, Coimbatore (2012). She is presently pursuing Master's degree in Communication System from Bannari Amman Institute of Technology, Sathyamangalam since 2012.She has extensive skills in QualNet, Exata Emulator, My SQL and Windows Operating System. She presented papers in various conferences. Her research interests are mainly in Wireless Sensor Networks, Wireless Mesh Networks and Cloud computing.

G.S. Sreetha Devi received the Bachelor's degree in Electronics and Communication from Vins Christian college of Engineering Nagercoil (2012). She is presently pursuing Master's degree in Communication System, Sathyamangalam since 2012.She has extensive skills in Java, Networking, Cloud computing and Load Balancing. She presented papers in various conferences. Her research interests are mainly in Cloud computing, Cloud and storage devices.

