Data Auditing In Cloud Using Trapdoor Commitment Scheme

Vijini Mary Kurian,
Department of Computer Science and
Engineering, Karunya University

Roshni Thanka.M
Department of Computer Science and
Engineering, Karunya University

Abstract

Cloud computing can be defined as a technology used to share data as well as service in an effective way. Cloud computing helps to reduce the overall burden of client storage. Many users place their data in the cloud, so correctness of data and security is an important factor. This work studies the problem of ensuring the integrity and security of data storage in Cloud Computing. To ensure this auditing of cloud data is done and the security of auditing is enhanced with Trapdoor Commitment Scheme

1. Introduction

Cloud computing can be simply defined as the delivery of computing and storage capacity as a service to a miscellaneous community of end recipients. With the use of cloud computing, users can easily keep their data in the cloud and use on-demand high-quality applications. The concept of Cloud Computing has been derived from the combination of Grid Computing, Software as a Service and Utility Computing, and essentially represents the increasing trend towards the external deployment of IT resources. Cloud computing can be defined as a general term for anything that involves delivering hosted services through the internet. These services are mainly divided into three categories: Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS) and Software-as-a-Service (SaaS).

There are three major cloud service models [1] they are

- Infrastructure as a Service (IaaS). It allows the consumer to deploy and run arbitrary software, which include several applications and operating systems. It also provides the consumer with the capability to equipping processing, storage, networks, and other fundamental computing resources.
- Platform as a Service (PaaS). It provides the consumer with the capability to deploy onto the cloud infrastructure; consumer created or acquired applications that are developed using programming languages and tools which are supported by the cloud service provider.
- Software as a Service (SaaS). It provides the consumer with the capability to use the provider's applications running on a cloud infrastructure. The various applications are accessible from various client devices such as a web browser.

There are four major deployment models

- Private cloud. In a private cloud the cloud infrastructure is operated for a private organization. It can be managed either by the organization or by a third party, and may exist on or off the premise of third party.
- Community cloud. In a community cloud the cloud infrastructure is shared by several organizations and supports a specific community that has communal importance. It can be managed either by the organization or by a third party, and may exist on or off premise.
- Public cloud. In a public cloud the cloud infrastructure is made available to the general public or a large industry group and is owned by an organization selling cloud services.
- Hybrid cloud. The hybrid cloud is defined as an infrastructure which is a composition of two or more clouds (private, community, or public) that remain as unique entities, but are bound together by any standardized or proprietary technology, that enables data and application.

A cloud service has three distinct characteristics that differentiate it from traditional hosting. It is sold on demand, typically by the minute or the hour; it is elastic - a user can have as much or as little of a service as they want at any given time and the service is fully managed by the cloud service provider (the consumer needs nothing but a personal computer and Internet access).The advantage of cloud is cost savings. The prime disadvantage is security. Cloud computing is used by many software industries nowadays. Since the security is not provided in cloud, many companies adopt their unique security structure. For eg) Amazon has its own security structure. Since the data placed in the cloud is accessible to everyone, security is not guaranteed. To ensure security, cryptographic techniques cannot be directly adopted. Sometimes the cloud service provider may hide the data corruptions to maintain the reputation. To avoid this problem, several auditing process are existing but still the security of these audit process are not ensured. In this paper the technique of Trapdoor commitment scheme is used to ensure the integrity of third party auditor to audit the user's outsourced data when needed.

2. Existing System

There are several existing approaches which try to provide security for the data which are stored at an untrusted server.

2.1. Interactive Audit Scheme

A cryptographic interactive audit scheme also known as interactive PDP or IPDP [9]. It is used to carry out the audit system in clouds. Auditing is done in order to keep the integrity of data in cloud. This scheme is developed on the standard model of interactive proof system, which can provide the confidentiality of secret data and the undeceivability of invalid tags.

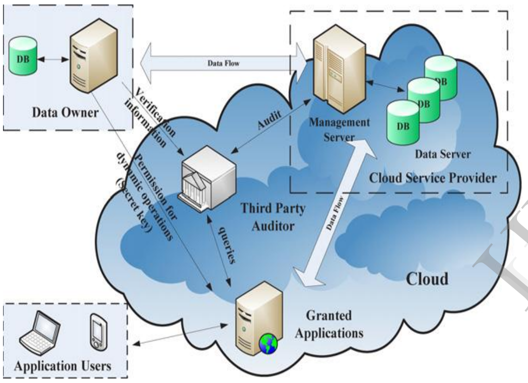

Fig.1. Architecture of interactive audit scheme

For the purpose of auditing a cryptographic interactive audit scheme S is used [9]. It is a collection of two algorithms and an interactive proof system, S = (K, T, P):

- KeyGen(1s): The key generation algorithm takes a security parameter s as input, and process it to return a public-secret keypair (pk, sk);
- TagGen(sk, F): The tag generation algorithm takes two inputs, one is the secret key sk and other one is the file F, and returns the triples (ζ, ψ, σ) where ζ denotes the secret used to generate verification tags, ψ is the set of public verification parameters u and index information χ, i.e., ψ = (u, χ); σ represents the set of verification tags;
- Proof (CSP, TPA): The interactive proof system is a public two-party proof protocol of retrievability between CSP (prover) and TPA (verifier), that is (pk, ψ), where CSP takes as input a file F and a set of tags σ, and a public key pk and a set of public parameters ψ are the common input between CSP and TPA. At the end of the protocol run, TPA returns, where 1 means the file is correctly stored on the server and 0 means the file is corrupted.

Where, the notation P(x) denotes the subject P holds the secret x and (x) denotes both parties P and V. It share a common data x in a protocol. This protocol is provably privacy preserving, and thus may not leak user data information to the auditor. In this audit mechanism the integrity of data is preserved by using the above scheme which keeps the original data secure from both the Cloud Service Provider as well as the Third Party Auditor. Security is assured only by sending some verification data not the full data.

2.2. Drawbacks of Existing System

Interactive Provable Data Possesion provides integrity for the cloud data through auditing. Auditing is done with the help of a Third Party Auditor. Most of existing schemes cannot give a strict security proof against the untrusted CSP's deception and forgery, as well as information leakage of verified data in verification process. These drawbacks greatly affect the impact of cloud audit services. Thus, new frameworks or models are desirable to enable the security of public verification protocol in cloud audit services. In the existing system there is an audit system in which the random key which is generated is sent to the Third Party Auditor (TPA) using that he will do the auditing process. But in this there is no mechanism is used to ensure the credibility of Third Party Auditor (TPA). So if the Third Party Auditor (TPA) is a cheater means there is a chance for data loses. This limitation is overcome in the proposed system.

3. Proposed System

Auditing is one of the efficient techniques among the various mechanisms used to secure data in cloud. In this work the term auditing is used for the process of informing the data owner in case of any kind of data modification or deletion of the cloud data. The enhancement work is mainly based on how to provide better security for audit mechanism. In order to reduce the complexity of auditing rather than auditing the cloud data as whole trapdoors are set to audit the cloud data for a particular.

3.1. Trapdoor commitment Scheme

The enhancement is planned to execute by adding a simple code to do batch auditing. Since there are several types of Cloud services that clients can request from providers, security measures will also vary. As a general rule, with Infrastructure and Platform as a Service clients and users will have more control over their security solutions. This changes when it comes to Software as a Service as in this case, providers also supply security measures along with endpoint applications[11].

Now, when it comes to pre contractual audits, providers often have a hard time reaching an agreement with clients because of their demands regarding the transparency of cyber security measures. Often, reaching a consensus proves to be impossible and this is mostly because users demanding too many details about the security policies applied by providers, which ends up being a violation of the exact same policies. To put it simply, when security transparency is too high it can end up affecting its efficiency even if users and clients are the only one having access to information.

A valid solution to all these pre-contractual issues is to audit the data using Trapdoor Commitment Scheme. Trapdoor commitment scheme that enables a lower-level user to send a short trapdoor to the cloud service provider before retrieving files. This scheme allows the CSP to participate in the partial decipherment, so as to reduce computational overhead on the users without leaking any information about the plaintext. If a lower-level user wants to retrieve a file with limited bandwidth, CPU and memory, the trapdoor which will largely helps to reduce computational power.

3.2. How it works

Initially the Data Owner has to sign the file by using his/her private key to get a signature as ρS. By using the concept of RSA signature (Beuchat et al., 2007), first splits the private key of the upper-level user into two parts. Then encrypt the file by using first part of the private key and sign it. So the Data Owner can get his/her partial signature as ρ1 by signing the original file with respect to his/her partial private key. Now send the partial signature ρ1 to the TPA and commit the trapdoor by executing TCcom commitment algorithm. There should be a valid answer to de commit the trapdoor by revealing that valid answer to the receiver[11].

Once getting the valid answer, the TPA now de commits the trapdoor and gets the sender's partial signature ρ1. After that, the TPA has to get another partial signature from the sender for decrypting the original file. For that, the TPA has to send his/her identity proof and that should be verified by the Data Owner. If the verification is successful then the Data Owner can send the second partial signature ρ2 to the TPA. Now the TPA can get the full signature of the Data Owner by combining the partial signatures. Now the TPA should check whether the second partial signature is valid or not. This can be done by combining the received partial signatures and verify that whether it produce the Data Owners full signature ρS. If so, then the TPA can decrypt the original file from the sender's signature by using his/her public key (Boneh et al., 2004). The data flow of trapdoor commitment scheme is shown in Fig.3.1

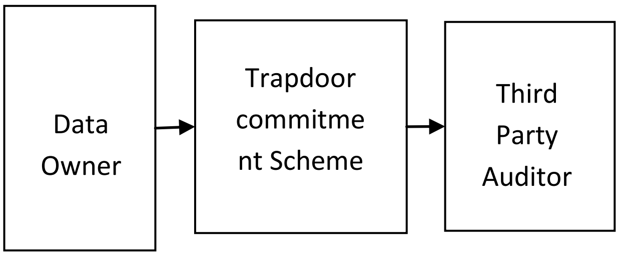

Fig.2. Data Flow of Trapdoor Commitment Scheme

There are three major phases involved in the auditing process. They are

- Cloud Storage
- Audit Phase
- Alerting Data Owner

Among the three phases the Cloud Storage essentially means that the owner (client) of the data moves its data to a cloud service provider and provides verification information to a Third Party Auditor which is supposed to keep the integrity of the data with it and provide audit to the owner whenever required. In the Audit phase where every Third Party Auditor (TPA) is not authorized to do every auditing process. At the time of data upload a verification data will be sent to the Third Party Auditors. It consists of a binary data as well as a metadata generated from the uploaded file. At the same time a cryptographic key will be generated and sent to the data owner. Through the trapdoor commitment scheme only after giving the identity of the Third Party Auditor the cryptographic key will be given to the auditor. With that cryptographic key the data will be audited. Any kind of modification will be alerted to the data owner through mail in the third phase which is alerting data owner. This mechanism will not prevent any kind of data modification instead of that it will alert the data owners about the modification.

4. Previous Works

(Ateniese et al. 2007) are the first one to study about the public auditability in their defined "Provable Data Possession" (PDP) model for ensuring possession of data files on untrusted storages. Their scheme make use of the the RSA-based homomorphic authenticators for auditing outsourced data. It does not sample the whole data instead of that only do sampling a few blocks of the file. But the problem with their approach is that the public auditability in their scheme needs the linear combination of sampled blocks exposed to external auditor. That means when used directly, their protocol is not fully provably privacy preserving, and thus the user data information can be leak to the auditor.

(Juels et al. 2007) describe a "Proof Of Retrievability" (PoR) model. Two methods called spot-checking and error-correcting codes are used in POR to ensure both "possession" and "retrievability" of data files on remote storage service systems. In this mechanism the number of audit challenges a user can perform is a fixed and must be given as priori. The public auditability is also not supported in their main scheme. They describe their concept with the help of a straightforward Merkle-tree construction for public PoR but this approach only works with encrypted data. (Shacham et al. 2008) design an improved PoR scheme built from BLS signatures with full proofs of security in the security model defined in it. (Ateniese et al. 2008a) describes a partially dynamic version of the prior PDP scheme that uses only symmetric key cryptography. However, the system imposes a priori bound on the number of audits and does not support public auditability.

A simple comparison of the several techniques show that all above schemes provide methods for efficient auditing and provable assurance on the correctness of remotely stored data, none of them meet all the requirements for efficient audit service for data integrity in Cloud Computing, as supported in the Interactive Audit Scheme. Table.1 and Table.2 gives the comparison of various features of various methods [9].

5. Conclusion

Here addressing the construction of an efficient audit service for data integrity in clouds. Profiting from the trapdoor commitment scheme, here describes an interactive audit protocol to implement the audit service based on a third party auditor. In this audit mechanism, the third party auditor can issue a periodic verification to monitor any kind of modification of outsourced data by providing an optimized audit mechanism. To understand the audit model, we only need to the security of the third party auditor and deploy a lightweight daemon to execute the verification protocol.

More importantly this new audit approach is based on trapdoor commitment scheme. This approach greatly reduces the security related issues. The key is generated using RSA algorithm which can be obtained by the Third Party Auditor (TPA) only by using a trapdoor commitment. This can be also done by proving that the Third Party Auditor (TPA) is an authorised one. Thus this mechanism enhances the security of audit process.

6. Acknowledgement

I would like to express my gratitude towards my parents, my Staff members and friends who give me support an encouragement in doing this survey. I also thank for all the research scholars who did the previous study on this topic through which I got insight about my topic.

7. References

[1]. Dimitrios Zissis , Dimitrios Lekkas, "Addressing cloud computing security issues", Future Generation Computer Systems, 28(2012) 583-592, December 2010.

[2]. Armbrust, M., Fox, A., Griffith, R., Joseph, A.D., Katz, R.H., Konwinski, A., Lee, G.Patterson, D.A., Rabkin, A., Stoica, I., Zaharia, M, "A view of cloud computing", 2010, Communication ACM 53 (4), 50–58.

[3]. Ateniese, G., Burns, R.C., Curtmola, R., Herring, J., Kissner, L., Peterson, Z.N.J., Song, D.X., "Provable data possession at untrusted stores", International Proceedings of the 2007 ACM Conference on Computer and Communications Security, pp. 598–609, 2007.

[4]. Ateniese, G., Pietro, R.D., Mancini, L.V., Tsudik, G., "Scalable and efficient provable data possession", International Proceedings of the 4th International Conference on Security and Privacy in Communication Networks, Secure Communication, pp. 1–10, 2008.

[5]. Erway, C.C., Küpc¸ ü, A., Papamanthou, C., Tamassia, R., "Dynamic provable data possession", International Proceedings of the 2009 ACM Conference on Computer and Communications Security, pp. 213–222, 2009.

[6]. Juels Jr., A., Kaliski, B.S., "Pors: proofs of retrievability for large files", International Proceedings of the 2007 ACM Conference on Computer and Communications Security, pp. 584–597, 2007.

[7]. Shacham, H., Waters, B., "Compact proofs of retrievability", 14th International Conference on the Theory and Application of Cryptograpy and Information Security, pp. 90–107, 2008.

[8]. Wang, C., Wang, Q., Ren, K., Lou, W., "Privacy-preserving public auditing for data storage security in cloud computing", International Conference on Computer Communications Proceedings IEEE, pp. 1–9, 14-19, 2010.

[9]. Yan Zhua,b,, Hongxin Huc, Gail-Joon Ahnc, Stephen S.Yauc, "Efficient audit service outsourcing for data integrity in clouds", The Journal of Systems and Software 85 (2012) 1083– 1095, 2011.

[10]. Balakrishnan.S, Saranya.G, Karthikeyan.S and Shobana.S, ",Introducing Effective Third Party Auditing (TPA) for Data Storage Security in Cloud", International Journel of Computer Science and Technology IJCST Vol. 2, Iss ue 2, June 2011.

[11]. M. Sowmya Varshini, D. Palanikkumar, G. Rathi, "An Improved Security Enabled Distribution of Protected Cloud Storage Services by Zero-Knowledge Proof based on RSA Assumption", International Journal of Computer Applications (0975 – 8887) Volume 40– No.5, February 2012.