Privacy-preserving public auditing for data integrity in cloud

M Shaik Saleem and M Murali
SRM Institute of Science and Technology, Kattankulatur
E-mail: saleemshaik375@gmail.com and murali.m@ktr.srmuniv.ac.in

Abstract. Cloud computing which has collected extent concentration from communities of research and with industry research development, a large pool of computing resources using virtualized sharing method like storage, processing power, applications and services. The users of cloud are vend with on demand resources as they want in the cloud computing. Outsourced file of the cloud user can easily tampered as it is stored at the third party service providers databases, so there is no integrity of cloud users data as it has no control on their data, therefore providing security assurance to the users data has become one of the primary concern for the cloud service providers. Cloud servers are not responsible for any data loss as it doesn’t provide the security assurance to the cloud user data. Remote data integrity checking (RDIC) licenses an information to data storage server, to determine that it is really storing an owners data truthfully. RDIC is composed of security model and ID-based RDIC where it is responsible for the security of every server and make sure the data privacy of cloud user against the third party verifier. Generally, by running a two-party Remote data integrity checking (RDIC) protocol the clients would themselves be able to check the information trustworthiness of their cloud. Within the two party scenario the verifying result is given either from the information holder or the cloud server may be considered as one-sided. Public verifiability feature of RDIC gives the privilege to all its users to verify whether the original data is modified or not. To ensure the transparency of the publicly verifiable RDIC protocols, Let’s figure out there exists a TPA who is having knowledge and efficiency to verify the work to provide the condition clearly by publicly verifiable RDIC protocols.

1. Introduction
Cloud computing is a information technology paradigm, a miniature for enabling everywhere to get the shared configurable resources which can be immediately vend with minimum authority creation, usually over the internet. Cloud computing allows users and business with different computing potential to store and process data either in a private cloud or on a public cloud. There will be an third party also in the data centre. Thus how the cloud makes the data access function more dynamic and significant. With the sharing of the resources the cloud computing achieve the economic scale and the coherence.

The cloud users get number of assets from the Cloud computing. Users expense for the CAPEX is almost negligible while software and service charge will vary as per the usage. The user can access wide range of applications and data immediately where ever they have a network. Cloud computing applications extent over in many domains like social networks, life science, health care, business, data analytics etc.

The users are still facing the problem over the data integrity both in the internal and external within large area. The growth in the cloud computing with in infrastructure where the cloud service providers provision the computing and storage resources to their clients or business, to provide integrity guarantees for outsourced data management will be the significant importance. The tremendous problem relates with the untrusted servers provide data integrity verification in cloud data storage. The useful techniques of security in cloud storage is to verify data integrity stored in public cloud is(cloud storage auditing).
2. Related works

In [1] clients can remotely accessed the stored data and use the resources on demand without any difficulty of storage and maintenance using the cloud storage. In cloud the user do not have the ownership of the outer data makes protection in data integrity. For the usage of the cloud storage the user need not to verify the integrity they can use it as a local one, this makes the public auditability for the cloud storage is importance.

So in the [1] the author has proposed a secure cloud storage system supporting privacy preserving public auditing. Public auditing allows the external party to verify the correctness of remote store data. One of the ways to less harsh is using the data encryption before the outsourcing. So in [1] the author proposed how to enable the privacy preserving TPA protocol absolute to data encryption. In [1] the author address the problem that an individual auditing of the growing task can be difficult to handle.

To finally play out numerous reviewing charge in a cluster way outsourcing a natural demand is then enable. So to address this problem they used a technique of public key based homomorphic linear authentication [HLA]. The Homomorphic linear Authentication (HLA) is situated on BLS algorithm. By combining the HLA with random masking the protocol guarantee the TPA couldn’t know the learning of the information content put away in the cloud server amid the reviewing procedure.

Fig 1 shows that the architecture of cloud storage service where it is interconnect with the users, TPA, and the cloud server. The process of Data auditing Delegation is done between the user and the TPA after that the TPA make auditing of data which the cloud servers store it. The audited data will become secure and the data flow is done with the request of server or cloud.

Advantages:

- No new vulnerabilities towards user data privacy.
- Introduce no additional online burden to user.

Disadvantages: For the integrity verification it is not a practical solution that to download all the data. Integrity of unaccessed data within the network are not assured due to high I/O and transmission cost.

In [2] the client will stores vast amount of data in the server cloud which is an untrusted one, the client can ask the server to project a function on some section of outsourced data at any point. Instead of no longer knowing the inputs of the confide computation. The client must be able to efficiently verify the correctness of the result.[2] correctness proof and are verifiable by clients when the data is in cloud is estimate with the outsourced.

The loss of data is very common but the people runs the site can directly understand how nasty a thought of any information misfortune is, sometime it is not possible to uncover data iniquity when gaining the data because it is too late to get back the corrupted data. Therefore, it is important for the clients to check intermittently whether the outsourced information are put away appropriately or not. So for this remote data integrity is highly pleasant for checking a secure cloud storage.

Here in [2] they proposed a novel cryptographic technique that provides to solve the outsourced problem for the class of computation of quadratic polynomial over a variables which consists of large numbers. This class screens a broad range of imperative arithmetic computing notability more significant statistic to conform the completeness of solution.

In their work they focus on the relevant problem of delegation computation on constantly growing data and in outsourced to remote server. They proposed a protocol that gains all the desiderata requirements for a confined yet useful and practical. There work is related to the 1.memory delegation 2.streaming delegation.
Advantages:

- Within the time independent of the input-size clients should be able to verify.
- Capture a wide range of significant arithmetic computations.

Disadvantages:

- At a later time it not allow to supplement additional data.
- Solution can still not capture general-purpose computations.

In [3] to make the information safe of the secret key from the opponent there are many cryptographic pattern are construct under this theory. However, the ideology will not hold every time as the opponent could restore the data about the key through different factors, for example, side channel attacks, memory spillage attacks or by arrange the framework on which the keys are put away. These incursion, as often as possible credit to as break age stream. The hiding of the partial information about the file that constructed in the framework [3] i.e leakage-resilient identification (ID) protocols in the bounded retrieval model (BRM) from proofs of storage.

In [3] for hiding the incomplete information of file they provided a frame work for constructing leakage-resilient identification protocol in the bounded retrieval model from proofs of storage that hide the incomplete information of file. Opponent can get back the secret key with an complete upper bound on the total volume of information by bounded retrieval model Bounded retrieval model (BRM). Here they consider a problem of identification in BRM. For a distinct pattern based on RSA they provided a new different area. They showed the POS theory can be used to build leakage resilient identification protocol in the BRM. They expressed based on Rivest Shamir Adleman (RSA). Problem: Missing step towards an efficient compiler between homomorphism ID and leakage resilient ID scheme is to find an efficient compiler between Proofs of Storage (POS) and Zero Knowledge-Proofs of Storage (ZK-POS).

In [4] to corroborate the honesty of data in public cloud the cloud storage auditing is noticed as an imperative service. All the rules that are used for auditing in the existing system stands with the hypothesis that the client secret key is fully protected. Due to the expected low security framework or sensibility in security such premise may not always be held. In many of the common auditing protocol would necessarily will not work when a secret key for verifying is resolved.

In [4] the author explore a new problem setting in the cloud storage system is how to accomplish key introduction protection. It is brought into concentrate here on the best way to decline the harm of the client enter introduction in public storage auditing. To bring about this on different cryptographic design they apply the binary tree structure to update the clients secret key.

For the property of block verifiability and forward security the author also designed a unique authenticator support. Author proved that the guarantee of current protocol in the specified model security and sustain its work along with concrete asymptotic analysis. In this the author deliberate how to accord with the clients key clue in cloud storage auditing. The efficient resemblance between current protocol and easier protocol based on BLS sign algorithm has been granted.

Fig 2 system associate with 2 parties, the data owner and the cloud. User makes the file and upload in the cloud server and if the user calls for the file to the server the cloud makes the user to download it and each of the data in the files are going to be divided into a multiple blocks. Here the user can verify annually even if the file stored in cloud are correct or not. The files that are going to be saved in the cloud permanently and the cloud server will divide the file into T+1.
In [5] the cloud rather than its client a TPA is considered to validate the integrity of the dynamic data to safeguard the problem of integrity of data storage in cloud. In complete economies of scale for Cloud Computing the TPA concludes the action of the client through to examine whether the data which is stored in the cloud is easily not damaged. The aid for the most normal forms of data operations such as block modification, insertion and deletion, along with data dynamics is also an important step toward practicality. In cloud computing as the services are not confine to substitute only data. In the time the previous works on safeguard the remote data integrity frequently require the help of either open auditability or operations of dynamic information, solves these two problems.

The author improved the existing proof of storage models for execute efficient data dynamics by influence the classic Merkle Hash Tree construction for block tag authentication. The author further probe the technique of bilinear aggregate signature to extend their main result into a multi-user setting. In that to help adequate convey of different audit job where TPA can play out various reviewing assignments simultaneously.

In [6] for the discrete classes of functions, such as linear combinations, high-degree univariate polynomials, and multivariate quadratic polynomials in this work they construct a theory for arbitrary computations and highly efficient schemes for delegation. This solution make the clients to store a huge amount of encrypted data sets on a cloud server; they can receive encrypted results that can be efficiently verified as well as decrypted and query statistics over this data.

There convention accept the presence of a completely homomorphic encryption Fully homomorphic encryption (FHE) plot and a not necessarily private, but secure verifiable computation verifiable computation (VC) scheme. The basic idea is to encrypt the data with the FHE scheme, and to run the VC scheme on the function EvalFHE(f). This technique prevents the server from using the client as a decryption oracle for the FHE, since the acceptance bit is determined before decryption, just according to the correctness of the evaluation of EvalFHE(f).

In the work [6] the author focus on the problem where a client stores a large data set (x), on a server and later asks the server to compute functions f over x and the solutions work in the amortized model in which the client spends a single pre-processing phase whose cost is as running f(x), and later amortizes this one-time cost over several function evaluations. They constructed a VC scheme in the case the client stores several encrypted data sets at the server, and then asks it to compute a quadratic polynomial (f) on any of the outsourced sets.

In work [6] the solution after a single pre-processing for every (f), the client can verify results in constant time. This achieves input and output privacy. They considered the task of outsourcing linear combinations. For this they constructed a very clean and efficient solution which provides both input and function privacy, but has no efficient verification. To design there protocols they followed the blueprint of the generic scheme and they develop additional techniques that provide significant efficiency improvements.

In [6] the basic idea is to encrypt the data with a somewhat homomorphic encryption scheme (for privacy), and to add an authentication mechanism on top of the ciphertexts (for security). For the encryption, they chose a slightly modified version of the scheme by Brakerski and Vaikuntanathan (BV). The server stores t = (t0, . . . , tt) where ti is a BV encryption of xi. For authentication, they rely on homomorphic MACs. The generic idea for there schemes is to generate a MAC C i for every ciphertext ti, and then use the homomorphic property of the MAC to authenticate the BV homomorphic evaluation.

Advantages:
• The clients to store a huge amount of encrypted data sets on a cloud server, they can receive encrypted results that can be efficiently verified as well as decrypted and query statistics over this data.
• Compute several statistics on remotely stored data sets in a private and verifiable manner.

Disadvantages:
• The client’s acceptance bit may leak due to error messages, protocol termination.
• No definition of private and secure VC with presence of verification Queries.

3. Problem definition
In existing framework the Remote data integrity checking (RDIC) developments depend on (public key infrastructure) PKI where an advanced authentication is utilized to ensure the certified of a clients open key. These developments cause complex key administration strategies since certificate generation, storage, update revocation time is time consuming and expensive. In data integrity checking with open certainty, an auditor (or anybody) can confirm the honesty of the cloud information. In this situation, information security against the outsider verifier is extremely basic as the cloud clients can save private or sensitive files may the business deals or medicinal files to the cloud. The customers enormous information is outside his control.

The problem with the existing system is whenever the user uploads the data a digital certificate is generated. This declaration is utilized to ensure the honest to goodness of a clients public key. These developments bring about complex key administration strategies in light of the fact that each time at whatever point the client will transfer any information the declaration must be created regardless of what measure of information and how often the client going to transfer. Another issue is the time-consuming and expensive for the certificate revocation. It takes time and difficult to regenerate the certificate if the certificate is lost or cancelled. Whenever the user is storing the data in untrusted place the very important thing is that the user wants to make data secure. The user wants there should be no leakage of data, but in the existing system the will be a third party verifier where it can be get the whole user data. Against the TPA there will be no privacy.

• Public key approved based on client's absolute work by the digital certificate.
• These developments raise complex key administration techniques.
• Security against the outsider verifier. That the verifier will know every one of the information.
• Authentication denial is tedious and costly.

The RDIC architecture is outlined in Fig 3. There are four unique elements to be specific the cloud client, cloud server, KGC and TPA are associated with framework. The KGC produces secret keys for every one of clients with there identities. The cloud user has a large amount of files to be stored on cloud server without keeping a duplicate file and server has an critical storage space calculation assets then gives the data storage services to cloud clients. TPA has skill to uprightness of the cloud information for the benefit of the cloud client upon ask. Every element has its own commitments and advantages separately. Cloud server could act naturally interested.

4. Proposed system
The principle point of this undertaking is to actualize ID based RDIC protocol to give security to information in cloud server and zero learning protection against an third party verifier. In the proposed framework a novel development that is unique in relation to the past onces, by making
utilization of the possibility of another primitive called ID-based RDIC convention. The RDIC show with open absolute status allow anybody to review the integrity of the outsourced information, here lets accept that the third party verifier who has the ability and capability to do the conformation of work.

The ID-based RDIC design contains four substances specifically the Trusted key distribution center KGC, cloud server, cloud user, TPA are included in the framework. KGC creates secret keys for every user with there unique identites. The cloud client has a lot of files to be keep on the server without keeping duplicate file, and the cloud server has significant storage space, calculation assets then gives data storage service to the cloud clients. The TPAs work is to play out the information respectability keeping an eye for benefit the cloud client.

Fig 4 is based on the existing system where there are Four distinct elements specifically TPA, KGC, cloud user, cloud server are associated with framework. In this proposed model the files that are stored in the server cluster is divided into blocks. Instead of the cloud user it is the task of TPA to perform the data integrity checking and in this proposed model the TPA will not get the whole data of the users.

5. Modules

• User Key Generation
• Data Upload and Block Split
• Data Auditing and Integrity Checking
• File Download and Recovery

5.1. User Key Generation
First the user have to register themselves during the registration process at the site. For this procedure the user will give their very own data and the data will store in the servers database. Now the KGC will generates the secret key to users according to their identities (e.g. name, mail-id, contact number etc.). User gets the provenance to access this application once the key received from KGC.

5.2. Data Upload and Block Split
User transfer the files to the server the transferred files are saved into the cloud server. At the point when the user going to transfer the file in server the information will be encrypted and after that encrypted information will be encode and now at last encoded record will be splitted into various blocks utilizing dynamic block generation and signatures are stored in file system.

5.3. Data Auditing and Integrity Checking
Auditor performs Remote Data Integrity Checking on Cloud Data. Client send request to auditor to confirm the honesty of the information. Then auditor audits the file by block checking one by one. In Block Checking step: Two signatures are created for Block level Checking.

• From a File System a signature of a block recovers.
• To audit the block a new signature will be generate.

The above two signatures are cross checked for Block level Integrity Checking.
5.4. File Download and Recovery
In the event that clients need to download their file, at that point they will send request to cloud, now the cloud will perform block level signature checking process and the block contents are added and requested file will be downloaded in clients framework. Attackers can degenerate data in any of the clients information in cloud. On Data Integrity Checking done by the auditor, The recover process is done automatically when the data gets corrupted . To give the access confidentiality in the cloud blocks will be reallocated and the file system will be updated.

6. Figures and figure caption
The figures and figure captions are given below:

**Figure 1.** The architecture of cloud storage service

**Figure 2.** Cloud storage service architecture.

**Figure 3.** Existing Architecture

**Figure 4.** Proposed Architecture.
References

[1] Cong Wang, Student Member, IEEE, Sherman S.-M. Chow, Qian Wang, Student Member, IEEE, Kui Ren, Member, IEEE, and Wenjing Lou, Member, IEEE Privacy-Preserving Public Auditing for Secure Cloud Storage

[2] Michael Backes1,2, Dario Fiore1, and Raphael M. Reischuk2 1 Max Planck Institute for Software Systems (MPI-SWS) Verifiable Delegation of Computation on Outsourced Data

[3] Giuseppe Ateniese Sapienza, University of Rome ateniese@di.uniroma1.it Leakage-Resilient Identification Schemes from Zero-Knowledge Proofs of Storage

[4] V. Goutham, PhD HOD Dept. of CSE TKREC, JNTUH T.S, India B. Mounika Assistant Professor, Dept. of CSE TKREC, JNTUH T.S, India P. Shiva Datta M.TECH, Dept. of CSE TKREC, JNTUH T.S, India Enabling Cloud Storage Auditing with Key Exposure Resistance

[5] Qian Wang, Student Member, IEEE, Cong Wang, Student Member, IEEE, Kui Ren, Member, IEEE, Wenjing Lou, Senior Member, IEEE, and Jin Li Enabling Public Auditability and Data Dynamics for Storage Security in Cloud Computing

[6] Fiore, D., Gennaro, R. and Pastro, V., 2014, November. Efficiently verifiable computation on encrypted data. In Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security (pp. 844-855). ACM.

[7] Barsoum, A.F. and Hasan, M.A., 2015. Provable multicopy dynamic data possession in cloud computing systems. IEEE Transactions on Information Forensics and Security, 10(3), pp.485-497.

[8] Zhu, Y., Ahn, G.J., Hu, H., Yau, S.S., An, H.G. and Hu, C.J., 2013. Dynamic audit services for outsourced storages in clouds. IEEE Transactions on Services Computing, 6(2), pp.227-238.

[9] Liu, J., Huang, K., Rong, H., Wang, H. and Xian, M., 2015. Privacy-preserving public auditing for regenerating-code-based cloud storage. IEEE Transactions on Information Forensics and Security, 10(7), pp.1513-1528.

[10] Yu, Y., Au, M.H., Ateniese, G., Huang, X., Susilo, W., Dai, Y. and Min, G., 2017. Identity-based remote data integrity checking with perfect data privacy preserving for cloud storage. IEEE Transactions on Information Forensics and Security, 12(4), pp.767-778.

[11] Zhu, Y., Wang, H., Hu, Z., Ahn, G.J., Hu, H. and Yau, S.S., 2011, March. Dynamic audit services for integrity verification of outsourced storages in clouds. In Proceedings of the 2011 ACM Symposium on Applied Computing (pp. 1550-1557). ACM.