ISSN:0975-9646

Hash Based Approach for Providing Privacy and Integrity in Cloud Data Storage using Digital Signatures

N Gowtham Kumar[#1], K Praveen Kumar Rao[#2]

[#1]*Assistant Professor, Department of Computer Science, KITS,Singapuram , Karimnagar,Telangana,India*

[#2]*Associate Professor & Head, Department of Computer Science , KITS, Singapuram , Karimnagar,Telangana,India*

Abstract— **Cloud computing offers a prominent service for data storage and it refers to a broad set of technologies, policies and controls deployed to provide security for data and applications. As more businesses move to the cloud, security is a leading concern in the cloud environment, it's essential that companies work with partners that understand best practices of cloud security and provide transparency when it comes to their solutions. Service provider in cloud environment capable to controls the computing resources, the monitoring authority ensures the data integrity over out sourced data. In this paper we proposed hash based mechanism to provide privacy and integrity of outsourced data in cloud environment using private key security method.**

.

Keywords— Cloud Storage, TPP, Signature, Data Integrity.

I. INTRODUCTION

Cloud computing offer integration of web services and data centers such as distributed computing style. Major cloud service providers such as Microsoft, Amazon, Google, Yahoo, and others are offering cloud computing services. In 2002, Amazon web services were first to provide architecture for cloud based services and after that advancements and new models for cloud architecture had been proposed and developed.

Today there are many techniques of storing data on server storage to ensure client in terms of Integrity, Confidentiality, and Availability of data provided by cloud service providers. Integrity is an extent of confidence that what information is available in cloud, what is actually there, and is protected against accidental or intentional alteration without authorization. Confidentiality refers to keep the information from the out of hands. As well as legal protection, confidentiality is supported by technical tools such as access control and encryption. Availability means being able to use the system as predictable by cloud user.

Using widespread internet-enabled access, Cloud technologies can increase availability but the client is dependent on the timely and robust provision of resources. As well as well-defined contracts and terms of agreement, availability is supported by capacity building and good architecture by the cloud provider.

The system architecture includes three elements, one is cloud user has large amount of data files to be stored in the cloud, next cloud server is managed by the Cloud Service Provider to provide data storage service and has significant storage space and computation resources and finally Trusted Third Party (TPP) has expertise and capabilities that Cloud Users do not have and is trusted to evaluate the cloud storage service reliability on behalf of the user upon request. The basic cloud storage environment represented as shown in Figure1 [1]:

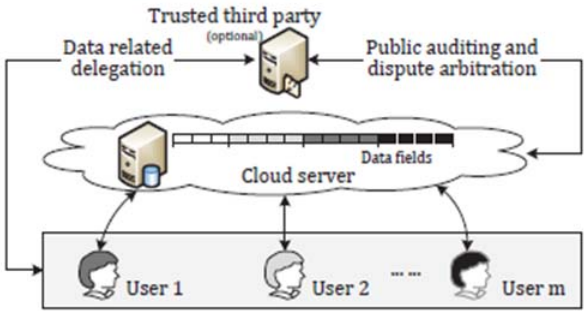

Figure 1: Cloud Storage Environment

Users can remotely store their data and enjoy the pull based high-quality applications and services from a shared pool of configurable computing resources using cloud storage, without the burden of local data maintenance and storage. However, the fact that users no longer have physical possession of the outsourced data makes the data integrity protection in cloud computing a difficult task, especially for users with inhibited computing resources. In addition, without worrying about the need to verify its integrity, users should be able to just use the cloud storage as if it is local. Consequently, enabling public auditability for cloud storage is of critical importance so that users can resort to a trusted third-party(TPP) to verify the integrity of outsourced data and be worry free. The auditing process should bring in no new vulnerabilities toward user data privacy and introduce no additional online burden to user.

Cloud data storage security focuses on the need of enforcing selective data access by providing an approach that supports the user in specification of security measures and access restrictions. Cloud storage specifies the storage on cloud with almost backup option and inexpensive storage for small enterprise solutions. The actual storage location may be on single storage environment or replicated

to multiple server storage based on importance of data. In general, typical cloud storage system architecture includes various clients and a master control server. The mechanism model of cloud storage consists of four layers: storage layer (which stores the data), basic management layer (which ensures security and stability of cloud storage itself), application interface layer (which provides application service platform), and access layer (which provides the access platform).

In cloud computing users no longer physically possess the storage of their data so classical cryptographic primitives for the purpose of data security protection cannot be directly adopted. Especially, integrity verification is not a practical solution due to the expensiveness in I/O and transmission cost across the network for downloading all the data. Further, it is often insufficient to detect the data corruption only when accessing the data, as it might be too late to recover the data loss or damage and does not data integrity to users for those un accessed data. The tasks of auditing the data correctness in a cloud environment can be expensive for the cloud users in case the large size of the outsourced data and the user's constrained resource capability. In addition, a user does not need to perform too many operations to use or retrieve the data, so that overhead of using cloud storage should be minimized as much as possible. In particular, users may not want to go through the complexity in verifying the data integrity. For easier management, it is desirable that cloud only entertains verification request from a single designated party.

II. Related Work

Organizations use the Cloud in a variety of different service models such as SaaS, PaaS, and IaaS and deployment models like Private, Public, and Hybrid.

There is number of security issues associated with cloud computing but these issues fall into two broad categories: security issues faced by cloud providers and security issues faced by their customers [3]. However, the responsibility goes both ways, the provider make sure that their infrastructure is secure and that their clients' data and applications are protected, the user must ensure that the provider has taken the proper security measures at the same time to protect their information. The user must take measures to use strong passwords and authentication measures.

Integrity Checking Functionality

One of the most important and difficult task in cloud security is maintaining data integrity. Cloud users, they are using cloud services provided by the cloud provider [2], in case of maintaining the integrity of the data, user cannot trust the service provider to handle the data, as provider can modify the original data and the integrity may be lost. In some cases, If a smart hacker hacks the cloud server and steals the data and modifies it then, this modification is not even identified by the cloud provider. So, in this case, with the help of a trusted third party makes sure that the data integrity is maintained.

Another important functionality about cloud storage is the function of integrity checking. The user no longer possesses the data at hand after a user stores data into the storage system. The user may want to check whether the data are properly stored in cloud storage servers or not. The idea of provable data possession [4], [5] and the perception of proof of storage [5], [7], [8] are proposed. Further, public auditability of stored data is addressed in [8]. Nevertheless all of them consider the messages in the clear text form. The procedure of integrity checking can be seen as a key's proficiency within the software, platform, and infrastructure security focus area of our cloud architecture. H.shacham and B.waters proposes a vision for helping assure ongoing system integrity in a virtualized environment includes an evolution of integrity checking competences [9]. It provides an increasing level of assurance and relies on secure startup enabled each phase in this evolution. It begins with one-time integrity checks at system or hypervisor startup, progresses to more frequent periodic integrity checks, and terminates in runtime integrity checks. In a classical computing environment, increasing restart frequency is very difficult because applications are tied to physical servers; restarting a production server can result an undesirable application downtime [9].

Online integrity checks help to identify and in some cases make good progress from integrity violations. Instead of performing checks for Integrity, some systems employ preventive methods to reduce the likelihood of an integrity violation [10]. The integrity assurance mechanisms divide into three main types, First, those that perform defensive steps so as to avoid exact types of integrity damages; Second, those that perform integrity checks and detect integrity violations and Finally those that is skilled of improving from loss once a violation is detected. The check summing techniques helps in detecting data integrity violations. These techniques cannot help recovery for two reasons [12]. First, a mismatch between the stored value and the computed value of the checksums just identify one of them was modified, but it cannot provide information about which of them is legitimate [2]. Further, stored checksums are also likely to be modified or corrupted. Next, checksums are generally computed using a one-way hash function and the data cannot be reconstructed given a checksum value.

Evaluation of Various Algorithms

The evaluation of different conventional key encryption algorithms, public-key key encryption algorithms and Digital Signature algorithms are studied based on previous researches and different resources. The conventional encryption algorithms studied are AES, DES, 3-DES, IDEA, Blowfish and RC5. These algorithms compared [13], [14], [15] based on attributes such as block size, key length, cipher text, cryptanalysis resistance, possible keys, possible ASCII printable character keys. From above study, that AES encryption algorithm is faster, more efficient, and superior in terms of time consumption (encryption/decryption) and throughput under the scenario of data transfer. So it would be better to use AES scheme in encryption of data stored at other end and need to decrypt

multiple time. On the other hand, public key encryption algorithms studied are RSA and Elliptic Curve Cryptography. These algorithms are compared [16] based on main attribute key size with various features such as private key generation time, signature generation time and signature verification time are calculated.

It was difficult to state which of asymmetric encryption algorithm is better because RSA performs better when there is no need to generate RSA keys for each use, but rather have fixed RSA keys. Using RSA, signature generation and signature verification time is also much less than ECC. However, ECC scores over RSA because of less key generation time [17]. When lot of users connects to cloud based services with small session time like cloud based storage, ECC is better option. To achieve non-repudiation and authentication purpose within cloud computing environment, digital signature has assumed great significance. There are no of digital signature algorithms which involves the generation of message digest (hash function). The study on various hash algorithms shows that MD5 is much faster than SHA-512 digital signature algorithm, but with respect to security concerns SHA-512 is more secure than MD5 and so far, no claim of successful attacks with optimal time complexity on SHA-512 has been done [16]. The study of various encryption algorithms and digital signature algorithms helps to choose the best one from each category to be used in proposed cryptographic module. The algorithms such as AES and ECC are used under asymmetric encryption algorithms respectively. The digital signature generation algorithm [SHA-512] is used in combination with ECC asymmetric key encryption algorithm.

III Assumptions And Methods Evaluation

Digest ID Generation using Hash Function

The hash function used for digest ID generation has one fundamental requirement, because it has a low collision probability of digest IDs. Even under a "perfect" hashing assumption that hash values are independent and identically distributed over a digest of m bits, the hashes of two distinct packets will collide with probability $p = 1/2^m$. In practice the collision rate may be higher than this due to non-uniformity of the hash distribution. To be specific, consider the case of One Way Delay (OWD) estimation. In [18] the expected proportion of reports pairs from the same packet that suffer a digest collision with a report on another packet that prevents correct estimation of OWD is bounded above by: $P(\text{wrong\_OWD}) = (2/3)\,[1-(1-p)^{RT}]$ (3)

Here, T is an upper bound on the time between receipt of reports on the same packet at the DC, and R an upper bound on the rate at which reports reach the DC. Hence, RT is an upper bound for the maximum number of packets that reach the collector between reports of a given packet, and hence $1-(1-p)^{RT}$ is an upper bound for the probability that a given packet has a hash collision. The factor 2/3 arises since not all orderings of report arrivals from packets with colliding digests actually give rise to an incorrect determination of the OWD. Computation speed is still a requirement for the hash function used for digest ID generation, but this requirement is less stringent than with the hash function used for packet selection, because the digest ID must be computed only on the packets that are selected during sampling. Hash collisions can be identified by the presence of collisions in hashes reported from measurement points located at the network ingress points. In this case, all colliding reports may be discarded. If export to the DC is unreliable, it is still possible to identify duplicates if MPs at ingress provide additional information on the set of packets selected [19].

Elliptic Curve Cryptography (ECC) with SHA-512:

An elliptic curve is given by an equation in the form of The finite fields those are commonly used over primes (FP) and binary field (F2n). The security of ECC is based on the elliptic curve discrete logarithm problem (ECDCP) and this problem is defined as, Given point M, N on elliptic curve, find z such that M=zN. To work with ECC the above steps are required with SHA-512 [9] [10].

ECC key generation**: For** generate a public and private key pair used in ECC communication the following steps are required:

1. Find an elliptic curve E(K), where K =finite field such as F2n or FP and a find point Q on E(K)(n is the order of Q).
2. Select a pseudo random number e such that $1 \le e \le (n - 1)$.
3. Compute point P = e*Q.
4. ECC key pair is (P, e), where P is public key and e is private key.

To create a signature S for message m, using ECC key pair (P, K) over E(k), the following steps are required:

1. Generate a random number k such that $1 \le k \le (n - 1)$.
2. Compute point kQ = (x1, y1).
3. Compute r = x1 (mod n). If r = 0, go to step 1.
4. Compute k-1 (mod n).
5. Compute SHA-512(m), and convert this to an integer e.
6. Compute s = k-1(e + xr) (mod n). If s = 0, go to step 1.
7. The signature for message m is S = (r, s).

To verify a signature s=(r,s) for message m over a curve E(k) using the public key P performing steps:

1. Verify r and s are integers over the interval [1, n - 1].
2. Compute SHA-512(m) and convert this to an integer e.
3. Compute w = s-1(mod n).
4. Compute u1 = ew (mod n) and u2 = rw (mod n).
5. Compute X = u1Q + u2P
6. If X = 0, reject S. Otherwise, compute v = x1 (mod n).
7. Accept if and only if v = r.

IV Proposed Scheme

Mathematical Model:

Let $R = \{R_1, R_2 \ldots R_N\}$ be the set of files records stored on cloud storage. Let the number of records to be power of two, so that we denote possible number of messages as $N = 2^m$.

Here the problem using N of Records to build a Merkle Hash Tree based data integrity scheme, so that maximum number of records are consecutively links one with other.

Next, To calculate result parameter which is Effective Bandwidth , the following formula is used
Let W = Effective bandwidth.
L= size of all records to be secure.
T = Time Taken for performing all operation that are part of cryptographic algorithms.
m= Number of users
DIt = Time required for achieving Data Integrity using Merkle Hash Tree.
Then It can given by , B = L /T

Achieve data integrity:
Step 1: Obtain the total number of records in one folder. It is require to ensure that the number of records are $N=2^m$. The users then stores the data records at the cloud server, publishes the verification metadata to TPP for later audit and also delete them. As part of pre-processing, the user may modify the data record by expanding or including additional metadata to it to be stored at server. The cloud user runs Key Generation to generate the system's public and secret parameters.

Step 2: Key Generation:
Generate public key k1 to generate public keys PUi and private key PRi. For every public key PUi with 1<= i <= 2m, a hash value hi=H (PUi) is computed. Using these hash values hi a hash tree is to be built. Assume a node of the tree x_{ij}, where i denote the level of the node. Every level of a node in a tree is defined by the distance from the node to a leaf. Assign a leaf of the tree has level i=0 and root has level i=m. then try to number all nodes of one level from left to right, so that x_i,0 is leftmost node of level. In Merkle tree the hash values hi are the leaves of a binary tree, therefore hi=a0,i. In the tree every inner node is the hash value of the concatenation of its two children. So a1,0=H (x0,0||x0,1) and x2,0=H (x1,0||x1,1).
In this way, a tree with 2n leaves and 2n+1 -1 nodes are built. The root of the tree an X, 0 is the public key pu_k of the Merkle Signature Scheme.

Step 3: Signature Generation
To provide a signature to a message M with Merkle signature scheme, the message is signed with one time signature scheme, resulting in a signature sign' first. It can be done by using one of the public & private key pairs (PUi, PRi). The corresponding leaf of the hash tree to one time public key PUi is x0, i=H(PUi) .Again, call the path in the hash tree from a0,I to the root A. The path X consist of n+1 nodes,X0,….Xn, with X0=x0,i being the leaf and Xn=xn,0 is public key pu_k being the root of the tree. To compute this path A, need every child of the nodes A_1,…., An, since Ai is a child of Ai+1.
To calculate the next node Ai+1 of the path A, there need to know both children of Ai+1. Hence it is required the neighbour node of Ai and call this node Dig_i. So that $A_{i+1}=H(A_i || Dig_i)$ Hence n nodes Dig0 …. Dig_{n-1} are needed to compute every node of the path A. Now calculate and save these nodes Dig0 …. Dig_{n-1} .These nodes plus one time signature sign' of M is the signature sign=(sign'|| Dig_0|| Dig_1||.....|| Dig_{n-1})

In the audit phase signature is verified. Next the process of decryption will assure that a hash value that will be compared along with the hash value that the cloud trusted third party compute it in its part. After finishing the verification, the TPP will inform the user if the CS was trusted or not.

Step 4: Signature verification
The receiver knows the public key pu_k, message M and signature sign'. Then the receiver verifies one time signature sign' of the message M. If sign' is a valid signature of M the receiver computes $A_0=H(PUi)$ by hashing the public key of one time signature. For j=1,2…..n-1 the nodes of A_j of the path A are computed with $A_j=H(A_{j-1} || Dig_{j-1})$. If A_n equals the public key pu_k of signature tree, signature is valid.

CONCLUSIONS

Cloud Computing has been envisioned as the next generation architecture of **IT** Enterprise There are many issues in cloud computing, one of them is integrity of data. Due to this issue, many users are worried of using cloud technology as security of their data is not guaranteed. Before this various frameworks have been proposed in order to resolve this issue but no framework had provided full security. In this paper proposed signature scheme to resolve the issue of integrity of user data with better performance using the traditional algorithms of network security in cloud storage. Cost is also optimized using multi-cloud concept and different platforms for various categories of the users.

REFERENCES

[1] Hong Liu, Huansheng Ning, Qingxu Xiong and Laurence T. Yang,"Shared Authority Based Privacy-preserving Authentication Protocol in Cloud Computing" IEEE TRANSACTIONS ON Parallel and Distributed Systems, VOL. PP, NO. 99, 25 February 2014
[2] P.Mell and T,Grance, "Draft NIST working definition of cloud computing", referred on June 3rd 2009.
[3] Yashpalsing Jadeja, Kirit Modi, "Cloud Computing -Concepts, Architecture and Challenges", 2012 International Conference on Computing, Electronics and Electrical Technologies [ICCEET], pages 877-880, 2012
[4] G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson, and D. Song, "Provable Data Possession at Untrusted Stores," Proc. 14th ACM Conf. Computer and Comm. Security (CCS), pp. 598-609, 2007.
[5] G. Ateniese, R.D. Pietro, L.V. Mancini, and G. Tsudik, "Scalable and Efficient Provable Data Possession," Proc. Fourth Int'l Conf. Security and Privacy in Comm. Netowrks (SecureComm), pp. 1-10, 2008.
[6] H. Shacham and B. Waters, "Compact Proofs of Retrievability," Proc. 14th Int'l Conf. Theory and Application of Cryptology and Information Security (ASIACRYPT), pp. 90-107, 2008.
[7] G. Ateniese, S. Kamara, and J. Katz, "Proofs of Storage from Homomorphic Identification Protocols," Proc. 15th Int'l Conf. Theory and Application of Cryptology and Information Security (ASIACRYPT), pp. 319-333, 2009.
[8] K.D. Bowers, A. Juels, and A. Oprea, "HAIL: A High-Availability and Integrity Layer for Cloud Storage," Proc. 16th ACM Conf. Computer and Comm. Security (CCS), pp. 187-198, 2009.
[9] H.shacham and B .Waters "compact proofs of retrivability "in proc. Of asiascrypt 2008
[10] Q.Wang ,C.Wang, j.Li, K.Ren, and W.lou, "Enabling public verifiability and data dynamics for storage security in cloud computing".

[11] Joel Gibson, Darren Eveleigh, Robin Rondeau, Qing Tan, "Benefits and Challenges of Three Cloud Computing Service Models", 978-1-4673-4794-5/12/$31.00_c 2012 IEEE"
[12] M.AShah,R.Swaminathan, and M.Baker"privacy-preserving audit and extraction of digital contents".
[13] D. S. Abdul. Elminaam, H. M. Abdul Kader and M. M. Hadhoud, "Performance Evaluation of Symmetric Encryption Algorithms" Vol. 8, 2009, p. 58-64.
[14] S.Hirani, "Energy Consumption of Encryption Schemes in Wireless Devices Thesis," university of Pittsburgh, April 9, 2003.
[15] "A Performance Comparison of Data Encryption Algorithms," IEEE [Information and Communication Technologies, 2005. ICICT 2005. First International Conference ,2006-02-27, P. 84- 89.
[16] Nicholas Jansma, Brandon Arrendond, "Performance Comparison of Elliptic Curve and RSA Digital Signatures" April, 2004.
[17] Veerraju Gampala, Srilakshmi Inuganti, Satish Muppidi, "Data Security in Cloud Computing with Elliptic Curve Cryptography" vol. 2 Issue 3, July 2012.
[18] S.Niccolini, M.Molina, S.Tartarelli, F.Raspall "Design and implementation of a One Way Delay passive measurement system" – IEEE Network Operations and Management Symposium 2004, Korea, Apr. 2004
[19] N.G. Duffield, M.Grossglauser, "Trajectory Sampling with Unreliable Reporting", IEEE Infocom 2004, Hong Kong, March 7-11, 2004.