Volume 4, Issue 1, January 2014 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

Dynamic Auditing Protocol using Improved RSA and CBDH for Cloud Data Storage

S.Hemalatha
PhD Research Scholar
Research Department of computer Science
N.G.M College, Pollachi, India

Dr. R.Manickachezian
Associate Professor
Research Department of Computer Science
N.G.M College, Pollachi, India

Abstract - In the current era of cloud computing, Cloud storage is a significant service of cloud computing where data stored in the cloud is being generated at high speed and thus the cloud storage system has become one of the key components in cloud computing. Data owner host their data and data consumers can use the data on and from the server. The security issue arises due to the data outsourcing. This paper proposes a new paradigm for security, an independent auditing service to verify the data integrity. Some existing methods are used only for static archive data. It cannot be applied for data updated dynamically. In this paper, an auditing architecture for cloud storage system is designed and an efficient and privacy preserving protocol is proposed to check the data is correctly stored in the cloud. Improved RSA is used in this paper for cryptography.

Keywords— Cryptography,Encryption,Privacy preserving,Dynamic Auditing,Batch Auditing.

I. Introduction

Cloud storage is very important service of cloud computing, that allow data owners to maneuver data from their native computing systems to the Cloud. Cloud storage becomes an increasing attraction in cloud computing paradigm,allows an user to store their data and access them where and whenever they go as pay-as-you-go manner with plenty and plenty {of data|of knowledge of information}and so the lots of owners begin selecting to host their data inside the Cloud [1]. Sometimes, cloud service suppliers can be dishonest.They might discard the data that has not been accessed or rarely accessed { to save|to avoid wasting} lots of domicile for storing and claim that the data area unit still properly hold on inside the cloud. Therefore, house owners have to be compelled to be convinced that the data are properly hold on inside the cloud. The pc code methodology is chiefly applicable for skinny users who have less resources and restricted computing capability. It satisfies all the security and performance wants of cloud data storage. Coding system methodology to boot supports public verifiability that allows TPA to verify the integrity of information whereas not retrieving original data from the server and likelihood detects data corruptions.To understand economical data dynamics, improve the prevailing proof of storage models by manipulating the classic Merkle Hash Tree construction for block tag authentication. To support economical usage of multiple auditing tasks, a tendency to explore the technique of additive mixture signature to extend proposed main result into a multiuser setting, wherever TPA can perform multiple auditing tasks at a similar time [2]. The TPA, World Health Organization has experience and capabilities that users don't, will sporadically check the integrity of all the data hold on inside the cloud on behalf of the users, that has plenty of easier and affordable method for the users to verify their storage correctness within the cloud. Moreover, additionally to assist users to gauge the chance of their signed cloud data services, the audit result from TPA would even be helpful for the cloud service suppliers to enhance their cloud-based service platform, and even serve for freelance arbitration purposes [3] [4]. A construction of dynamic audit services for un-trusted and outsourced storage. The efficient methodology for periodic sampling audit is to minimize the computation prices of third party auditors and storage service suppliers with the survey of current cloud storage suppliers. Dynamic audit experiments showed that proposed resolution includes a tiny, constant quantity of overhead, that minimizes computation and communication prices [5]. A dynamic audit service is for proving the integrity of associate degree untrusted and outsourced storage. Dynamic audit service is made supported by the techniques, fragment structure, sampling, and index-hash table, supporting demonstrable updates to outsourced data and timely anomaly detection [5] [6].Most of the schemes that use RSA primarily based verification however the key length for secure RSA use as magnified over recent years and this place a heavier process burden on applications exploitation RSA. To avoid this drawback, code primarily based verification scheme proposed [6].

Owners will check the data integrity supported two-party storage auditing protocols. In cloud Storage system, however, it's inappropriate to let either aspect of cloud service providers or owners conduct such Auditing, as a result none of them can be bound to give unbiased auditing result. In this state of affairs, third party auditing could be a natural selection for the storage auditing in cloud computing [7] [8].Provable data possession (PDP), that could be a science technique for verifying the integrity of information while not retrieving it at an untrusted server, are often used to

understand audit services. Provable data possession" (PDP) model is for guaranteeing possession of data files on untrusted storages. They utilize the RSA-based homomorphic linear authenticators for auditing outsourced information and recommend willy-nilly sampling some blocks of the file [9]. Recently, the importance of guaranteeing the remote information integrity has been highlighted by the subsequent analysis works underneath completely different system and security models [10] These techniques, whereas are often helpful to confirm the storage correctness while not having users possessing native information, area unit all focusing on single server situation. In [11], a good and versatile distributed theme with express dynamic information support, as well as block update, delete, and append. There is a tendency to estimate erasure-correcting code within the file distribution preparation to produce redundancy parity vectors and guarantee the data responsibleness. By utilizing the homomorphic token with distributed verification of era positive coded information, proposed theme achieves the combination of storage correctness insurance and information error localization, i.e., whenever data corruption has been detected throughout the storage correctness verification across the distributed servers of the misbehaving server(s). An economical and secure dynamic auditing protocol [12], proposed to unravel the information privacy drawback. To unravel the information privacy drawback, economical and secure dynamic auditing protocol is to come up with an encrypted proof with the challenge stamp by victimization.The Bilinearity property of the linear pairing, such the auditor will not decode it however can verify the correctness of the proof. While not victimization the mask technique, proposed methodology doesn't need any trusty organizer throughout the batch auditing for multiple clouds.

II. Related Work

Shingare Vidya Marshal [2], proposed an efficient and essentially secure dynamic auditing system to achieve guarantee of information integrity and availability in cloud and implement the value system for user by using TPA. Therefore client will trust on cloud storage service that is provided by cloud as a result of TPA works as a representative of data owner. In [3], the authors proposed a flexible distributed storage integrity auditing mechanism, utilizing the homomorphic token and distributed erasure-coded data. The proposed design allows users to audit the cloud storage with terribly light-weight communication and computation value. The auditing result not only ensures strong cloud storage correctness guarantee, however additionally at the same time achieves quick data error localization, i.e., the identification of misbehaving server. Yan Zhu [11] proposed a dynamic audit service for confirming the integrity of associate untrusted and outsourced storage. A dynamic audit service is built supported the techniques, fragment structure, sampling, and index-hash table, supporting obvious updates to outsourced and timely anomaly detection.Rakhi Bhardwaj, Vikas Maral [4] planned a dynamic audit service for un-trusted and outsourced storage. This technique additionally given Associate in Nursing efficient method for periodic sampling audit to minimize the computation prices of third party auditors and storage service providers with the survey of current cloud storage suppliers. A dynamic audit service incorporates a little, constant quantity of overhead, that minimizes computation and communication prices.

Qian Wang [5], proposed Third party auditor (TPA) to explore the matter of providing coincident public auditability and knowledge dynamics for remote proposed integrity sign on Cloud Computing. TPA eliminates the involvement of the consumer through the auditing of whether or not his data keep within the cloud square measure so intact, which may be vital in achieving economies of scale for Cloud Computing. Cong Wang [6], proposed privacy-preserving public auditing for cloud data storage below the said model. To attain privacy-preserving public auditing, the author planned to unambiguously integrate the homomorphic linear appraiser with random masking technique. During this planned TPA protocol, the linear combination of sampled blocks within the server's response is covert with randomness generated by the server. Most of the schemes that use RSA based verification, however the key length for secure RSA use as magnified over recent years and this put a heavier process burden on applications using RSA. To avoid this proposed, code based verification scheme [6]. Owners will check the information integrity supported two-party storage auditing protocols. In cloud Storage system, however, it's inappropriate to let either facet of cloud service providers or homeowners conduct such Auditing, as a result of none of them may be bound to offer unbiased auditing result. In this scenario, third party auditing may be a natural selection for the storage auditing in cloud computing [7] [8].Kan rule [12] planned an efficient and inherently secure dynamic auditing protocol. It protects the information privacy against the auditor by combining the cryptography technique with the property of bilinear paring, instead of using the mask technique. Thus, proposed multi-cloud batch auditing protocol doesn't need any additional organizer. The batch auditing protocol may support the batch auditing for multiple owners.

III. Proposed System

III-A System model

The cloud storage model consists of three main components as illustrated in Fig. 1.

1) **Cloud Owner:** the owner, who can be an individual or an organization originally hosting their data in cloud.
2) **Cloud Service Provider (CSP):** the CSP, who manages **cloud servers** (CSs) and provides a paid storage space on its infrastructure to users as a service and user access data from the server.
3) **Third Party Auditor (TPA) or Verifier:** the TPA or Verifier, who has expertise and capabilities that users may not have and verifies the integrity of outsourced data in cloud on behalf of users. Based on the audit result, the TPA could release an audit report to user.

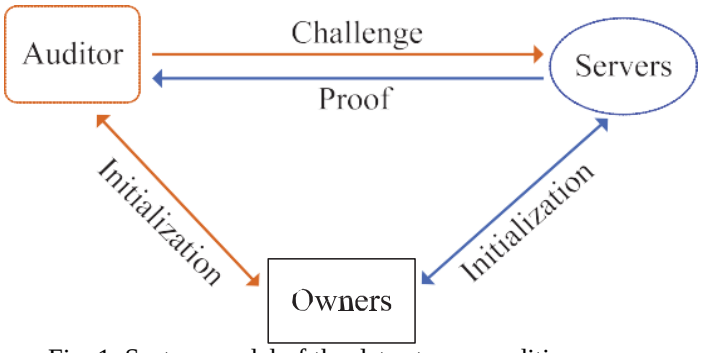

Fig. 1. System model of the data storage auditing.

Table 1 Notation

III-B. SECURITY MODEL

Server could be dishonest and may launch the subsequent attacks:

1)Replace attack: The server might choose another valid and uncorrupted pair of data block and data tag to interchange the challenged try of data block and data tag when it already discarded.

2)Forge attack. The server of data forge the data tag of information block and deceive the auditor; if the owner's secret tag keys unit of measurement reused for the varied versions of information.

3)Replay attack. The server might generate the proof from the previous proof or completely different data, with-out retrieving the particular owner's information.

III-C. STORAGE AUDITING PROTOCOL ALGORITHM

Suppose a file F has m data components as $F = (F_1, \ldots, F_n)$. Each data component has its physical meanings and can be updated dynamically by the data owners. For public data components, the data owner does not need to encrypt it, but for private data component, the data owner needs to encrypt it with its corresponding key. Each data component F_k is divided into n_k data blocks denoted as $F_k = (m_{k1}, m_{k2}, \ldots, m_{knk})$. Due to the security reason, the data block size should be restricted by the security parameter. For example, suppose the security level is set to be 160 bit (20 Byte), the data block size should be 20 Byte. A 50-KByte data component will be divided into 2,500 data blocks and generates 2,500 data tags, which incurs 50-KByte storage overhead.

By using the data fragment technique, the data blocks are further splitted into sectors. The sector size is restricted by the security parameter. One data tag is generated for each data block that consists of s sectors, such that less data tags are generated. In the same example above, a 50-KByte data component only incurs 50/s KByte storage overhead. In real storage systems, the data block size can be various. That is, different data blocks could have different number of sectors. For example, if a data block m_i will be frequently read, then s_i could be large, but for those frequently updated data blocks, si could be relatively small.

For simplicity, only one data component is considered in proposed construction and constant number of sectors for each data block. Suppose there is a data component M, which is divided into n data blocks, and each data block is further split into s sectors. For data blocks that have different number of sectors, the maximum number of sectors S_{max} among all the sector numbers s_i is selected first. Then, for each data block m_i with S_i sectors, $S_i < S_{max}$,it is simply considered that the data block m_i has S_{max} sectors by setting $m_{ij} < 0$ for $S_i < j <= S_{max}$. Because the size of each sector is constant and equal to the security parameter p, the number of data blocks can be calculated as n =sizeof(M)/ s.logp. The encrypted data component is represented as M = { m_{ij} }, $i \in [1, n]$, $j \in [1, s]$.

Let G1,G2, and GT be the multiplicative groups with the same prime order p and e G1 xG2 -> GT be the bilinear map. Let g1 and g2 be the generators of G1 and G2, respectively. Let h :{0, 1}*-> G1 be a keyed secure hash function that maps the M_{info} to a point in G1.

Computational Bilinear Diffie-Hellman(CBDH)

A group G of prime order p according to the security parameter. Let a, b, c ∈ z_b be chosen at random and g be a generator of G. When given g, g_a, g_b, g_c, the adversary must compute e(g, g)abc. An algorithm B that outputs e(g, g)abc has advantage ε in solving CBDH in G if |Pr[B(g, g_a, g_b, g_c) = e(g, g)abc]| ≥ ε.

$KeyGen(\lambda) \rightarrow (pk_t, sk_t, sk_h)$The key generation algorithm takes no input other than the implicit security parameter.

$TagGen(M, sk_t, sk_h) \rightarrow T$. The tag generation algorithm takes each data component M, the secret tag key sk_t, and the secret hash key sk_h as inputs.

$Chall(M_{info})$ The challenge algorithm takes the abstract information of the data M_{info} as the input.

$Prove(M, T, C) \rightarrow P$. The prove algorithm takes as inputs the data M and the received challenge$C = (\{i, v_i\}_{i \varepsilon q}, R)$.

III.D - CONSTRUCTION OF PROPOSED PRIVACY-PRESERVING AUDITING PROTOCOL:.

As illustrated in Figure. 2, the proposed storage auditing protocol consists of three phases: owner initialization, confirmation auditing, and sampling auditing. Throughout the system initialization, the owner generates the keys and also the tags for the information. Once storing the data on the server, the owner asks the auditor to conduct the confirmation auditing to form positive that their information is properly pair on the server. Once confirmed, the owner will the data delete the local copy of the info. Then, the auditor conducts the sampling auditing sporadically to check the info integrity.

Symbol	Physical Meaning
sk_t	Secret tag key
pk_t	Public tag key
sk_h	Secret hash key
M	Data component
T	Set of data tags
N	Number of blocks in each component
S	Number of sectors in each data block
M_{info}	Abstract information of M
E	Challenge generated by the auditor
P	Proof generated by the server

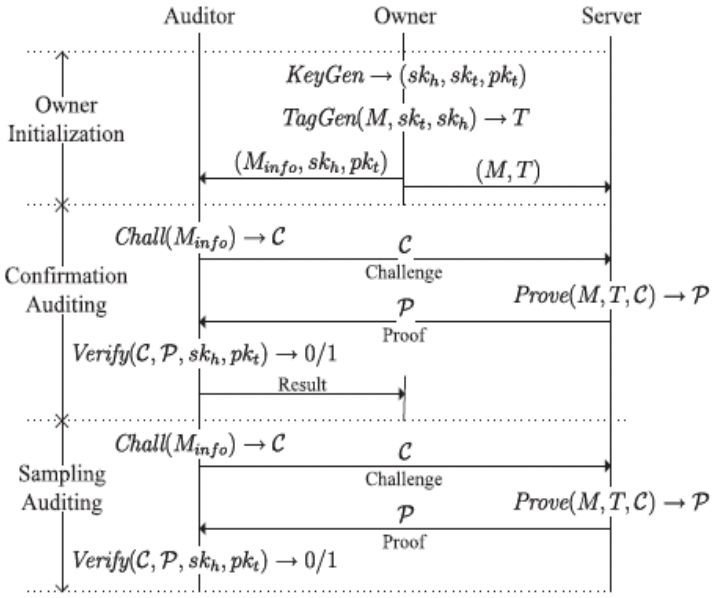

Figure 2: Framework of proposed privacy-preserving auditing protocol

Phase 1: Owner initialization. The owner runs the key generation algorithm KeyGen to get the key hash key and thus the pair of secret-public tag key.

Phase 2: Confirmation auditing. In proposed auditing construction, the auditing protocol solely involves two-way communication: Challenge and Proof. Algorithm the confirmation auditing sporadically, the owner desires the auditor to see whether or not the owner's information unit properly hold on the server.

Phase 3: Sampling auditing. The auditor will do the sampling auditing periodically by difficult a sample set of data blocks. The frequency of taking auditing operation depends on the service agreement between the data owner and thus the auditor (and additionally depends on how much trust the data owner has over the server). Just like the confirmation auditing in part 2, the sampling auditing procedure also contains two-way communication.

III.E - SECURE DYNAMIC AUDITING

In cloud storage systems, the {data /the info/ the information} owners can dynamically update their data. As an auditing service, the auditing protocol should be designed to support the dynamic data, as well as the static archive data. However, the dynamic operations could create the auditing protocols insecure. Specifically, the server could conduct 2 following attacks: 1) Replay attack. The server might not update properly the owner's data on the server and should use the previous version of the information to pass the auditing. 2) Forge attack. once the information owner updates the

information to this version, the server could get enough data from the dynamic operations to forge the information tag. If the server might forge the information tag, it will use any knowledge and its cast data tag to pass the auditing.

1. Algorithms and Constructions for Dynamic Auditing.

The dynamic auditing protocol consists of 4 phases: owner format, confirmation auditing, sampling auditing, and dynamic auditing. The primary 3 phases square measure just like proposed privacy preserving auditing protocol as represented in the above section. The only variations square measure the tag generation rule.

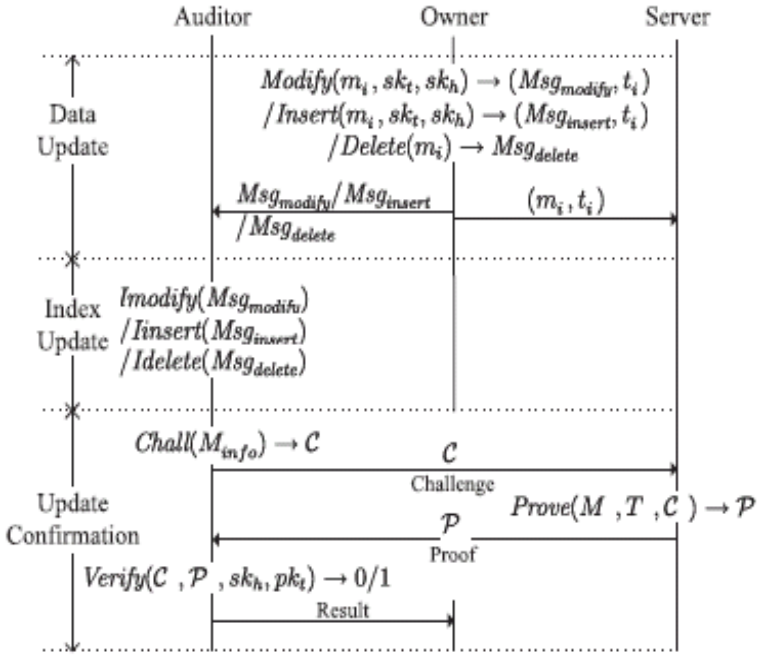

Figure: 3 Framework of auditing for dynamic operations

TagGen and also the ITable generation throughout the owner format part. Here, as illustrated in Fig. 3, the dynamic auditing part is described , which contains 3 steps: data update, index update, and update confirmation.

Step 1: knowledge update. There are three types of data update operations that may be employed by the owner: modification, insertion, and deletion. For every update operation, there is a corresponding algorithm within the dynamic auditing to process the operation and facilitate the longer term auditing, outlined as follows:

$\mathrm{Modify}(m_i^*, sk_t, sk_h) \rightarrow (\mathrm{Msg}_{\mathrm{modify}}, t_i^*)$. The modification algorithm takes as inputs the new version of data block m_i^* the secret tag key sk_t, and the secret hash key sk_h.

$\mathrm{Insert}(m_i^*, sk_t, sk_h) \rightarrow (\mathrm{Msg}_{\mathrm{insert}}, t_i^*)$. The insertion algorithm takes as inputs the new data block m_i^* the secret tag key sk_t, and the secret hash key sk_h.

$\mathrm{Delete}\ (m_i) \rightarrow \mathrm{msg}_{\mathrm{delete}}$. The deletion algorithm takes as input the data block mi. It outputs the update message $\mathrm{Msg}_{\mathrm{delete}} = (i, B_i, V_i, T_i)$.

Step 2: Index update. Upon receiving the three types of update messages, the auditor calls three corresponding algorithms to update the I Table. Each algorithm is designed as follows:

$\mathrm{IModify}(\mathrm{Msg}_{\mathrm{modify}})$.The index modification algorithm takes the update message $\mathrm{Msg}_{\mathrm{modify}}$ as input. It replaces the version number V_i by the new one V_i^* and modifies T_i by the new time stampT_i^*.

$\mathrm{IInsert}(\mathrm{Msg}_{\mathrm{insert}})$. The index insertion algorithm takes as input the update message $(\mathrm{Msg}_{\mathrm{insert}})$

$\mathrm{IDelete}\ (\mathrm{Msg}_{\mathrm{delete}})$. The index deletion algorithm takes as input the update message $\mathrm{Msg}_{\mathrm{delete}}$

Step 3: Update confirmation. Once the auditor updates the I Table, it conducts a confirmation auditing for the updated data and sends the result to the owner. Then, the owner will favor to delete the native version of data according to the update confirmation auditing result.

III.F - BATCH AUDITING FOR MULTIOWNER AND MULTICLOUD

Data storage auditing is an important service in cloud computing that helps the owners check the data integrity on the cloud servers. As a result of the massive variety of data owners, the auditor could receive many auditing requests from multiple owners. In this situation, it might greatly improve the system performance, if the auditor could combine these auditing requests along and solely conduct the batch auditing for multiple owners at the same time. The previous work [14] cannot support the batch auditing for multiple owners.As a result of parameters for generating the info tags employed by every owner is totally different, and thus, the auditor cannot combine the info tags from multiple house owners to conduct the batch auditing.On the other hand, some data owners could store their data on quite one cloud servers. To confirm the owner's data integrity altogether the clouds, the auditor can send the auditing challenges to every cloud server that hosts the owner's data and verify all the proofs from them.To cut back the computation price of the

auditor, it's very fascinating to mix of these responses together and do the batch verification.

1. Algorithms for Batch Auditing for Multiowner and Multi cloud

Let O be the set of owners and S be the set of cloud servers. The batch auditing for multi owner and multi cloud can be constructed as follows:

Phase 1: Owner initialization. Each owner $O_k(k \varepsilon O)$ runs the key generation algorithm KeyGen to generate the pair of secret-public tag key $(sk_{t,k}, pk_{t,k})$ and a set of secret hash key $\{sk_{h,kl}\}_{l\epsilon S}$

Phase 2: Batch auditing for multi owner and multi cloud. Let O_{chal} andS_{chal} denote the involved set of owners and cloud servers involved in the batch auditing, respectively. The batch auditing also consists of three steps: batch challenge, batch proof, and batch verification.

IV PERFORMANCE ANALYSIS

Storage auditing is a very difficult service in terms of computational cost, communication cost, and memory space. In this section, the communication price comparison and computation complexity comparison between proposed scheme and two existing works are described. The Audit protocol proposed by Wang et al, Zhu et al. The storage overhead analyses are going to be shown within the supplemental file, available online.

IV.A COMMUNICATION COST

The communication cost throughout the initialization is sort of an equivalent in these 3 auditing protocols; compare the communication cost between the auditor and the server, which consists of the challenge and therefore the proof. Think about a batch auditing with K owners and C cloud servers. Suppose the number of challenged data block from every owner on totally different cloud servers is that the same, denoted as t, and the data blocks are split into s sectors in Zhu's IPDP and proposed theme.The comparison under an equivalent probability of detection is done. That is, in Wang's theme, the quantity of information blocks from every owner on every cloud server ought to be set. As it is known, in large-scale cloud storage systems, the full range of information blocks could be very large. Therefore, Wang's auditing scheme could incur high communication cost.

IV.B COMPUTATION QUALITY

The computation of the owner, the server, and therefore the auditor on a Linux system with associate Intel Core, a pair of couple CPU at 3.16 ghz are simulated with appropriate RAM. The code uses the improved RSA cryptography to simulate proposed auditing scheme and Zhu's IPDP scheme (Under an equivalent detection of probability, Wang's theme needs for more information blocks than proposed scheme and Zhu's scheme, such the computation time is sort of s times over proposed theme and Zhu's IPDP, and thus, it's not comparable).

Improved RSA

Key generation

Generate k random primes $p_1,\ldots,p_k$, each $\lfloor lg(n)/kc \rfloor$ bits in size, with $gcd(p_1 - 1,\ldots,p_k - 1) = 2$, and compute $n = \prod_{i=1}^{k} pi$; Generate k random s-bit integers $dp_1,\ldots,dp_k$ such that $gcd(d_{p_1}, p_1 - 1) = \ldots = gcd(dpk, pk - 1) = 1$ and dp1 _ … _ dpk (mod 2);

Apply the CRT to obtain d such that $d \equiv dp_i (\text{mod } p_i - 1)$ for $1 \leq i \leq k$

Calculate $e = d^{-1} \text{ mod } \emptyset(n)$

The public key is $\langle n, e \rangle$, while the private key is $\langle p_1,\ldots,p_k, dp_1,\ldots,dp_k \rangle$

This is more clearly stated as: solve for d given d· e ≡ 1 (mod φ(n))

This is often computed using the manhattan algorithm. d is kept as the private key exponent.

Encryption

The term $e = O(n)$ is used instead of $O(1)$ as in plain RSA, leading to more costly public-key operations.

$C = m^e (mod\ n)$

The improved RSA scheme and Zhu's IPDP have the same total communication cost during the challenge phase. During the proof phase, the communication cost of the proof in proposed scheme is only linear to C, but in Zhu's IPDP, the communication cost of the proof is not only linear to C and K, but also linear to s. That is because Zhu's IPDP uses the mask technique to protect the data privacy, which requires sending both the masked proof and the encrypted mask to the auditor. In proposed scheme, the server is only required to send the encrypted proof to the auditor and, thus, incurs less communication cost than Zhu's IPDP.

The elliptic curve is used to tend MNT d159 curve, wherever the bottom field size is 159 bit and therefore the embedding degree is 6. The d159 curve contains a 160-bit cluster order, which implies p could be a 160-bit length prime. All the simulation results are the mean of 20 trials.

1. COMPUTATION COST OF THE AUDITOR

The computation time of the auditor versus the number of data blocks, the number of clouds, and the number of owners are compared as shown in Fig.4. Fig. 4a shows the computation time of the auditor versus the number of

challenged data blocks in the single cloud and single owner case. In this figure, the number of data blocks goes to 500 (i.e., the challenged data size equals to 500 KByte), but it can illustrate the linear relationship between the computation cost of the auditor versus the challenged data size. From Fig. 4a, it is shown that the proposed scheme incurs less computation cost of the auditor than Zhu's IPDP scheme, when coping with large number of challenged data blocks. In real cloud storage systems, the data size is very large,(e.g., petabytes).The proposed scheme apply the sampling auditing method to ensure the integrity of such large data.

The sample size and the frequency are determined by the service-level agreement. From the simulation results, it is estimated that it requires 800 seconds to audit for 1-GByte data. However, the computing abilities of the cloud server and the auditor are much more powerful than proposed simulation PC, so the computation time can be relatively small. Therefore, proposed auditing scheme is practical in large-scale cloud storage systems. Fig. 4b describes the computation cost of the auditor of the multi cloud batch auditing scheme versus the number of challenged clouds.

It is easy to find that proposed scheme incurs less computation cost of the auditor than Zhu's IPDP scheme, especially when there are a large number of clouds in the large-scale cloud storage systems. Because Zhu's IPDP does not support the batch auditing for multiple owners, in this simulation, the computation is repeated for several times that is equal to the number of data owners. Then, as shown in Fig. 4c, the computation cost of the auditor between the multi owner batch auditing and the general auditing protocol that does not support the multi owner batch auditing (e.g., Zhu's IPDP)is compared. Fig. 4c also demonstrates that the batch auditing for multiple owners can greatly reduce the computation cost. Although in proposed simulation the number of data owners goes to 500, it can illustrate the trend of computation cost of the auditor that proposed scheme is much more efficient than Zhu's scheme in large-scale cloud storage systems that may have millions to billions of data owners.

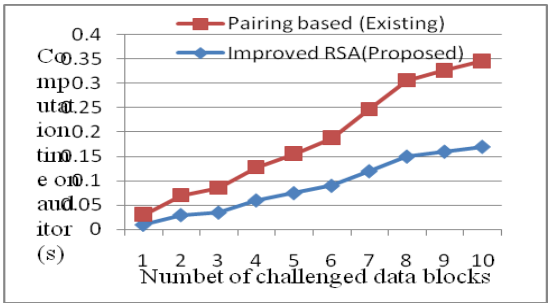

Figure 4a: Single owner single cloud

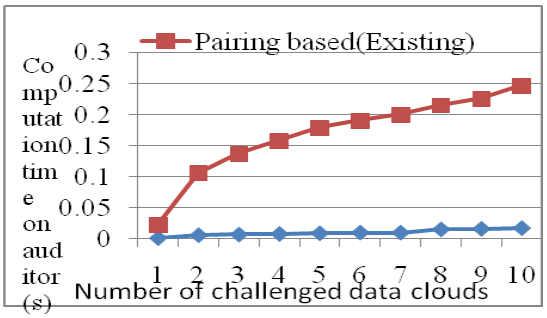

Figure 4b: single owner 5 blocks/ cloud

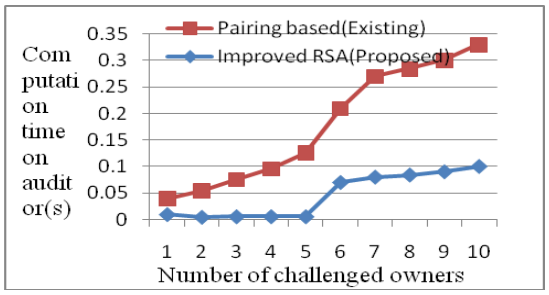

Figure 4c: single owner, 5 blocks/ cloud

Fig 4.Comparison of computation cost of the auditor (s =50).

2. COMPUTATION COST OF THE SERVER

The computation cost of the server versus the number of data blocks in Fig. 5a and the number of data owners in Fig. 5b are compared.The Proposed scheme moves the computing loads of the auditing from the auditor to the server, such that it can greatly reduce the computation cost of the auditor.

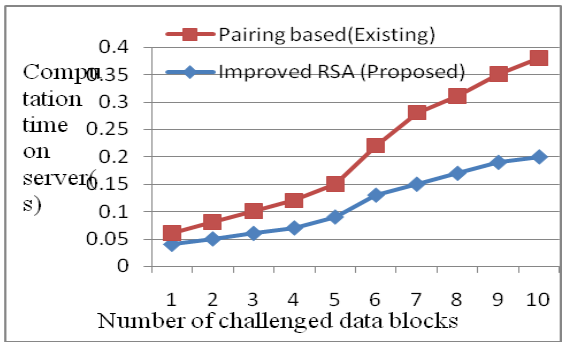

Figure 5(a): Single owner single cloud

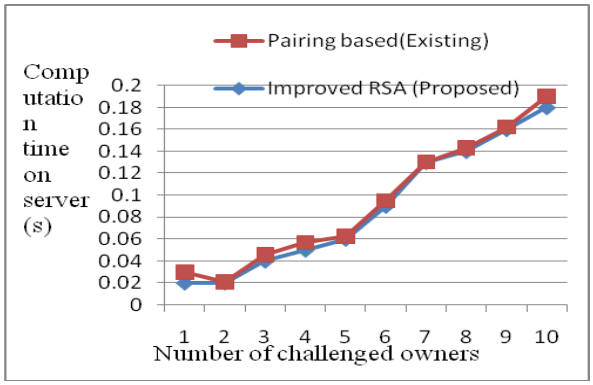

Figure 5(b): Single cloud, 5 blocks /owner

Fig 5. Comparison of computation cost on the server (s =50)

IV. CONCLUSION

In this paper, the improved RSA cryptography with bilinear property of computational bilinear Diffie Hellman to ensure the data privacy is proposed. It protects the data privacy against the auditor. Thus, multi cloud batch auditing protocol doesn't need any additional organizer. Batch auditing protocol can even support the batch auditing for multiple owners. Moreover, proposed auditing theme incurs less communication cost and less computation cost of the auditor by moving the computing loads of auditing from the auditor to the server that greatly improves the auditing performance and might be applied to large-scale cloud storage systems

References

[1] Amazon elastic compute cloud (Amazon EC2), http://aws.amazon.com/ec2/.

[2] Shingare Vidya Marshal "Secure Audit Service by Using TPA for Data Integrity in Cloud System" International Jproposednal of Innovative Technology and Exploring Engineering (IJITEE) ISSN: 2278-3075, Volume-3, Issue-4, September

[3] Cong Wang, Student Member, IEEE, Qian Wang, "Toward Secure and Dependable Storage Services in Cloud Computing" IEEE transactions on services computing, vol. 5, no. 2, April-June 2012.

[4] Rakhi Bhardwaj, Vikas Maral " Dynamic Data Storage Auditing Services in Cloud Computing" International Jproposednal of Engineering and Advanced Technology (IJEAT) ISSN: 2249 – 8958, Volume-2, Issue-4, April 2013

[5] Qian Wang, Student Member, IEEE, Cong Wa "Enabling Public Auditability and Data Dynamics for Storage Security in Cloud Computing" IEEE transactions on parallel and distributed systems, vol. 22, no. 5, may 2011

[6] Cong Wang, Member, IEEE, Sherman S.M. Chow "Privacy-Preserving Public Auditing for Secure Cloud Storage" IEEE transactions on computers, vol. 62, no. 2, February 2013.

[7] Syam Kumar P, Subramanian R "An Efficient and Secure Protocol for Ensuring Data Storage Security in Cloud Computing" IJCSI International Jproposednal of Computer Science Issues, Vol. 8, Issue 6, No 1, November 2011 ISSN (Online): 1694-0814 www.IJCSI.org

[8] Md.Tajuddin*, K.China Busi "An Enhanced Dynamic Auditing Protocol in Cloud Computing" International Jproposednal of Engineering Trends and Technology (IJETT) - Volume4 Issue7- July 2013

[9] Jean-Luc Beuchat, Je´ re´mie Detrey "Fast Architectures for the Pairing over Small-Characteristic Super singular Elliptic Curves" IEEE transactions on computers, vol. 60, no. 2, February 2011

[10] Kirsten Eisentr Ager, Kristin Lauter "Improved weil and tate pairings for elliptic and hyper elliptic curves"

[11] Yan Zhu, Member, IEEE, Gail-Joon Ahn "Dynamic Audit Services for Outsproposedced Storages in Clouds" IEEE transactions on services computing, vol. 6, no. 2, april-june 2013

[12] Kan Yang, Student Member, IEEE, and Xiaohua Jia, Fellow, IEEE "An Efficient and Secure Dynamic Auditing Protocol for Data Storage in Cloud Computing" IEEE transactions on parallel and distributed systems, vol. 24, no. 9, September 2013

[13] Zhu, Y., Hu, H., Ahn, G., Yu, M.: Cooperative provable data possession for integrity verification in multi-cloud storage. IEEE Trans. Parallel Distrib. Syst. 23(12) 2231–2244 (2012) .

[14] Y. Zhu, H. Hu, G. Ahn, and M. Yu, "Cooperative Provable Data Possession for Integrity Verification in Multi-Cloud Storage," IEEE Trans. Parallel and Distributed Systems, vol. 23, no. 12, pp. 2231-2244, Dec. 2012

