Global Journal of Researches in Engineering: J
General Engineering
Volume 14 Issue 2 Version 1.0 Year 2014
Type: Double Blind Peer Reviewed International Research Journal
Publisher: Global Journals Inc. (USA)
Online ISSN: 2249-4596 & Print ISSN: 0975-5861

Indirect Mutual Trust and Allowing Dynamic Data for Cloud Storage System

By Amol Bombe & Shekhar Jagtap
Pune University, India

Abstract- Cloud Computing shifts the databases and application software to the centralized huge data centers, where the organization of the data and services can not be completely truthful. Different organizations generate a huge quantity of responsive data including private information, electronic health report, and economic information. A data owner paid for a desired level of protection and has to get some returns in case of any misbehavior dedicated by the cloud service providers (CSP). This work studies the difficulty of ensuring the reliability of data storage in Cloud Computing. In exacting, we consider the task of allowing a trusted third party (TTP), to confirm the reliability of the dynamic data stored in the cloud. Nearly all universal types of data operation, such as block insertion, deletion and modification, is also a important step toward reasonableness, while services in Cloud Computing are not limited to backup data or archive only. We studied cloud-based storage method so as to let the data owner to advantage from the services offered by the CSP and allows indirect mutual trust between data owner and CSP. It make sure that authoritative users (i.e., persons who have the right to access the owner's data or files) obtain the most recent version of the outsourced data it permits the data owner to grant access or revoke access to the outsourced data.

Keywords: access control, cloud computing, data security, data outsourcing, cloud service provider, mutual trust.

GJRE-J Classification : FOR Code: 089999

Strictly as per the compliance and regulations of :

Indirect Mutual Trust and Allowing Dynamic Data for Cloud Storage System

Amol Bombe [α] & Shekhar Jagtap [σ]

Abstract- Cloud Computing shifts the databases and application software to the centralized huge data centers, where the organization of the data and services can not be completely truthful. Different organizations generate a huge quantity of responsive data including private information, electronic health report, and economic information. A data owner paid for a desired level of protection and has to get some returns in case of any misbehavior dedicated by the cloud service providers (CSP). This work studies the difficulty of ensuring the reliability of data storage in Cloud Computing. In exacting, we consider the task of allowing a trusted third party (TTP), to confirm the reliability of the dynamic data stored in the cloud. Nearly all universal types of data operation, such as block insertion, deletion and modification, is also a important step toward reasonableness, while services in Cloud Computing are not limited to backup data or archive only. We studied cloud-based storage method so as to let the data owner to advantage from the services offered by the CSP and allows indirect mutual trust between data owner and CSP. It make sure that authoritative users (i.e., persons who have the right to access the owner's data or files) obtain the most recent version of the outsourced data it permits the data owner to grant access or revoke access to the outsourced data.

Index Terms: access control, cloud computing, data security, data outsourcing, cloud service provider, mutual trust.

I. Introduction

Now a day in the existing time of digital world, different organizations generate a huge quantity of responsive data including private information, electronic health report, and economic information.

The local organization of such large quantity of records is challenging and expensive due to the necessities of large storage space capacity and trained personnel. For that reason, Storage-as-a-Service presented by cloud service providers (CSPs) emerged as a resolution to ease the load of huge local records storage space and decrease the preservation price through means of outsourcing data storage space. Since the owner of data physically releases responsive data to a remote Cloud Service Provider, there are a number of concern about, access control, integrity, and confidentiality of the data [2].

The confidentiality feature be able to assured by the owner via encrypting the information previous to outsourcing toward distant servers. For verifying information honesty over cloud servers, researchers have projected provable data possession method to authenticate the intactness of data stored on remote sites. To well confirm the reliability of data A number of PDP protocols have been presented, evidence of retrievability was introduced as a stronger method than PDP in the sense that the complete data file be able to reconstructed from parts of the data that are consistently stored on the servers.

Normally, traditional access control techniques believe the existence of the storage servers and the data owner in the same trust domain. This assumption, on the other hand, no longer grip after the data is outsourced to a remote Cloud Service Provider, which obtain the full charge management of the outsourced data, and The data owner lives in outside of the trust domain. A possible resolution can be obtained to allow the owner to implement right to use control of the data stored on a remote untrusted cloud service providers. Through this resolution, the information is encrypted under a assured key, which is common only with the authoritative client. As they do not have the decryption key, the illegal client, including the cloud service providers, are not capable to use the data.

This common resolution has been broadly incorporated interested in existing schemes, which aspire at providing information storage protection on remote servers which is untrusted. One more class of resolutions makes use of characteristic-based encryption to complete fine-grained retrieve control [3]. Different approaches contain examined that give confidence to the owner of data to outsource the data, and propose some type of assurance interrelated to the access control, integrity, and confidentiality of the outsourced data. These move toward avoid and identify malicious procedures from the cloud service providers side. On the additional, the CSP desires to be defended from a untruthful owner, who efforts to achieve prohibited compensations by untruly arguing data corruption above cloud servers. This concern, if not perfectly handled, can reason the cloud service providers to depart out of business [5]. In this paper, we suggest a design that deals with important concerns associated to outsourcing the storage space of data, specifically *dynamic data, newness, mutual trust,* and *access control.* The remotely accumulated data can be not just accessed by authoritative users, but as well

Author α σ: Department of Computer Engineering University of Pune. Zeal education society, Dnyanganga collage of engineering and research, narhe-pune, Maharashtra.
e-mails: Amolbombe26@gmail.com, Shekharjagtap8593@gmail.com

updated and ranged by the owner. After modifying, authorized clients should obtain the newest version of the data, a method is essential to identify whether the received data is stale. Mutual trust between the data owner and the CSP is another imperative issue, which is attended to in the projected method. A method is introduced to establish the untruthful party, misbehavior.

Since any side is identified and the dependable party is recognized. Final but not slightest, the access control is measured, which permits the owner to revoke or grant rights of access to the outsourced data.

a) Main Contributions

Our contributions can be summarized in two main points.

i. The completion and plan of a cloud-based storage system that has the following features:

- It allows a data owner to outsource the data to a secluded CSP, and execute full dynamic operations at the block-level, i.e., it chains operations such as block insertion, modification, deletion, as well as append.
- It ensures the freshness property, i.e., the authoritative users receive the mainly fresh translation of the data.
- It establishes not direct common trust between the CSP and the data owner since each social gathering resides in a dissimilar trust domain.
- It enforces the access power for the outsourced data.

b) We talk about the security facial appearance of the future scheme. As well, we give good reason for its presentation through experimental and theoretical analysis evaluation of storage, communication, and computation overheads.

II. Related Work

Existing study work can be establish in the area of honesty verification of outsourced information, data storage security on untrusted remote servers and access control of outsourced information. The name cloud had previously come into profit-making use in the near the beginning 1990s to large Asynchronous Transfer Mode networks. In 21st century, he name ''cloud computing'' had appear, even though major focus at this instant was on Software as a Service (SaaS). They practical many technologies of user web sites like Google and Yahoo! to industry applications. They also provide the concept's like ''on demand'' and ''Software as a Service'' with their real industry and successful clients. Storage as a Service is a significant service of cloud computing referred as Infrastructure as a Service (IaaS). Amazon's Elastic Compute Cloud (EC2) and Amazon Simple Storage Service (S3) are well known examples of cloud data storage. On the other side next to with these benefits' cloud computing faces large challenge i.e. data storage security problem which is an important aspect of Quality of Service .Once client puts data on the cloud rather than locally, he has no power over it i.e. unauthorized users could modify client's data or destroy it and even cloud server plan attacks. Cloud client are mostly concerned about the security and reliability of their data in the Remote Server. Amazon's S3 [1] is such a good example.

a) Integrity Verification of Outsourced Data

For verify data integrity over cloud servers, researchers have planned provable data possession technique to validate the intactness of data stored on remote sites. A amount of PDP protocols have been offered to efficiently validate the honesty of data. Proof of retrievability was introduce as a stronger system than PDP in the logic that the entire data file can be reconstruct from portion of the data that are consistently irretrievability of files on archive service systems. Build irretrievability" (POR) model for ensure the remote data integrity. Their system combines spot-checking and error-correcting code to make sure both possession and stored on the servers. Describe a official "proof of on this model and construct a random linear function based homomorphism authenticator which enable unlimited number of query and requires less communication overhead.

b) Data Storage Security on Untrusted Remote Servers

Normally, usual right to use control techniques imagine the existence of the data owner and the storage servers within the same trust domain. This statement, though, no longer holds when the information is outsourced to a remote cloud server provider, which takes the full accuse of the outsourced data management, and reside exterior the trust area of the data owner. A practical solution can be presented to enable the owner to impose access control of the data store on a remote entrusted CSP. The information is encrypted under a convinced key, which is public only with the authorized users. The unauthorized clients, including the cloud service provider, are incapable to access the information seeing as they do not have the decryption key. This common answer has been widely built-in into existing schemes, which plan at provided that data storage security on entrusted remote servers. A few authorized users of the information have the concession to read and write, while others can only read the information. A data owner encrypts the blocks with same information keys which are encrypted by using a master public key. The data owner keep a master private key to decrypt the same data keys.

c) Access Control of Outsourced Data

The idea of over-encryption to impose access control has also been used by Wang et al. In their system, the owner encrypts the data block-by-block, and construct a binary tree of the block keys. The binary tree enable the owner to decrease the number of keys

given to each client, where different keys in the tree can be generate from one general parent node. The remote storage server performs over-encryption to avoid revoke clients from receiving access to simplified data blocks. Another class of answer utilizes attribute-based encryption to accomplish fine-grained access control. but these system do not implement mutual trust between the data owner and the remote servers. Different approach have been investigate that give confidence the owner to outsource the information, and offer some sort of guarantee related to the privacy, integrity, and access control of the outsourced data. On the another way, the CSP needs to be protected from a untruthful owner, who attempts to get unlawful compensations by falsely claiming data dishonesty over cloud servers. This fear, if not correctly handle, can cause the CSP to go out of industry. In this job, a system is planned that addresses important issue linked to outsourcing the storage of data, namely privacy, integrity and access control. Mutual trust in between the data owner and the CSP is another vital issue, address in the proposed system. A mechanism is introduced to determine the untruthful party, i.e. naughtiness from any side is detected and the answerable party is recognized.

The proposed cloud-based storage system has the following features:

i. It allow a information owner to outsource the data to a cloud service provider, and it ensure that only authorized client (i.e., Those who have the true to access the owner's file) receive the outsourced data i.e. It enforce the right of entry control of the outsourced data.

ii. It establishes indirect mutual trust in between the data owner and the cloud service providers since each party reside in a dissimilar trust field.

III. Our System and Assumptions

a) System components and relations

i. *Data owner*

That can be the group / separate generating complex or sensitive data to be stored in the cloud and made accessible for controlled outside use.

ii. *Cloud Service Provider (CSP)*

Who achieves cloud servers and provides paid storage interplanetary on its substructure to store the holder's or owner's files and make them accessible for approved users.

iii. *Authorized users*

A set of owner's clients who have the right to right of entry the inaccessible information.

iv. *Trusted third party (TTP)*

An entity who is important by all other method components, and has skills to detect/require untruthful parties. The cloud computing storage classic well-thought-out in this work contains of four main components as showed in Figure 1. The relationships between dissimilar method components are characterized by double-sided arrows, where hard and sunk arrows represent belief and disbelief relationships, correspondingly. For example, the data owner, the authorized users i.e. client and the CSP (cloud service provider) trust the TTP (Trusted Third Party). On the further hand, the data owner and the authorized users have shared distrust relationships with the CSP. Therefore, the TTP is used to permit incidental shared trust between these three components. There is a through belief relationship among the data owner and the authorized users.

b) Outsourcing and Accessing

For secrecy, the owner encrypts the information earlier sending to cloud servers. To admittance the data, the approved user sends a data-access invitation to the CSP, and receives the information file in an encrypted form that can be decrypted using a top-secret key created by the approved user. It is supposed that the communication between the owner and the authorized users to validate their individualities has previously been completed, and it is not well-thought-out in this work. The TTP is a self-governing entity, and therefore has no inducement to scheme with any party. Though, any thinkable leakage of data in the way of the TTP must be prohibited to save the outsourced data private. The TTP and the CSP are continuously online, while the owner is spasmodically online. The approved users are able to access the information file from the CSP smooth when the owner is offline [9].

c) Threat Model

The CSP is entrusted, and therefore the concealment and honesty of information in the cloud may be at danger. For financial inducements and keeping a status, the CSP may hide information loss, or regain storage by clearance information that have not been or is infrequently accessed. On the further hand, a data owner and authorized users may scheme and untruthfully accuse the CSP to become a certain amount of recompense. They may untruthfully claim that data honesty over cloud servers has been dishonored [9].

d) Security Requirements

i. *Confidentiality*

Outsourced information must be confined from the trusted third party, the cloud service provider, and clients that are not access.

ii. *Integrity*

Outsourced information is required to remain integral on cloud servers. The data owner and authorized users must be enable to identify data dishonesty over the cloud service provider area.

iii. *Access Control*

Only authorized client are permissible to access the outsourced information.

iv. *CSP's defense*

The cloud service provider must be protected against false accusation that may be claim by dishonest owner/users, and such a hateful behavior is required to be exposed.

IV. Proposed Framework

a) Existing System

A directly promote result to detect corrupted from any side is from end to end digital signatures. For each file owner attaches digital signature earlier than outsourcing. The CSP (cloud service provider) first checked digital signature of owner before storing data on cloud. In case of unsuccessful confirmation, the CSP discards to store data and asks the holder to resend the accurate signature. If the signature is applicable, equally the file and signature are stored on the cloud servers. The digital signature achieves non-repudiation from the holder side. When an authoritative user (or the holder, or the owner) needs to get back the data file, the CSP sends file, CSP's signature and owner's signature on (file || owner's signature). The authorized user first checks the CSP's signature. In case of unsuccessful verification, the user asks CSP to re-perform the communication process. If CSP's signature is applicable, the user then checks owner's signature. If authentication fails, this indicates the dishonesty of data more than the cloud servers. The CSP cannot reject such dishonesty for the owner's signature is before checked and stored by the CSP next to with file. Because CSP's signature is connected with the established data, a dishonest owner cannot wrongly accuse the CSP as regards data reliability. The over explanation increases the storage transparency on cloud as owner's signature is stored next to with the file on cloud servers. Furthermore, there is an improved calculation overhead; CSP has to checked signature of owner earlier than storing file on cloud, and the authorized user checks two signatures for each acknowledged file. If the CSP receives file from trusted person other than the owner, the signature authentication is not needed since the trusted entity has no motivation for negation or agreement. Therefore, delegating minute part of owner's work to the TTP reduces both the computation and storage overheads. But the outsourced information must be kept private and any escape of data toward the TTP must be not permitted.

V. System Preliminaries

a) Lazy Revocation

The future system in this work allows the data owner to cancel the right of some users for accessing the outsourced data. In lazy revocation, it is suitable for users to read (decrypt) unchanged data blocks. However, modernized or new blocks must not be accessed by such cancelled users.

The idea is that allowing cancelled users to read unchanged information blocks is not a important loss in security. This is corresponding to accessing the blocks from cashed copies. Restructured or new blocks following a revocation are encrypted underneath latest keys. Lazy revocation trades re-encryption and data access charge for a degree of protection. However, it causes destruction of encryption keys, which is data blocks could have more than one key [5].

b) Key Rotation

Key rotation is a method in which a sequence of keys can be generated from an primary key and a master top secret key [7].

The progression of keys has two main properties:

- Only the owner of the master top secret key is able to generate the next key in the progression from the recent key, and
- Any authoritative user significant a key in the sequence is able to generate all before versions of that key. In other words, known the i-th key Ki in the sequence, it is computationally infeasible to compute keys Kl for $l > i$ exclusive of having the master top secret key, but it is straightforward to compute keys Kj for $j < i$.

The first property enables the data owner to cancel access to the data by producing latest keys in the progression, which are used to encrypt modernized/new blocks following a revocation (lazy revocation).

It is proposed to avoid a user cancelled during the i-th time from receiving access to data blocks encrypted during the l-th time for $l > i$. The second property allows authoritative users to maintain access to blocks that are encrypted underneath older versions of the recent key.

It enables the data owner to shift only a single key Ki to respected users for accessing all data blocks that are encrypted under keys Kig (rather than transferring a potentially large set of keys fK1;K2; : : : ; Kig). Therefore, the second property reduces the communication overhead on the holder side. The proposed scheme in this work utilizes the key rotation method]. Let N = pq denote the RSA modulus (p&q are prime numbers), a public key = (N; e), and a master top secret key d. The key d is acknowledged only to the data owner, and ed = 1 mod (p- 1) (q - 1).

Whenever a user's access is cancelled, the data owner generates a latest key in the progression (rotating forward). Let ctr point to the index/version number of the recent key in the keys progression.

The owner generates the next key by exponentiation Kctr with the master top secret key d: Kctr+1 = Kdctr mod N. Authoritative users can recursively generate older versions of the current key by exponentiations with the public or unrestricted key component e:Kctr-1 = Kectr mod N (rotating

backward). The RSA encryption is used as a pseudorandom digit generator; it is not likely that frequent encryption consequences in cycling, for if not, it can be used to thing the RSA modulus N [7].

c) Broadcast Encryption

Broadcast encryption (bENC) allows a presenter to encrypt a message for an chance subset of a collection of users. The users in the subset are only acceptable to decrypt the message. However, even if all users outside the subset scheme they cannot access the encrypted message. Such systems have the agreement struggling property, and are used in lots of practical applications as well as TV contribution services and DVD content protection. The proposed method in this work uses bENC to implement access control in outsourced data [9].

The bENC is together of three algorithms: SETUP, ENCRYPT, and DECRYPT.

i. *Setup*

This algorithm takes as contribution the number of system users n. It defines a bilinear group G of major order p with a generator g, a repeated multiplicative group GT, and a bilinear map ê : G×G → GT, which has the properties of bilinearity, computability, and no degeneracy.

The algorithm picks a unsystematic α ϵ Zp, computes gi = g(αi) ϵ G for i = 1,2,...,n,n+2,...,2n, and sets v = gβ ϵ G for β ϵ R Zp. The outputs are a public key PK = (g, g1,...,gn, gn+2,...,g2n,v) ϵ G2n+1, also n private keys {di}1≤i≤n, where di = gi β ϵ G.

ii. *Encrypt*

This algorithm takes as contribution a subset S ϵ {1,2,...,n}, and a public\key PK. It outputs a couple (Hdr, K), where K is a message encryption key And Hdr is called the header (broadcast cipher text). Hdr = (C0, C1) ϵ G2, wherever for t ϵ R Zp, C0 = gtas well as C1 = (v •πj ϵ S gn+1−j)t.

The key K = ê(gn+1, g)t is used to encrypt a message M (symmetric encryption) to be transmit to the subset S.

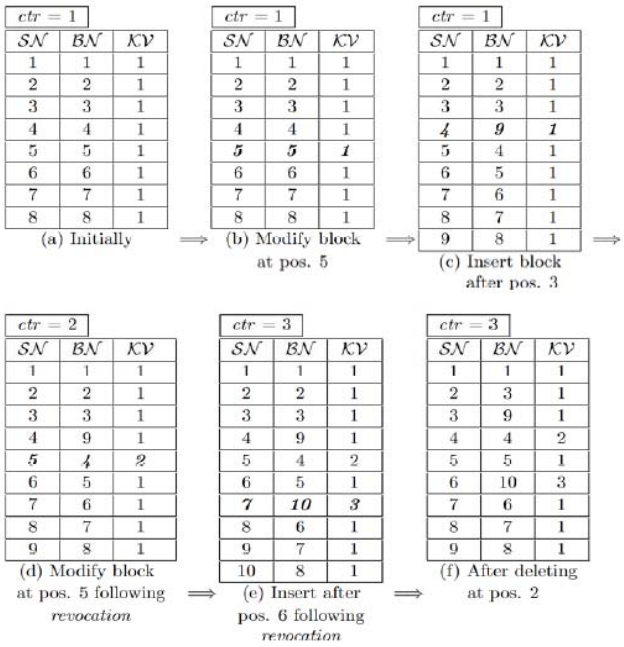
iii. *Decrypt*

This algorithm takes as contribution a subset S ϵ {1,2,...,n}, a user-ID i ϵ {1,2,...,n}, the private key di for user i, the header Hdr = (C0,C1), and the public key PK. If i ϵ S, the algorithm outputs the key K = ê(gi,C1)/ê(di • π j ϵ S ,j≠I gn+1−j+i,C0), which can be used to decrypt the encrypted description of M.

In the above structure of the bENC, a private key contains only one factor of G, and the broadcast cipher text (Hdr) consists of two factors of G. On the further hand, the public key PK is comprised of 2n + 1 factors of G. A second structure, which is a simplification of the first one, was accessible in to trade the PK size for the Hdr size. The main idea is to run several parallel instances of the first structure, where each instance can broadcast to at most B users. Setting B= [√n] results in a system with O(√n) factors of G for each of PK and Hdr. The private key is at a standstill just one factor.

d) Block Status Table

The Block Status Table (BST) is a minute dynamic data structure used to restructure and access file blocks outsourced to the CSP. The BST consists of three columns: Serial Number (SN), Block Number (BN), as well as Key Version (KV). SN is an indexing to the file blocks. It indicates the physical location of each block in the information file. BN is a counter used to build a logical numbering/indexing to the file blocks. Therefore, the relation between BN and SN can be viewed as a mapping between the logical number BN and the physical location SN. KV indicates the report of the key that is used to encrypt every block in the data file [9].



The BST is implemented as a connected list to make things easier the insertion and deletion of table entries. During completion, SN is not required to be store in the table; SN is measured to be the entry/table index. Thus, each table entry contains just two integers BN and KV (8 bytes), i.e., the total table size is 8m bytes, where is the number of file blocks. When a information file is initially created, the owner initializes both ctr and KV of each block to 1. If block alteration or addition operations are to be perform following
a revocation, ctr is incremented by 1 and KV of that customized/new block is set to be equal to ctr.

Fig. 2: change in BST Due to Different active Operation on a File F={bj}1≤j≤8 When a data file is initially created, the data owner initializes both ctr and KV of each block to 1. If block alteration or placing operations are to be performed following a revocation, ctr is incremented by 1 and KV of that customized/new

block is set to be equal to ctr .Figure shows some examples representing the changes in the BST due to dynamic operations on a data file F = {bj}1≤j≤8. When the file blocks are initially formed (Fig.(a)), ctr is initialized to 1, SNj =BNj = j, and KVj = 1: 1≤ j ≤ 8. Fig. (b) Shows no modify for update the block at location 5 since no revocation is performed. To add a new block after location 3 in the file F, Fig. (c) shows that a new entry h4,9,1i is added in the BST after SN3, where 4 is the physical location of the newly added block, 9 is the new logical block number compute by incrementing the maximum of all previous logical block numbers, and 1 is the version of the key used for encryption.

A first revocation in the scheme increments ctr by 1 (ctr = 2). Modifying the block at position 5 following a revocation (Fig.(d)) answers in setting KV5 = ctr. Thus, the table entries at location 5 become h5, 4, 2i. (Fig. (e)) shows that a new block is to be added after position 6 following a second revocation, which Increments ctr to be 3. In Fig. (e), a new table entry h7, 10, 3i is insert after SN6, where KV7 is set to be equal to ctr (the Most recent key version). Deleting a block at position 2 from the Data file requires deleting the table entry at SN2 and shifting all Ensuing entries one position up. Note that during all Dynamic operations, SN indicates the real physical positions of the information blocks in F.

VI. Experimental Evaluation

In this sector we experimentally calculate the computation overhead the planned scheme passes to a cloud storage system that has been commerce with static data with only confidentiality requisite. The experiments are showed using .NET on a method with an Intel(R) Xeon (R) 2-GHz processor and 3GB RAM running Windows XP. We are use algorithms hashing, broadcast encryption and digital signatures are executed using MIRACL library version 5.5.4. For a 128-bit safekeeping level, bENC uses an elliptic curvature with a 256-bit set order. In the experiments, we apply SHA-256, 256-bit BLS mark, and Barreto-Naehrig (BN) curvature defined over major field GF(p) with p = 256 bits and inserting degree = 12 (the BN curve with these limits is provided by the MIRACL library).

To assess the computation overhead on the owner or holder side due to dynamic actions, we execute 100 different block processes from which 50% are executed following cancelations (this percent is higher than an regular value in real applications). Scalability (i.e., how the method performs when more operators are added) is an main feature of cloud storage systems. The access regulator of the proposed scheme be contingent on the square root of the complete number of method users.

In the poorest case, the TTP executes only 4 hashes per dynamic demand to reflect the modification on the outsourced data. Thus, the maximum computation overhead on the TTP side is near 0.08 milliseconds, i.e., the proposed system brings light overhead on the TTP during the ordinary method actions. The computation overhead on the user side due to data access comes from five features separated into two groups.

The first group includes signatures confirmation and hash actions to confirm the acknowledged data (file and table). The second group includes broadcast decryption, backward key replacements, and hash actions to calculate the DEK. The first set costs about 10.77 seconds, which can be simply unknown in the getting time of the data (1GB file and 2MB table). To consider the computation time of the second set, we access the file later running 100 dissimilar block actions (50% of them are done subsequent revocations). Furthermore, we implement the regressive key rotations in the adjusted way. The second set costs around 1.03 seconds, which can be measured as the user's computation overhead due to information access.

As a reply to the information access appeal, the CSP computes two signatures: F and T. Thus, the computation overhead on the CSP lateral due to information access is about 10.75 seconds and can be simply unseen in the broadcast time of the data (1GB file and 2MB table).

To classify the corrupt party in the method in case of disagreements, the TTP authenticates two initials (F and T), computes joint hashes for the information (file and table), and relate the calculates hashes with the reliable values (THTTP and FHTTP). Therefore, the computation overhead on the TTP adjacent is about 10.77 seconds. Finished our experiments, we use individual one desktop computer to fake the TTP and achieve its work. In practice, the TTP may select to divide the work amongst rare devices or use a only device with a multi-core processor which is attractive dominant these days, and therefore the computation time on the TTP lateral is meaningfully reduced in several applications.

VII. Conclusion

Cloud provides a higher security and privacy to our data by maintaining encryption and decryption standards. Our data is provided with better security and data integrity due to cloud and the main aim of our system helps to support features like privacy, integrity, access control of the information. The cloud is planned that allow owner to advantage from facilities offered by the cloud service provider and enable indirect mutual trust in between them. To decide dispute that may occur concerning data honesty, a trusted third party is invoke to determine the untruthful party (owner/users or Cloud Service Provider).

We have some of the safety features into our system which are prior such as data privacy,

recognisation of data integrity, use of Trusted Third Party and finding untruthful owner. Data privacy is based on the safety of underlying encryption algorithm. Recognition of data integrity abuse base on the primate and second-primate confrontation properties of the utilize cryptographic hash function enforcement of right to use control based on Trust third party gives encrypted key to only authorized client and only authorized client can decrypt this key and get the key to study the outsourced data and finding of untruthful owner/user through a TTP.

So all above mentioned feature enhances our system and provide it with features such as privacy, integrity and data control which we have implemented into our system.

References Références Referencias

1. Amazon.com, "Amazon Web Services (AWS)," Online at http://aws.amazon.com
2. A. F. Barsoum and M. A. Hasan, "On verifying dynamic Multiple data copies over cloud servers," Cryptology ePrint Archive, Report 2011/447, 2011, 2011, http://eprint.iacr.org/.
3. S. Yu, C. Wang, K. Ren, and W. Lou, "Achieving secure, scalable, and fine-grained data access control in cloud computing," in Proceedings of the 29th conference on Information communications, ser. INFOCOM'10. IEEE Press, 2010, pp, 543-542.
4. Verifiability and data dynamics for storage security in cloud Computing," in Proceedings of the 14th European Conference On Research in Computer Security, 2009, pp. 355–370.
5. R. A. Popa, J. R. Lorch, D. Molnar, H. J. Wang, and L. Zhuang, "Enabling security in cloud storage SLAs with cloudproof," in Proceedings of the 2011 USENIX conference, 2011.
6. Z. Hao and N. Yu, "A multiple-replica remote data possession checking protocol with public verifiability," in Second International Symposium on Data, Privacy, and E-Commerce, 2010.
7. C. Wang, Q. Wang, K. Ren, and W. Lou, "Privacy-preserving public auditing for data storage security in cloud computing," in INFOCOM, 2010, pp. 525–533.
8. S. Narayan, M. Gagn´e, and R. Safavi-Naini, "Privacy preserving EHR system using attribute-based infrastructure," in Proceedings of the 2010 ACM workshop on Cloud computing security workshop, ser. CCSW '10. ACM, 2010, pp. 47–52.
9. Ayad Barsoum, Anwar Hasan, "Enabling Dynamic Data and Indirect Mutual Trust for Cloud Computing Storage Systems," in IEEE transactions on parallel and distributed systems.

This page is intentionally left blank