Enhanced Data Storage Solution for Security and Privacy Protection in Cloud Environment

1Ananta Malpani, 2Anchal Didwania, 3Chahat Puri, 4Kushagra Keshaw, 5Mrs. Radhika T.V
1student, 2student, 3student, 4student, 5Assistant Professor
Dayananda Sagar College of Engineering, Bangalore, India

Abstract - **With the high-speed evolution of cloud computing, cloud storage has been adopted by an increasing number of corporations and individuals, in there working as a suitable and on-demand outsourced application. Although, in a situation of lost data control, it becomes a top-priority requirement for the users to confirm whether cloud service providers have stored their data securely.Consequently, many researchers have put so much of effort for designing the auditing protocols directed at outsourced data. In this particular paper, with global and sampling block less verification as well as batch auditing we present an efficient public auditing protocol , where data dynamics to a great extent are supported with the state of the art. The dynamic structure in our protocol comprises of a doubly linked-list information table and a location array.In addition to that, structure, computational and communication overheads will also be reduced. The proposed protocol can also achieve all the desired properties with security analysis and numerical analysis.**

keywords - **Cloud computing, Security, cloud service providers, Cryptography, Batch Auditing**

1.INTRODUCTION

Cloud storage, which is one of the main cloud service, serves as a important tool by making the data outsourcing to the cloud a new trend. The high-speed growth of such type of cloud service has various causes such as its on-demand outsourcing function, ubiquitous network access, and location independent resources. For example, data are no any more local with the cloud storage, making sure that during the software or hardware failures data owners (DOs) do not have to worry. In addition to that, most importantly the maintenance, financial cost, time, and other resources would be reduced, decreasing the load on Data Owners and local devices. Although, data which are outsourced to the cloud are not kept securely and still go through from a different types of security attacks both internal and external. On the one side, malevolent network attacks, which are external and familiar to Internet users, threaten cloud data. Hackers might gain and steal cloud users data. They also might even corrupt and remove the data, destroying its privacy, integrity, and availability. Moreover, the outsourced data might suffer from cloud service providers' (CSPs') illegal behaviors. In particular, to save space for other clients' data a Cloud Service Provider could secretly remove some data in its storage cycle without authorization. On top of this, a CSP might attempt to obtain the data outsourced to the cloud. Thus, whether the attacks are internal or external, the confidentiality and integrity of cloud data are in danger. Hence, the design of an efficient auditing protocol directed at cloud data has become a research hotspot concerning cloud storage in cloud computing. With auditing, a DO who has already deleted the local copy of data and lost direct control of the data could remotely verify whether their data are stored correctly in the cloud.

1.1 Problem Statement

In existing works by other researchers, the file information simply consists of the file ID, making the length of the file ID long. With the increasing number of files increases over time, it becomes more difficult to make each file ID unique. In the real world, both the file ID bit and the user ID bit will be larger, and the number of identifiers will consequently be increased.

1.2 Objectives

To enable secure and efficient dynamic public auditing for cloud data, our protocol design should achieve the following objectives:

1) Public auditing**:** to allow anyone trusted (not just the DO itself) to verify the cloud data on demand without retrieving a copy of the data.

2) Storage completeness**:** to make sure that the CSP can never permit the auditor's verification when it does not store the owners' data completely.

3) Dynamic operations support**:** to allow the DOs to perform updates (insertion, deletion and modification) on their outsourced data files while promissing the efficient public auditing.

4) Storage freshness: to allow the cloud and the auditor to keep the updated version of data blocks and corresponding authenticators.

2. Existing System

Data deployed to the cloud are not kept safely and still go through from a variety of threats both internal and external. On the one side, malevolent network attacks, which are external and familiar to Internet users, threatens the cloud data. Hackers might access and steal cloud users' data or even corrupt and remove those data, destroying its privacy, integrity, and availability. On

the other side, the outsourced data might suffer from cloud service providers' (CSPs') illicit behaviors. A CSP could delete some data on its own in its storage cycle without authorization to save space for other clients' data. On top of this, a CSP might also try to attempt to obtain the data outsourced to the cloud environment. Consequently, whether the attacks are external or internal, the confidentiality and integrity of data stored on cloud are in danger. With auditing, a DO who has already deleted the local copy of data and lost direct control over the data could remotely check whether their data are stored correctly in the cloud. To make the process of verification considerably more convenient and energy efficient, a new entity named a third-party auditor (TPA) was introduced, which accepts the auditing delegation from DOs and then executes it. To support data dynamics, Dynamic-hash-table (DHT) is designed, which is a single linked sequence table. Though this idea is an efficient auditing scheme based on a two-dimensional data structure, but it still has some demerits. Firstly, the CSP could suffer from collusion attacks from the DO and the TPA, because time stamps for verification are generated by the DO and the auditor only serves the DO. Secondly, no index switcher is designed in, which can not indicate the relationship between the index number and the sequence number of a certain data block. Thirdly, the computational cost of the protocol is still comparatively high.

Disadvantages: **The computational cost of the protocol is still comparatively high. Protocols are too expensive in terms of communication and also in terms of computation. The other protocols while supporting data dynamics might suffer from low efficiency.**

3. LITERATURE SURVEY

1.**Role-Based Cryptosystem:** A New Cryptographic RBAC System, Based on Role-Key Hierarchy by Yan Zhu & Gail-Joon Ahn have proposed a role-key hierarchy structure along with hierarchical RBAC model. This was introduced to accommodate the requirements of cryptographic access control for large-scale systems. Based on this hierarchy model, we further proposed many practical role-based security mechanisms to provide support to support signature, authentication and encryption constructions on elliptic curve cryptosystem. Our experiments clearly stated the proposed schemes are flexible and efficient enough to support large-scale systems.

Advantages:

- Supports an infinite number of users.
- Supports a bigger size role hierarchy with arbitrary complex structures.
- Comes up with a good tracing ability owing to the uniqueness of the user's private key.
- Has better performance and efficiency on security mechanisms.

Disadvantages:

- Cost is high.

2.**Provably Secure Role-Based Encryption with Revocation** Mechanism by Yan Zhu & Gail-Joon Ahn. In this paper, authors introduced a generic role-based encryption over RBAC model to support a flexible encryption of resources in RBAC systems. The proposed scheme supports fully collusion security under a special case of the GDDHE problem and implements the revocation at minimal cost and constant-size cipher- texts and decryption keys. Their scheme has better performance and scalability than existing solutions in encrypted file systems.

Advantages:

- Flexible.
- Provides security.

Disadvantages:

- Performance is low.

3.**Fuzzy Identity-Based Encryption:** The suggested method of Fuzzy Identity-Based Encryption was given by Amit Sahai and Brent Waters. Here authors introduced the concept of Fuzzy Identity Based Encryption, which allows for error-tolerance between the identity of a private key and thus the general public key used to encrypt a cipher text. Authors described two practical applications of Fuzzy-IBE of encryption using biometrics and attribute-based encryption. They presented our construction of a Fuzzy IBE scheme that uses set overlap because the space metric between identities. Finally, they proved our scheme under the Selective-ID model by reducing it to an assumption which can be viewed as a modified version of the Bilinear Decisional Diffie-Hellman assumption.

Advantages:

- A Fuzzy IBE scheme are often applied to enable encryption using biometric inputs as identities.
- Secure against collusion attacks
- Error-tolerant.

4.**Ciphertext-Policy Attribute-Based Encryption:** Ciphertext-Policy Attribute-Based Encryption by John Bethencourt & Amit Sahai. Authors created a system for Ciphertext-Policy Attribute Based Encryption. Their system allows for a replacement quite encrypted access control where user's private keys are specified by a gaggle of attributes and a celebration encrypting data can specify a policy over these attributes specifying which users are able to decrypt. This method allows policies to be expressed as any monotonic tree access structure and is resistant to collusion attacks during which an attacker might obtain multiple private keys. Finally, they provided an implementation of our system, including several optimization techniques.

Advantages:

- Encrypted data are often kept confidential albeit the storage server is untrusted.
- Secure against collusion attacks.

Disadvantages:

- It is proved secure under the generic group heuristic rather under a more standard and non-interactive assumption.

5. **Enforcing Role-Based Access Control for Secure Data Storage within the Cloud:**by Lan Zhou, Vijay Varadharajan and Michael Hitchens. during this paper, authors have considered security requirements for storage of data within the cloud and proposed a hybrid RBE scheme that mixes role-based access control with encryption to deal with them. Authors have constructed a selected RBE scheme using the BE scheme. Authors have conducted security analysis of our scheme and have given proofs to point out that our scheme is secure against adaptive attack and revocable- ID attack. Authors have discussed the performance and efficiency of our scheme and have compared it with other previously related work. Authors have shown that our scheme has several superior characteristics like constant size ciphertext and decryption key, efficient user revocation and user management, and therefore the ability to handle role hierarchies. Authors have also considered some aspects which will be optimized to realize efficient implementation.

Advantages:

- Suitable for giant scale systems..
- Provide good performance.
- Highly efficient.

Disadvantages:

- Not scalable.

In Role-Based Cryptosystem: a replacement Cryptographic RBAC System supported Role-Key Hierarchy proposed a more practical cryptographic RBAC model, called role-key hierarchy model, to support a selection of security measures including signature, identification and encryption supported role-key hierarchy. the most objective is to map the role hierarchy in RBAC into a key management system. With the help of rich algebraic structure of elliptic curve, introduced a role-based cryptosystem construction to verify the rationality and validity of the proposed model. This construction provides more efficient and versatile control than other hierarchical key assignments. More importantly, some unique security mechanisms, like role based signature, authentication, and encryption, are supported by the event . additionally, several advanced features, like role or user revocation, tracing, and anonymous, could be implemented also . As a crucial a part of role-based cryptosystem, Role-Based Encryption (RBE) enables an access control mechanism over encrypted data by hiding access permissions and as- signed roles into private keys and ciphertexts. one among the benefits of RBE is that the ability to simply meet up with the prevailing RBAC models and systems.

It focuses on the development of a cryptosystem compatible with hierarchical RBAC model. With the help of bilinear pairings, we present an enhanced Role-Based Encryption with revocation mechanism. The new scheme provides more flexible control than other schemes, also as an efficient revocation mechanism to support any number of users (or identities) and roles. Furthermore, our scheme has following new properties: key hierarchy can support arbitrary partial-order structures and a vast number of roles; a manager can dynamically add infinitely many users without revising the prevailing ciphertexts and user's private keys; and encryption is collusion-secure for arbitrarily large collusion of users. Moreover, our construction also achieves an optimal bound of over- head rate for both ciphertexts and decryption keys. The proposed scheme supports fully collusion security under a special case of the GDDHE problem and implements the revocation at minimal cost and constant-size cipher-texts and decryption keys. Our scheme has better performance and scalability than existing solutions in encrypted file systems.

The authors proposed a replacement sort of Identity-Based Encryption that we call Fuzzy Identity-Based Encryption during which we view identities as a group of descriptive attributes. during a Fuzzy Identity- Based Encryption scheme, a user with the key key for the identity ω is during a position to decrypt a ciphertext encrypted with the overall public key ω' if and as long as ω and ω' are within a specific distance of each other as judged by some metric. Therefore, our system allows for a specific amount of error-tolerance within the identities. Fuzzy-IBE gives rise to 2 interesting new applications. the primary is an Identity-Based Encryption system that uses biometric identities. that's we'll view a user's biometric, as an example an iris scan, as that user's identity described by several attributes then encrypt to the user using their biometric identity. Since biometric measurements are noisy, we cannot use existing IBE systems. However, the error-tolerance property of Fuzzy-IBE allows for a private key (derived from a measurement of a biometric) to decrypt a ciphertext encrypted with a rather different measurement of an equivalent biometric. Secondly, Fuzzy IBE are often used for an application that we call "attribute based encryption".

In this application a celebration will wish to encrypt a document to all or any or any users that have a specific set of attributes. suppose a head of some department wants to encrypt documents. In this case it'd encrypt to the identity {"hiring-committee", "faculty", "systems"}. Any user who has an identity that contains all of those attributes could decrypt the document. The advantage to using Fuzzy IBE is that the document are often stored on an easy untrusted storage server instead of relying on trusted server to perform authentication checks before delivering a document. So they offer a secure tree-based search scheme against the encrypted cloud data, which have attributes like multi keyword ranked search and dynamic operation on the document collection. Specifically, the vector space model and thus widespread "term frequency (TF) × inverse document

frequency (IDF)" model are combined within the index construction and query generation to supply multi keyword ranked search. so as to urge high search efficiency, we construct a tree-based index structure and propose a "Greedy Depth-first Search" algorithm supported this index tree. thanks to the special structure of our tree-based index, the proposed search scheme can flexibly achieve sub-linear search time and affect the deletion and insertion of documents. The secure kNN algorithm is employed to encrypt the index and query vectors, and meanwhile ensure accurate relevance score calculation between encrypted index and query vectors. To resist different attacks in several threat models, we construct two secure search schemes: the essential dynamic multi-keyword ranked search (BDMRS) scheme within the known ciphertext model, and therefore the enhanced dynamic multi-keyword ranked search (EDMRS) scheme within the known background model. within the proposed scheme, the data owner is responsible for generating updating information and sending them to the cloud server. Thus, the info owner must store the unencrypted index tree and thus the knowledge that are necessary to recalculate the IDF values. Such an active data owner won't be very suitable for the cloud computing model. it'd be a meaningful but difficult future work to style a dynamic searchable encryption scheme whose updating operation are often completed by cloud server only, meanwhile reserving the power to support multi-keyword ranked search. additionally, because the foremost of works about searchable encryption, our scheme mainly considers the challenge from the cloud server. Actually, there are many secure challenges during a multi-user scheme. Firstly, all the users usually keep the same secure key for trapdoor generation during a symmetric SE scheme. during this case, the revocation of the user may be a big challenge. If it's needed to revoke a user during this scheme, we'd wish to rebuild the index and distribute the new secure keys to all or any or any or any the authorized users. Secondly, symmetric SE schemes usually assume that every one the data users are trustworthy.

The authors proposed a hybrid scheme that integrate access control with cryptography and key distribution to address security needs for data storage in the cloud.Hybrid scheme is reffered as role-based encryption (RBE) as it involves the use of cryptography with role-based access control to untangle security management. The data is encrypted by the owner in such a way that only the users with significant roles as specified by a role-based access control policy can decrypt the data and view the data. Recall that the central notion of role-based access control is that permissions are associated with roles, and users are assigned to appropriate roles, which simplifies the management of permissions. Roles are created for the various group of users based on users responsibilities and ability.A role manager is used to assign a role to a user, or revoke a role from a user. The owner of data can grant permissions to the roles while adding new data, and can also revoke the permissions from the roles on some existing data as needed. Even the cloud provider (who stores the data) will not be able to see the content of the data if the provider is not given the appropriate role.The data is encrypted by the owner for the role that user wants to join hence, the user will be able to access that data from then on, and the owner does not need to re-encrypt the data. A user can be call off at any time [e.g if he/she is found to be mischievous], in which case the revoked user will have no longer access to any future data encrypted to this role. In gain,our RBE scheme is able to deal with role hierarchies, whereby roles can inherit authorization from other roles. A role can have sub-role,in the hierarchical structure. If a role P inherits all the permissions that role Q has, then we say role P is a predecessor role of role Q, and role Q is a successor role of role P. In this scheme, the ciphertext and the decryption key that the user needs to keep is constant in size, and the user can be revoked from the role without affecting the owners and other users of the identical role.

In this task, we supply the first construction of a ciphertext-policy attribute-based encryption (CP-ABE) to address this problem, and give the first construction of such a scheme. In our system, a user's private key will be linked with an arbitrary number of attributes expressed as strings. On the other side, when a message in our system is encrypted by the party, they specify an associated access structure over attributes. A user can only decode a ciphertext if that users attributes pass through the ciphertexts access structure. At a scientific level, access structures in our system are described by a monotonic 'access tree', where nodes of the access structure are composed of threshold gates and the attributes represented as leaves.In key-policy attribute based encoded,ciphertexts are associated with group of descriptive attributes, and user's keys are associated with policies (the reverse of our situation). We stress that in key- policy ABE, the encryptor exerts no control over who has access to the data he/she encrypts, except by his/her choice of descriptive attributes for the data. Rather, he/she must trust that the key-issuer issues the suitable keys to grant or deny access to the appropriate users. In other words, the "intelligence" is assumed to be with the key issuer, and not the encryptor. In our context, the encryptor must be able to intelligently decide who should or should not have access to the data that he/she encrypts. At a technological level, the main objective is to attain the collusion-resistance: If multiple users fascinate, they will be only able to decode a ciphertext if at least one of the users could decode it on their own. In the construction private keys will be identified with a set S of descriptive attributes. The users that wishes to encode a message will determine through an access tree structure a policy that private keys must satisfy in order to decode. Each interior node of the tree is a threshold gate and the attributes are associated by the leaves. For example, we can show a tree with "OR" and "AND" gates by using respectively 2 of 2 and 1 of 2 threshold gates..The authors develop a novel method of keyword transformation based on the unigram. For misspelling of one character, this procedure reduces the Euclidean distance between the correct keyword and the misspelled keyword. Moreover, this method is also effective for other spelling mistakes. Additionally,we introduce the stemming algorithm to obtain the root of the word. Using this technique, the keywords with the same root can also be queried. We take the keyword weight into consideration in constructing the ranked list of the results. The files will have greater chances to appear first on the list which are more relevant to the keyword. Here, we consider a cloud system consisting of data owner, data user and cloud server. In the system model, data owner has a set of n data files F0 = (f1, f2, f3, · · · , fn). It is then outsources in the encoded form C to the cloud server. In these encrypted files, data owner will build a secure searchable index I on the keyword set W extracted from F to enable efficient search operation. Both the the encrypted data files C and index I are outsourced to the cloud server. The encrypted data files for the given keywords are searched and the authorized user computes a corresponding trapdoor T and sends it to cloud server.when the

trapdoor T is received, the cloud server is responsible to search the index I and return the corresponding group of the encoded documents. To upgrade the file retrieval accuracy and save the communication cost, the search result should be graded by the cloud server and return the top-K relevant files to the user as the research outcome. In the threat model, both data owners and data users are trusted. Hence, the cloud server is honest but curious. The cloud server may try to obtain other sensitive information from user search requests while performing keyword-based search over C even when data files are encrypted. So the search should be performed in such a way that allows data files to be retrieved in a secure manner hence,revealing as little information as possible to the cloud.

E-Health clouds are gaining popularity by facilitating the storage and sharing of big data in healthcare centre. Authentication is essential to ensure the security and privacy of highly sensitive health data. Three-factor authentication combining password, mobile device and biometrics perfectly matches this necessity by providing high security strength. In addition, their scheme is also susceptible to offline password guessing attacks in the login and password update phase, in the case that the mobile device is lost or stolen. Furthermore, it fails to provide user calling off when the mobile device is lost or stolen.Then,we have proposed a robust three-factor authentication protocol to remedy these drawbacks in Wu et al.'s scheme. We have proved that these factors fulfills mutual authentication in the BAN logic. In addition, through the informal security analysis, we have displayed that the proposed protocol can withstand various known attacks and can provide more security features compared with Wu et al.'s protocol. The authors propose a public auditing way for the recreation of code based on storage on cloud in which the checking of integrity and regeneration (of failed data blocks and authenticators) are executed by a third-party auditor and a trustworthy proxy separately in place of the owner of the data.Rather than directly acclimatising the existing public auditing way to the multiple-server setting, a authenticator is designed by us, which is more useful and appropriate for regeneration of codes. The coffecients are encrypted to prevent privacy against the auditor which is much better than applying the data blind method and proof blind technique. Different threats and problem may arise in our new system model with a proxy and the analysis provided by our security shows that our system can handle these situations very well. An innovative homomorphic validator based on BLS type signature is designed which can be made and generated by a couple of secret keys and can be verified publicly.The authenticators can be efficiently computed by utilising the linear subspcae of the regenerating codes. The data owners with low end computation devices can adapt this in which only the native blocks needs to be signed in. This is the first scheme which allows preserving of privacy public auditing for regenerating code-based cloud storage. The leakage of original data can be avoided by masking the PRF(Pseudorandom Function) during the setup phase.The online burden for the regeneration of blocks and authenticators at faulty server on the data owners can be completely avoided and it also provides privilege to a proxy for reparation. Several measures are taken to rigidity and efficiency of our auditing scheme. The storage overhead and computational overhead of the data owner and the communication overhead during the audit phase can be reduces efficiently. This model involves four entities. Firstly the data owner, who owns large amounts of data files to be stored in the cloud. Secondly, the cloud which are managed by the cloud service provider. Third is the TPA(Third party auditor) who has expertise and capabilities to conduct public audits on the coded data in the cloud. Last the proxy agent who is semi-trusted and acts on behalf of the data owner and cloud servers to regenerate data blocks and autuenticator on the failed servers during the repair procedure. The proxy is suppose to be less powerful than cloud servers in terms of memory capacity and computation but more powerful than the data owners. The data owner adapted to TPA for integrity and repearation to the proxy to save resources as well as the online burden which is brought up by accidental repairing and periodic auditing.

4. PROPOSED METHODOLOGY

In this paper, we design an efficient public auditing protocol with novel dynamic structure for outsourced data within the cloud, which performs better than the state of the art. Note that global and sampling verification is presented to realize mutual trust between DO's and CSP's. The new dynamic structure efficiently provides data dynamics. Additionally, various auditing properties, like blockless verification and batch auditing, are supported. The summarized contributions of the paper is as follows.

4.1 Global and Sampling Verification Is Proposed within the Protocol: In practice, cloud and DOs may distrust one another before or during their cooperation. In our protocol, we support global and sampling verification to deal with this issue. Guarantee of sampling verification makes owners believe that the cloud has properly stored their data. The global verification gives the cloud confidence against owners who aren't always behaving appropriately and eliminate the fear of being wrongly accused.

4.2 Efficient Data Dynamics With a completely unique Dynamic Structure Are Provided within the Protocol: To the simplest of our knowledge, we are the primary to style a dynamic structure combining a doubly linked info table and a location array to efficiently support data dynamics. The structure successfully handles the connection between a given data block and its specific location, making data update and batch auditing significantly more convenient. Additionally, in practice, owners aren't always online for data updates to save lots of time and energy. Hence, although data freshness is reduced, lazy updates are supported to scale back overhead.

4.3 Various Auditing Properties Are Supported by the Protocol: Various important properties are established within the proposed auditing protocol to offer it greater practical value. Public auditing is supported with a trustworthy TPA equipped with professional knowledge introduced to alleviate the burden on owners. Blockless verification is realized to guard exact outsourced data from CSPs and auditors when auditing tasks are executed. Batch auditing is achieved to save lots of time and energy when several DOs would really like to verify multiple data files simultaneously.

4.4 Advantages: The proposed methodology will save longer and energy. the benefits of the DLIT are going to be reflected in batch operations at lower costs when checking out a particular element.

5. MODULES

5.1 Setup Phase: In this phase, some preparations are made for system setup, which are the responsibility of the DO. First, the DO generates the secret and public key pair in KeyGen. Then, some pre-processing tasks for the CSP and TPA are performed by the DO in Filepro2C and Filepro2T, respectively, where "2" means "to". The designs of the three algorithms are shown below.

5.1.1 KeyGen: The DO runs KeyGen to generate public and secret parameters. First, the DO chooses a random signing key pair (ssk, spk) for the signature of the file name. Then, the DO picks a $\in$ Z p randomly as one part of the system secret key and ssk as the other part. Consequently, the secret key would be sk = (a, ssk), which is known only by the DO. Moreover, the DO picks random generators g,u $\in$ G, and let v = ga. Then, the system public key would be pk = (u, g, v, spk). In summary, the running result of the algorithm KeyGen is (sk, pk) = {(a, ssk), (u, g, v, spk)}.

5.1.2 Filepro2C: The DO runs Filepro2C to pre-process files to be outsourced to the cloud. First, the DO divides *F* into blocks or even sectors, where *F* represents the cloud file, as mentioned above. For simplicity, we only divide *F* into *n* blocks here:

$F = \{m1, m2, \cdots, mi, \cdots, mn\}$, where *mi* is the general name of data blocks, and $i \in [1, n]$. Then, the DO generates an authenticator for each block *mi* : $\sigma i = (h(Vi \,||Ti) \cdot umi)a$, of which the aggregated set is

$\sigma = \{\sigma i\}i \in [1,n]$.

In the above equation, the mentioned *Vi* is *mi* 's version number, and *Ti* is its time stamp. Moreover, the DO generates file tags based on *ssk* to ensure the integrity of the unique file information:

$\vartheta = UI\ D||FI\ D||SIG(UI\ D||FI\ D)ssk.$

Finally, the DO uploads $\{F, \sigma, \vartheta\}$ to the cloud for storage. At this point, the pre-processing tasks for the CSP have all been completed.

5.1.3 Filepro2T: The DO runs Filepro2T to pre-process information to be stored in the TPA. Specifically, the file information and the block information are required for the *doubly linked info table*, while the specific location of each block is required for the *location array*. Specifically, *FI D*, *UI D*, *Vi* , *Ti*, and *Loci* are uploaded by the DO to the TPA. Upon receiving all the information, the TPA establishes the DLIT and the LA and then saves them. At this point, the pre-processing tasks for the TPA have all been completed.

5.2 Verify Phase: The second phase performs data verification, therein involving the DO, CSP, and TPA. Specifically, challenges are launched from the TPA to the CSP in Chal-Gen, after which the CSP responds in ProofGen to prove the integrity of the cloud data for which it is responsible. In VerifyProof, auditing results are obtained from the TPA's calculation and returned to the DO. The detailed design of each algorithm is as follows.

5.2.1 ChalGen: The TPA runs ChalGen to launch the verification challenge to the CSP on behalf of the DO. First, the DO delegates the verification task of a certain file to the TPA. Then, the TPA asks the CSP for the corresponding file tag ϑ and verifies the correctness of it by *spk*. If this fails, the TPA would inform the DO that the file has been corrupted; otherwise, verification continues. Then, the TPA picks *a random* element *ax,y* in the LA and obtains the corresponding blocks' information in the DLIT. Finally, the TPA sends the verification challenge, $chal = \{i, ri\}i \in [1,s]$

to the CSP, where $s \in [1, n]$ and *ri* is randomly picked from *Z p*.

5.2.2 ProofGen: The CSP runs ProofGen to generate corresponding proofs of the required blocks, which contain two parts: a tag proof indicating the authenticator's correctness and a data proof indicating the data's integrity. Upon completion, the complete proof *will* be sent back to the TPA as the response of the verification
Challenge.

5.2.3 *VerifyProof:* The TPA runs VerifyProof to check whether the proof returned from the CSP is valid. According to the data information stored in the DLIT, the TPA computes $DIi = e(h(Vi \,||Ti), v)$ for each data block to be verified. Finally, the TPA checks whether $e(T, g) = DI \cdot e(uD, v)$ holds. If it does, the data outsourced to the cloud is complete; otherwise, the data are not complete.

5.3 Global and Sampling Verification: In our design, the verify phase includes the global verification and sampling verification. Specifically, the sampling verification is the common one described in the existing auditing schemes, while the global verification is the concept proposed in this paper. In the verification process of the protocol designed in this paper, the time stamp participating in is generated by the DO itself as well. That is where the problem lies. Specifically, the TPA is a trustworthy third party chosen by the DO to undertake the auditing task, which is responsible for the DO to monitor the CSP. In other words, the TPA and the DO are on the same side. There is a strong possibility that the DO will collude with the TPA by uploading a data block mismatching the time stamp. So that, the uploaded data block will never pass the data auditing, and the DO and the TPA can carve up the fraudulent compensation for data corruption from the CSP.

Therefore, we introduce the concept of "global verification" in our protocol to address such a collusion attack mentioned above. Specifically, at the moment that the data file is uploaded or updated in the cloud, all this data should be verified, which is the meaning of "global". If these data uploaded can pass the verification, the CSP will provide storage services to them; otherwise, the CSP will refuse to store these data. After the global verification, the concerns of the CSP can be reduced. Certainly, the global verification requires some additional operations in the verify phase. In the ChalGen algorithm, the TPA inserts all blocks' indexes in *chal.*

chal = $\{i, ri\}i \in [1,n]$ when the whole file is uploaded or *chal* = $\{i, ri\}i \in [1,c]$ when updates appear, where c is the total number of changed blocks. Then, the CSP responds with the corresponding proof in ProofGen, and the follow-up process remains unchanged.

5.4 Batch Auditing: In real-world scenario, the TPA may not serve only one DO or even be responsible for only one data file. Multiple auditing tasks from various DO delegations may be waiting for processing simultaneously. Hence, batch auditing is introduced into our design to address the above phenomenon. It's support in our work is divided into two categories: one for multiple files from one DO and the other for multiple files from multiple DOs. We suppose that ui $\in$ [1, us] users delegate fi $\in$ [1, fs] files' i $\in$ [1, s] blocks' auditing tasks to the TPA simultaneously.

5.5 Dynamic Auditing: To support efficient dynamic operations on data along with its information, we design DLIT and LA. Changes in the two objects have been mentioned briefly in Section 4 when file-layer updates occur. Specific operations are introduced, involving block insertion (*Binsert*), block deletion (*Bdelete*) and block modification (*Bmodi f y*).

6. CONCLUSION

In this work, with a very unique dynamic structure composed of a doubly linked info table and a location array, a public auditing protocol is proposed. Compared with the state of the art, an appropriate relationship between the DLIT and thus the LA makes our protocol perform better both in terms of efficient dynamic support and reduced overhead. Moreover, some basic challenges in cloud auditing, like batch auditing, blockless verification and lazy update, are conquered by our protocol. The protection of our protocol is indicated by the sufficient theoretical proofs available.. Extensive numerical analysis and experimental comparison results are utilized to validate the performance of our protocol, which makes it more convincing. Hence through this work we expect secured data access and faster data transformation processes like insert, delete and update.

REFERENCES

[1] P. Mell and T. Grance, "The NIST definition of cloud computing," Nat. Inst. Standards Technol., vol. 53, no. 6, p. 50, 2011.
[2] R. Buyya, C. S. Yeo, S. Venugopal, J. Broberg, and I. Brandic, "Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility," Future Generat. Comput. Syst., vol. 25, no. 6, pp. 599–616, 2009.
[3] K. Yang and X. Jia, "Data storage auditing service in cloud computing: Challenges, methods and opportunities," World Wide Web, vol. 15, no. 4, pp. 409–428, 2012.
[4] Z. Xia, X. Wang, X. Sun, and Q. Wang, "A secure and dynamic multi keyword ranked search scheme over encrypted cloud data," IEEE Trans. Parallel Distrib. Syst., vol. 27, no. 2, pp. 340–352, Feb. 2016.
[5] Z. Fu, X. Sun, Q. Liu, L. Zhou, and J. Shu, "Achieving efficient cloud search services: Multi-keyword ranked search over encrypted cloud data supporting parallel computing," IEICE Trans. Commun., vol. 98, no. 1, pp. 190–200, 2015.
[6] J. Shen, H. Tan, S. Moh, I. Chung, Q. Liu, and X. Sun, "Enhanced secure sensor association and key management in wireless body area networks," J. Commun. Netw., vol. 17, no. 5, pp. 453–462, 2015
[7] M. Green, "The threat in the cloud," IEEE Security Privacy, vol. 11, no. 1, pp. 86–89, Jan./Feb. 2013.
[8] Q. Jiang, J. Ma, and F. Wei, "On the security of a privacy-aware authentication scheme for distributed mobile cloud computing services," IEEE Syst. J., to be published.
[9] D. A. B. Fernandes, L. F. B. Soares, J. V. Gomes, M. M. Freire, and P. R. M. Inácio, "Security issues in cloud environments: A survey," Int. J. Inf. Secur., vol. 13, no. 2, pp. 113–170, Apr. 2014. [10] L. F. B. Soares, D. A. B. Fernandes, J. V. Gomes, M. M. Freire, and P. R. M. Inácio, Cloud Security: State of the Art. Berlin, Germany: Springer, 2014