DATA ENCRYPTION STANDARD FOR AUDITING APPLICATION IN CLOUD STORAGE

M. Ramkumar and R. Krishnakumar

Department of Computer Science and Engineering, Gnanamani College of Technology, India

Abstract

In recent years, cloud computing has become paramount and the security problems associated with the cloud model are also increasing. To mitigate the threats associated with cloud security, existing systems use an isolated authentication method to audit by third parties or personal auditing. In this method, to direct that audit, the data owners need to be online. This article discusses a novel method using the personal auditing that is supported on the generation of One Time Password (OTP). Using this generation or creation technique of OTP (secret key) the documents can be protected from fraudulent users. Data Encryption Standard (DES) algorithm assists the owner in uploading the file. The files will be converted into smaller units and piled up in different instances of the cloud. For enhanced security, both the files as well as the content will be encrypted and stored in various instances on the cloud server. This authentication process can prevent the errant user's attempt to hack at the cloud end. The OTP will be generated and propelled to the user upon download. This OTP has to be used to verify the user before downloading the required files.

Keywords:

Cloud Computing, Personal Auditing, Security Issues

1. INTRODUCTION

Using this technique to generate or create an OTP (secret key), the documents can be protected from fraudulent users. Data Encryption Standard (DES) algorithm assists the owner in uploading the file. The files will be converted into smaller units and piled up in different instances of the cloud. For enhanced security, both the files as well as the content will be encrypted and stored in various instances on the cloud server. This authentication process can prevent the errant user's attempt to hack at the cloud end. The OTP will be generated and propelled to the user upon download. This OTP has to be used before downloading required files to verify the user [1] [2] [3].

From the user's point of view, collecting data in a remote cloud environment using a supple on-demand mode that is accessible for both people and IT endeavors, is a great advantage in terms of releasing the load for running storage, widespread access to data with local independence, and escaping capital outflows on hardware, software, and employee maintenance, etc. [6] [7]. While cloud computing creates these benefits more seductive than it has yet, it also carries novel and challenging security threats just before data is outsourced by users [8].

Due to the fact that cloud service providers (CSP) are dividing managerial units, data outsourcing prevents users from directly accessing their data [9]. First of all, although the cloud infrastructures are much more powerful and reliable than personal computing devices, they still face the wide choice of both domestic and peripheral data reliability threats [10].

With the slow endorsement of the claim, any private information in the cloud computing conveniences can be established on any tackle. The forthcoming parts of this study strives to define principles in the designing of cloud computing services to ensure that the user memorandum and protective information would not be tricked out [11] [12], and that the user information is defended from disclosure.

In order to solve the regeneration difficulty of failed authenticators in the absence of data owners, the author starts a substitute for the conventional public audit scheme, which is advantageous to restore authenticators. A novel, demonstrable public authenticator, is generated by a pair of keys and restored using fractional keys. This scheme can thus completely free data owners from online trouble.

In cloud computing, to secure and dependable storage services. Towards addressing Safety hazards to data accuracy in the cloud, a safe and reliable cloud storage service is implemented to concentrate on this new challenge and auxiliary. To take care of the data error localization problem, only binary consequences are given for the authentication of the storage. We are looking at the data security challenge in cloud data storage, which is basically a dispersed storage system. It gives no assurance on honesty and accessibility of the data. This problem, if not addressed accurately, may obstruct the successful architectural design operation of the cloud [4].

This does not completely decipher the crisis of securing unprotected data, but merely diminishes it to one of the organizations. Illegal outflow of data remains a dilemma due to the potential experience with encryption keys [5]. The auditing outcome should not only acknowledge the accuracy of the data but also be clever in concluding that the entity is responsible for the difficulty that may occur.

The process used here does not create it optimally sparse and directly over random linear coding to shortened communication, storage and computation price. The key benefit of decentralized erasure codes is that there is no harmonization requirement between the data nodes [13]. We demonstrate that data nodes performing arbitrarily and separately, and with sparse construction, can make excellent erasure codes.

Randomized network algorithm and Wiedemann algorithm. The key problem here is accessibility. So, this vigorous dispersed storage with negligible computation and communication is unlikely to be realized. A Generic Proxy Cryptography Construction. In the projected main auditing support scheme, users can securely delegate integrity checking tasks to third-party auditors (TPA) and are free to use cloud storage. TPA should be able to audit cloud data storage efficiently without requiring local data copies, and they do not impose any additional online burden on the cloud user.

Method for regenerating-code-based cloud storage by third party public audit (TPA). To solve the renewal challenge of botched authenticators in the absence of data owners, we initiate a surrogate, which is advantageous for regenerating authenticators

into the customary replica of the public audit method. A new demonstrable public authenticator created by a pair of keys that can be regenerated with partial keys. Thus, online saddle this system can save the data owners entirely. Additionally, to conserve data privacy, the program coefficients could be randomized with pseudorandom codes. Wide-ranging safety analysis shows that new design is verifiably secured under a random oracle model and tentative evaluation that specifies that the proposed scheme is highly competent and can be incorporated into the reviving code-based cloud storage services [14].

The cryptographic systems for the data protection principle cannot be directly controlled by the user. Accordingly, authentication of accurate data storage in the cloud must be carried out without unambiguous knowledge of all data. Given the different data types for each client stored in the cloud, and the demand for long-term constant promises their data security, the difficulty of validating data storage accuracy in the cloud becomes even more demanding. This is not merely a warehouse of data from third parties. Clients may frequently rationalize the data stored in the cloud.

2. PROPOSED METHODOLOGY

For the users we suggested an effective OTP-based personal audit. It can do away with users of fraud. We use the DES algorithm to allocate the file to smaller units and pile up in the cloud in various storage systems, and we can access the file from cloud services. Here we have applied upload encryption and download file decryption. The cloud storage audit system which shows the owner privileges in personal auditing. The proprietors create the data and host it in the cloud.

2.1 USER INTERFACE

Users are able to generate the account themselves and for that they have separate pages, that page consists of user particulars. So it allows only the approved users to access the new system through the method of verification. Users are able to generate the account themselves and for that they have separate pages, that page consists of users' particulars. So it allows only the approved users to access the new system through the method of verification.

2.2 SECRET KEY GENERATION

At first, while uploading the file, the OTP (secret key) will be created as the preliminary step, and when the file is uploaded, we will have the sole secret key. Each file will take this key as credit. The OTP (secret key) is created to use for process upload and download. If the user wishes to download any file and gives the download request, that file's OTP (secret key) will be generated after the owner authenticates. The generated OTP will be sent to the user to check his credentials and thereby avoid to some extent the entry of errant users.

2.3 FILE UPLOADING PROCESS

File uploading is only possible after the owner authorization has been checked. When the author establishes his access with his login details on the cloud, he acquires separate OTPs (secret keys) for the corresponding mail identity to ensure that the approved owner uploaded them.

2.4 MAIL ALERT PROCESS

Initially the user receives the OTP (secret key) through the email for the uploading and downloading process. Then the secret key is sent for accessing the encrypted data stored in the server and for decrypting the data file downloaded from the server storage system.

2.5 FILE DOWNLOADING PROCESS

For downloading the file, the user has to acquire the OTP (secret key) to their corresponding email, and then file can be decrypted using it.

3. DES ALGORITHM

The DES algorithm allows for approval of the exact types of computations on the cipher text and for an encrypted result when decrypted matches the outcome of operations performed on the plaintext. The data in the cloud is encrypted before it is propelled, the operations are performed in the encrypted data and the results are decrypted, it is identical to the operations carried out on the original data. This encryption cryptosystem offers privacy and data protection. The cipher texts then proliferate in incredible fallouts which, when decrypted, is the same as plaintext.

An algorithm is symmetric when prepared for the purposes of calculation, subtraction, and multiplication. We employ multiplication in the proposed method while encrypting the data. This algorithm is made up of the following steps: key generation, encryption and decryption. With this algorithm the file is transformed into smaller units. The data owner encrypts its files and keywords and piles them up in different cloud storage. The number of times the files have to be encrypted can be specified, and the cloud service provider has to do as required when the data owner recovers the file.

Without knowing anything about the files, the cloud service provider will execute computations on the encrypted data. It will propel the data owner to rear the results. Encryption algorithm plays a very important role in giving secure communication over the network. It is the basic tool for data defense. Encryption algorithm uses a key to change the data into scrambled form and then the user has a key to decrypt the data. In DES, data encryption and decryption are done using the same key.

4. PERSONAL AUDITING (PA)

The Personal Auditing (PA) is responsible for validating a user and for issuing user feature keys. If the user is allowed to access the attributes, the secret code key will be generated during the protected protocol that connects the PA and the data warehouse. They employ the secure arithmetic protocol with their own master secret keys, and give a user unreliable key components. Then, with key components received separately from authorities, the user can provoke the entire secret keys.

The secure protocol deters them from knowing the master secrets of each other, so that no one can evoke the user's entire secret keys alone. In this setting, the data storage center knows the revocation list and does not infringe the security needs, as it is only permissible to re-encrypt the cipher texts and is unable to obtain any features about the user's attribute keys. In this section,

the proposed scheme is based on recapitulating certain definitions into our construction, such as access tree, encryption, and decryption algorithms.

Fig.1. Personal Auditing

The Fig.1 shows diagrammatic representation of PA. This architecture guarantees the owners more security. It can reduce the complexity of time as well as the security. Security analysis with personal audit security architecture meets security requirements for authentication, data reliability, and secrecy, which follows directly from the use in our system of the typical cryptographic primeval, essentially message authentication code, and encryption. The erring users can only be repudiated if the client can give a special representation of the trusted authority (TA) that he knows about.

PA performs a cloud storage expertise audit system as shown below; normally it involves data owners (owner) and the cloud server (server). The owners store the data and transmit the data into the cloud servers. The cloud servers keep data from the owners and get the users (data consumers) access to the data. The auditor is a trusted party with expertise and capabilities to provide both proprietors and cloud servers with a data storage auditing examination. The auditor can be a trusted government-managed organization that can deliver impartial audit results for mutual data proprietors and cloud servers.

Three types of parties interact in the protocol: the Data Owner, the Cloud Service Provider, and Content Providers. The protocol comprises the following main components.

Key Generation. The HE is run by the Data Owner. Keygen algorithm for the homomorphic encryption scheme is either a private key or a public key version. The Data Owner shares the private encryption key with the Content Providers for the private key version, and all store the key securely locally. The Data Owner publishes the public key for the public key version, and stores the private key locally securely.

Encryption of Training Data. Content providers encrypt confidential data which is labeled for uploading to the cloud. It is equal to the symmetric version's secret key, and in the asymmetric version of the scheme, the public key. Alternatively, the content providers may encrypt pre-processed versions of the training set data for each training vector class, e.g. synthetic data such as class sums or class-conditional covariance matrices (i.e. sufficient statistics).

Training. The Cloud Service Provider calculates an encrypted Learned Model. This means that the homomorphic encryption algorithm evaluates the machine learning training phase. Train the encrypted training vectors homomorphically. The encrypted form of the Learned Model is stored by the Cloud Service Provider and can be returned to the Data Owner upon request.

Classification. An encryption that was not normally used in the training stage, is sent by the data owner or the content providers to the cloud service provider. The ML classification phase is assessed by the cloud service provider. The encrypted learned model is used to classify the machine learning task on the encrypted test vector, and the encrypted classifications is returned to the Data Owner. The Data Owner decrypts the results to obtain the ratings.

Verification of the Learned Model. The Data Owner tests the Learned Model as probabilistic. The Data Owner encrypts known classification test vectors and sends the cipher texts to the cloud service provider. The Cloud Service Provider homomorphically classifies the encrypted vectors, returning encrypted classification results to the Data Owner. The Data Owner decrypts the results and compares the test error of the Learned Model in the Cloud to the known classification labels.

Security Considerations. The protocol assumes a model in which the cloud is an honest but curious party, i.e. the cloud will follow the protocol to provide the desired functionality and will not deviate or fail to deliver the service or return results. But, it is curious in the sense that it would look at the available information. This assumption is reasonable to model a rational, economically motivated cloud service provider: the cloud is motivated to provide excellent service, yet it would be motivated to take advantage of the additional information available. A malicious Cloud is a much stronger opponent who would potentially mishandle calculations, delete data, refuse to return results, collide with other parties, etc. In most of these malicious behaviors, the Cloud is likely to get caught and thus damage its reputation if it tries to run a successful business.

The verification step we are proposing is analogous to a naive version of the protocols Proof-of-Storage (PoS). Verification requires the Data Owner to store a certain number of labeled samples locally so that the computations of the Cloud can be tested correctly (and determine test errors). The Data Owner encrypts the test vectors and queries the cloud after the training stage in order to provide encrypted classifications of the test vectors, and then the Data Owner decrypts and compares with the correct label. Since we assume an honest but curious Cloud model, the Data Owner only needs to store enough test vectors to determine the cloud test error (or to detect any accidental error). We also implicitly assume that Content Providers do not behave maliciously and correctly by encrypting and uploading data.

For the functionality required, the cloud must necessarily learn a certain amount of information. The cloud computes uses an encrypted Learned Model from a Stage 1 collection of encrypted and labeled training vectors and provides encrypted classifications of encrypted test vectors in Stage 2. This includes knowing how many vectors were used in the training phase, and how many test vectors were submitted for classification. Additionally, our scheme reveals the number of vectors within each class, and an upper bound on the test vector entries can also be deduced once the parameters for the HE scheme and the number of test vectors are known. The HE schemes are assumed to be randomized and have semantic security against passive adversaries, a property that ensures an adversary is unable to distinguish between a message encryption and another. The Cloud handles encrypted data and performs HE operations, and can encrypt messages of your choice in the public key setting.

5. RESULTS

The performance results using the machine learning technique is explained here in terms of data encryption and data decryption.

Data Encryption: The data owner first separates the data into numerous components according to logical granularities before outsourcing the data to the cloud.

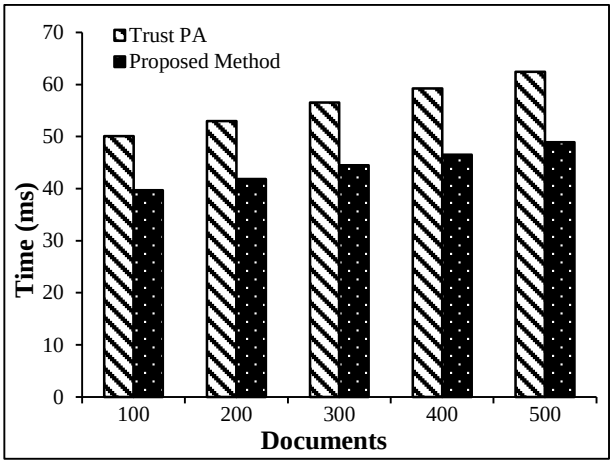

Fig.2. Encryption Time

In Fig.2, 20 documents were encrypted with 5s, in which 40 documents were encrypted in 12s. The time used for encryption is shown as 20s when it has reached 60 files. Similarly the model took 27 and 31s to encrypt 80 and 100 documents respectively. The time variation is dependent on the document size.

Data Decryption: After the OTP (secret keys) is obtained and the user sends the cipher text to the owner, the owner uses DES algorithm for encryption. DES decryption is used to acquire the original plaintext which the server has relocated. Consequently, the user receives the file proposed in a secure way.

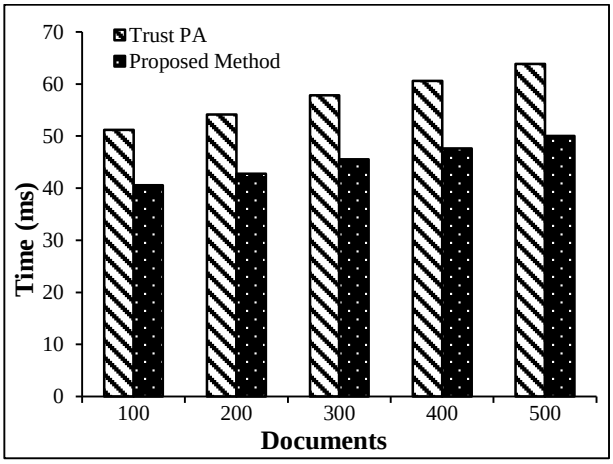

Fig.3. Decryption Time

The decryption performance is shown on Fig.3, where the decryption time is comparatively smaller than the time taken for encryption. As far as time is concerned, the performance of third party auditing and OTP-based Personal Auditing are discussed here. In Fig.2 and Fig.3, the existing model took 150s for downloading 5 files, where the proposed model took 130s. For 20 files, the time difference between the two approaches is seen. The new model used nearly 25s less in each point than the existing model.

6. CONCLUSION

In this paper, we propose a privacy-preserving personal auditing system in cloud computing for data storage security. Personal auditing based on OTP tackled security of storage and time management (OTP-based personal auditing) validates the user by combining the secured parameter and the secret key. By setting the validity of the OTP, the possibility of sharing the secret key is avoided in this work. Hacking and sharing mails are also reduced to a certain extent by checking the combination of secured parameter and OTP. Performance time is also analyzed and compared to the existing model for third party auditing. The innovative model uses lesser time to decrypt 20 files compared to the existing model.

REFERENCES

[1] D.C. Chou, "Cloud Computing Risk and Audit Issues", *Computer Standards and Interfaces*, Vol. 42, pp. 137-142, 2015.

[2] B. Schneier, "*Applied Cryptography*", John Wiley and Sons, 1996.

[3] E. Bacis, S.D.C. Di Vimercati and S. Paraboschi, "Dynamic Allocation for Resource Protection in Decentralized Cloud Storage", *Proceedings of IEEE International Conference on Global Communications*, pp. 1-6, 2019.

[4] K. Ren, C. Wang and Q. Wang, "Security Challenges for the Public Cloud", *IEEE Internet Computing*, Vol. 16, No. 1, pp. 69-73, 2012.

[5] D. Song, E. Shi, L. Fischer and U. Shankar, "Cloud Data Protection for the Masses", *Computer*, Vol. 45, No. 1, pp. 39-45, 2012.

[6] M. Balaganesh, R. Sureshkumar and J. Venkateshan, "A Survey on Techniques for Third Party Auditor in Cloud Computing", *International Journal of Emerging Technology and Advanced Engineering*, Vol. 3, No. 12, pp. 372-389, 2013.

[7] S. Mahdavi Hezavehi, Y. Alimardani and R. Rahmani, "An Efficient Framework for a Third-Party Auditor in Cloud Computing Environments", *The Computer Journal*, Vol. 3, No. 2, pp. 1-12, 2019.

[8] S.M. Hezavehi and R. Rahmani, "An Anomaly-Based Framework for Mitigating Effects of DDoS Attacks using a Third-Party Auditor in Cloud Computing Environments, *Cluster Computing*, Vol. 23, No. 1, pp. 1-19, 2020.

[9] V. Sharma, "Determine Dishonest Role of Third-Party Auditor in Cloud Computing", *Technology*, Vol. 1, No. 1, pp. 1-12, 2019.

[10] R. Saxena and S. Dey, "DDoS Prevention using Third Party Auditor in Cloud Computing", *Iran Journal of Computer Science*, Vol. 2, No. 4, pp. 231-244, 2019.

[11] B.P. Gajendra and V.K. Singh, "Achieving Cloud Security using Third Party Auditor, MD5 and Identity-based Encryption", *Proceedings of International Conference on Computing, Communication and Automation*, pp. 1304-1309, 2016.

[12] S. Hiremath and S. Kunte, "A Novel Data Auditing Approach to Achieve Data Privacy and Data Integrity in Cloud Computing", *Proceedings of International Conference on Electrical, Electronics, Communication,*

Computer, and Optimization Techniques, pp. 306-310, 2017.

[13] K.S. Reddy and M. Balaraju, "Comparative Study on Trustee of Third-Party Auditor to Provide Integrity and Security in Cloud Computing", *Materials Today: Proceedings*, Vol. 5, No. 1, pp. 557-564, 2018.

[14] S. More and S. Chaudhari, "Third Party Public Auditing Scheme for Cloud Storage", *Procedia Computer Science*, Vol. 79, pp. 69-76, 2016.