# Ensuring the Data Integrity and Confidentiality in Cloud Storage Using Hash Function and TPA

Swapna V. Tikore
Computer Science and Engineering Department
SIETC
Paniv, India
*tikore_swapna@rediffmail.com*

Deshmukh Pradeep K.
Computer Science and Engineering Department
SIETC
Paniv, India
*principalsietc@gmail.com*

Dhainje Prakash B.
Computer Science and Engineering Department
SIETC
Paniv, India
*dhainjeprakash@gmail.com*

***Abstract***—Main call for Cloud computing is that users only utilize what they required and only pay for whatever they are using. Mobile Cloud Computing refers to an infrastructure where data processing and storage can happen away from mobile device. Research estimates that mobile subscribers worldwide will reach 15 billion by the end of 2014 and 18 billion by at the ending of 2016. Due to increasing use of mobile devices the requirement of cloud computing in mobile devices arise, which evolves Mobile Cloud Computing. Mobile devices require large storage capacity and maximum CPU speed. As we are storing data on cloud there is an issue of data security. As there is risk associated with data storage many IT professionals are not showing their interest towards Mobile Cloud Computing. To ensure the users' data correctness in the cloud, here we are proposing an effective mechanism with salient feature of data integrity and confidentiality. This paper proposed a solution which uses the RSA algorithm and mechanism of hash function along with various cryptography tools to provide better security to the data stored on the cloud. This model can not only solve the problem of storage of massive data, but also make sure that it will give data access control mechanisms and ensure sharing data files with confidentiality and integrity.

***Keywords***- *Cloud; confidentiality; data security; data storage; integrity; mobile cloud computing;*

*****

## I. INTRODUCTION

Since cloud computing is an evolving paradigm, data centralization or outsourcing to cloud becomes a trend. Storing data remotely in the cloud in a flexible on-demand manner brings appealing benefits in terms of storage and computation. Large numbers of clients are storing their important documents in remote servers in the cloud, without even keeping a copy in their local machines. Sometimes the data stored in the cloud is so important that the clients must ensure it is not lost or corrupted. Even though it is easy to check data integrity after completely downloading the data to be checked, it is not a practical solution due to the expensiveness in I/O and transmission cost across the network. A lot of works have been done on designing remote data integrity verifying protocols, which can access data integrity to be checked without completely downloading the data. But all these methods deal with the integrity of encrypted text or plain text. The issue is that performing computations on encrypted data is a difficult task. Instead, data can be anonymized to enhance privacy. Anonymization refers to a privacy preservation technique that translates data so as to make the data worthless to anyone except the data owner. Statistical computations are possible on anonymized data without concern that other individuals may capture the data. Hiding, hashing, permutation, shift, substitution, enumeration and truncation are some of the traditional techniques to obscure data. In the anonymization scheme proposed by Intel, anonymization takes place on the VMs sending the data and deanonymization within a secure enclave. But this scheme is not feasible, especially for thin clients as it takes more computational power of client. A better efficient data anonymization scheme is proposed which saves the computational power and storage space of the client by performing anonymization and deanonymization within the secure enclave. Also, the remote integrity checking protocol to check the integrity of anonymized data is explored. Today, the use of mobile phones is increasing day by day. Everyone has a mobile phone which provides the facility to move anywhere and access the data anytime. The increasing use of mobile devices gave birth to Mobile Cloud Computing (MCC). MCC is the like marriage between mobile web and cloud computing. MCC provides new type of the services to mobile users to fully utilize the advantages of Cloud Computing.

Here sensitive data is stored and processed outside the mobile devices on a centralized computing platform located in clouds. The main issue in using mobile cloud computing is securing the data of mobile user stored on mobile cloud. The data/file of a mobile user is very sensitive; any unauthorized person can do changes in it, to harm the data. So the main concern of cloud service provider is to provide the security of data/files created and manipulated on a mobile device or cloud server. The data/file security is very essential for owner of the data/file as it can contain any confidential information of his. For user the integrity of the data is very important. If any unauthorized person performs changes in data of other person then it can harm the integrity of the data. Any person after finding confidential information of other person can harm that person. So, data confidentiality is also a concern of data owner. To protect data of user, encryption is used to secure data in the cloud.

New service architectures are necessary to address the security concerns of the mobile users for using mobile cloud

techniques. A number of schemes have been provided but there is lack of any concrete framework presented till now which is up to the remarkable position. A number of solutions have been provided by a number of researchers and many are still working on it. In this research paper the ultimate objective is to evolve an integrated framework/solution for achieving the mobile data security on cloud environment in various possible conditions, so that this technology may be implemented in applications of versatile nature without any flaw.

## II. LITERATURE SURVEY

Mobile Cloud Computing is a service that allows resource constrained mobile users to adaptively adjust processing and storage capabilities by transparently partitioning and offloading the computationally intensive and storage demanding jobs on traditional cloud resources by providing ubiquitous wireless access [2].

Some organizations, for example Google, watch Mobile Cloud Computing as a new paradigm for mobile applications whereby most of the processing and data storage associated with the applications IS moved off the mobile to high powerful, centralized platforms of computing, located in the Cloud [3].

According to survey 74% of IT Executives and Chief information Officers are not willing to adopt cloud services due to the risks associated with security and privacy [4]. In [1], a Provable Data Possession (PDP) model based on the concept of SA-based homomorphic verifiable tags for remote data checking is constructed. The client pre-processes the file, generates a piece of metadata that is stored locally, sends the file to the server and may delete its present local copy. The file is then stored and responded to challenges issued by the client. Using its local metadata, the client verifies the response. In [4], a new scheme is proposed which allows a third party auditor (TPA), apart from the cloud client, verifies the integrity of the dynamic data stored in the cloud. It uses the classic Merkle Hash Tree construction for block tag authentication. A Merkle Hash Tree (MHT) is a well-studied structure of authentication, which is intended to efficient and secure prove that a set of elements are undamaged and unaltered. Binary tree is constructed where the leaves in the MHT are the hashes of authentic data values.

## III. PROPOSED SYSTEM

RSA Algorithm RSA is a commonly adopted public key cryptography algorithm. RSA can be used for public and private key exchange, generated digital signatures, or encryption of small size blocks of data. RSA algorithm intended to use a variable size encryption block and a variable size key. RSA has been widely used for establishing secure communication channels and for authentication and the identity of service provider over insecure medium of communication. In this authentication scheme, the serveris binded to implements public key authentication with client by signing a unique message from the client with its private key, thus creating what is called a digital signature[5]. In the proposed scheme RSA algorithm is used to find out the key pair for both Mobile User and TPA, these keys are used to encrypt and decrypt the file.

Hash Function. A cryptographic hash function takes a message of arbitrary length and creates a message digest of fixed length. A hash function produces a short and fixed length message digest, which is unique for each message. The main requirements for the security of hash functions are that they must be one-way functions, and they must be collision resistant [6]. A collision occurs when for two different inputs, the hash function gives the same output, for instance, Hash (mI) = Hash (m2).

Here in the proposed scheme hash of the file is calculated so that integrity can be maintained. DES Algorithm The Data Encryption Standard (DES) is a symmetric-key block cipher, having a 64-bit block size and a 56-bit key. At the encryption site, DES takes a 64-bit plaintext and creates 64-bit cipher text; at the decryption site, DES takes a 64-bit cipher text and creates a 64-bit block of plain-text.

The same 56-bit cipher key is used for both encryption and decryption [7]. In the proposed scheme TPA uses the DES algorithm to provide better security to the file of mobile user. TPA perform DES algorithm on the file before sending it to the cloud for storage. By using DES more security can be provided to the user's data. Proposed Mechanism Here a mechanism/scheme is proposed to provide secure data storage in Mobile Cloud Computing. This proposal uses the concept of Hash function along with several cryptographic tools to provide better security to the data stored on the mobile cloud. Here we also have a Trusted Third Party Auditor (TPA) who is very well trusted. TPA checks the integrity of the data stored on mobile cloud on behalf of the data owner. TPA checks the hash and message to verify the integrity of the data. The Integrity Verification is provided by the TPA which reduces a lot of work of the mobile user. In this scheme data owner has two keys, one of which is only known to him called private key and another is public key. Here message/file is encrypted twice firstly, by owner's private key and secondly by public key of TPA. So this provides the confidentiality to the data of mobile user. In proposed method RSA algorithm is used for performing encryption and decryption which provides message authentication. Here the hash function of the message is also calculated to provide security to the data. The proposed architecture is shown in Fig.I.

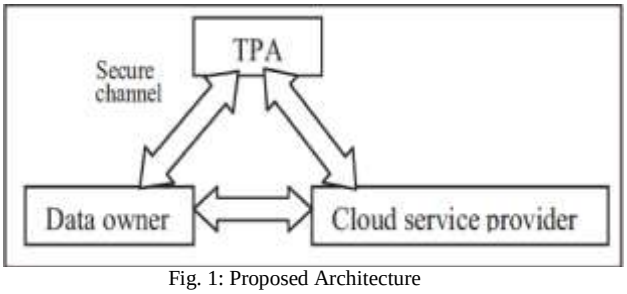

Fig. 1: Proposed Architecture

*1) Key Generation:* Data Data Owner uses RSA algorithm for generation of combination of public and private key for himself. TPA also uses RSA algorithm for generating key pair for its own. The private key of TPA is pkl and of Data Owner is pk2, while public key of TPA is dl and public key of data owner is d2.

*2) Key Sharing:* Key set of TPA: {pkl, dl} at TPA Key set of DO: {pk2, d2} at Mobile device. Here only public key of

TPA is exchanged between DO and TPA using secure channel.

*3) Encryption:* Firstly, At first, owner of data encrypt the message/ file (F) using his public key (d2) E(F,d2) and then generate the hash of encrypted message H(E(F,d2)). Now, the encrypted file is re-encrypted with public key (d l) of TPA E(E(F,d2),dl). After that the hash is re-encrypted with public key of TPA (dl) E(H(E(F,d2)),dl). Now, these two packages are appended and the result E(E(F,d2),dl) II E(H(E(F,d2)),dl) is sent to TPA. The encrypted Hash function of the message is stored by TPA to ensure the data integrity. TPA decrypts the package E(E(F,d2),dl) received, by its private key. TPA generates a random key for performing encryption on the message E(F,d2) generated after encryption. TPA uses DES (Data Encryption Standard) for performing encryption to provide better security. This generated random key is stored by TPA for performing decryption in future. The result is send to the cloud for storage.

*4) Decryption:* When required to verify the data correctness, the encrypted package {Encrypt(E(F ,d2))} after DES operation stored on cloud is send to TPA. TPA firstly decrypts the message by random key stored by him. Then TPA generates the Hash of the encrypted file obtained from cloud. Now, TPA decrypts the hash value stored by it, this decrypted hash value is compared with the one generated by it. Then according to the result obtained TPA sends file to owner indicating the correctness or not and the requested file. Here the file transferred to owner is encrypted by his public key so that only owner can decrypt it. Owner after receiving encrypted file, decrypt it by private key of himself. Here the algorithms involved in the proposed scheme are shown. Algorithm 1 shows the set up phase which includes the operations from starting to the storing data on the cloud.

**ALGORITHM 1: Set Up**

TPA: pk1, d1= GenKey()
Client: Pk2, d2= GenKey()
TPA → Client: d1
Client: F'= (E (F, d2)), H(F') = H(E (F, d2)) F"= E (F', d1), H'(F')=E (H(F'), d1)
Client → TPA: F"l H(F')
TPA: Store(H'(F')), k=Random(), F'= D(F",pk1), F"=Encrypt(F',k)
TPA → CSP: F"
CSP: Store(F")

**ALGORITHM 2: Verification**

CSP → TPA: F"
TPA: F'= D (F", k), newH'(F')=H(F'), retrieve( H'(F')), H(F')=D( H'(F'),pk1), Result= Compare (H(F'), newH'(F'))
TPA → Client: Send (Result)

**ALGORITHM 3: Message Retrieval**

Client → TPA: Request (F)
TPA → CSP: Request (F")
CSP → TPA: Send (F")
TPA: Verification (F"), F'=E (F, d2)
TPA → Client: Send (F')

Algorithm 2 is the verification, which shows how the integrity of the file/message is performed by the TPA and Algorithm 3 shows the file retrieval process which shows how the requested file by owner is transferred to him from Cloud Service Provider through TPA. Fig.2. shows interaction among data owner, Third Party Auditor (TPA) and Cloud Service Provider (CSP)

## IV. SECURITY AND PERFORMANCE ANALYSIS

Here, the proposed scheme is validated by performing two kinds of analyses Security Analysis and Performance Analysis. In security analysis the proposed scheme is analyzed by the security threats which may be possible. Here the mechanism is analyzed against its correctness and whether it provides confidentiality and security to the data of mobile user. In performance analysis the performance of proposed scheme is analyzed with respect to the number of operations involves in the scheme as well as the storage requirement of the scheme.

.

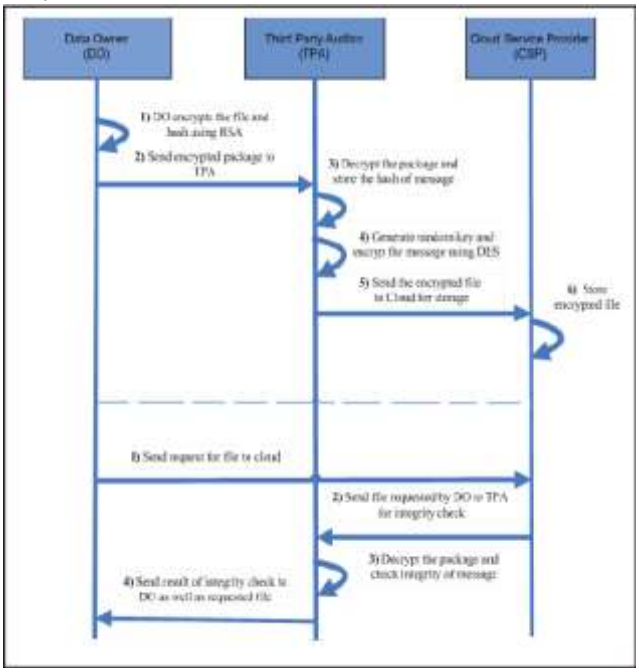


Fig. 2: Interaction among data owner, TPA and Cloud Service Provider

### A. Security Analysis

There are many providers, who offer security in their cloud storage systems, but the encryption and decryption processes are performed on the server. Moreover they do not support any kind of trustworthy data integrity mechanisms. Here this proposed mechanism provides integrity of the data, and verification of integrity as well as confidentiality and authentication to the data, so that the data is not disclosed to any unauthenticated person. Since the data is encrypted with the powerful encryption algorithms, it is practically impossible to decrypt it without the symmetric key. So, we can say that the attacks on the actual data without the presence of corresponding keys would not be successful. If it is assumed that the keys are kept secret and are not accessible in any way, then the confidentiality and integrity of data are guaranteed. There are following security analyses performed on the proposed scheme. Correctness: In this scheme only the data owner can decrypt the file received from the TPA because the file is encrypted by the public key of the data owner and to

decrypt it, the private key of Data owner is required, which is known to the owner only. So correctness of the proposed scheme is assured. Here when the owner wants the file stored on the cloud, the TPA request that file from Cloud Storage Provider and perform operations on it to verify the integrity of the data.

Authentication: The Data Owner signs the Hash of the message using his own key, no one else can signs the message, so TPA can easily finds whether the sender of the message is authenticated or not. If anyone else download the file from server by any mean and perform modification on the file after that he uploads the file on server, so for uploading he needs the private key of the owner for signing the file, and then the TPA can easily find out whether that person is an authenticated person or not. Privacy and Confidentiality: The file transferred between the TPA and Cloud Storage Provider is encrypted and encoded, which avoid Cloud Storage Provider knowing the content of the file and ensures privacy and confidentiality of the file. When the file is on the channel it is not the plain text, it is encrypted by some encryption mechanism, so any intruder could not get any information from the file in transmit and cannot use that file for own benefit. In this mechanism the encrypted file is stored on the cloud, so CSP could not get any information about the file stored on the cloud. Attack: In internet, users can be attacked from anywhere. As long as the internet is accessed to send a message, there is a risk that the message could be attacked in transmitted by any intruder? In this mechanism, the existence of the personal data in the cloud must be through internet. Owner stores their data in the cloud through internet and can access the data through internet. An attacker may be present during data transmission but in this proposed scheme, transmission is encrypted asymmetrically and one way hash function is also used. The transmission of data also includes the hash function that is encrypted and cannot be decrypted by any intruder by performing some action. If any person perform modification in file as well as in hash function it can be easily verified by the TPA and the result of verification is transferred to the owner.

### *B. Performance Analysis*

We measure the performance of our protocol in Windows 7 operating system. All experiments are conducted on the Intel(R) Core(TM) i3 processor with 2 GB RAM. Here various operations are performed by Mobile device, TPA and Cloud Storage Provider. A mobile user performs more encryption operations than decryption while TPA performs both encryption and decryption operations which reduces the work of mobile user. Here no encryption/decryption is performed by CSP. CSP also stores the encrypted file. The computation overhead of mobile terminal, TPA and Cloud Storage Provider is evaluated showing by Table III. Firstly the number of expensive cryptographic operations i.e. exponential operations, pairing operations and number of hash function are evaluated. These cryptographic operations are performed on mobile terminal and TPA. Here in table, a comparison of exponential operations, hash function and pairing operations performed on mobile device and TPA is shown during encryption and decryption process. Firstly, the number of exponential operations performed on mobile terminal during encryption process is analyzed and then on TPA. So, firstly one exponential operation is performed during encryption process of the file. After that one exponential operation is performed during re-encryption process of the encrypted file and one exponential operation is performed during encryption of hash. So, a total of three exponential operations are performed on the mobile device. Now, on TPA one exponential operation is performed during decryption of the received message and another exponential operation is performed during further encryption of the message before storing it on the mobile cloud. So a total of two exponential operations are performed on TPA during encryption. During decryption process only one exponential operation is performed on mobile terminal to decrypt the message. While on TPA a total of two exponential operations are performed. This shows that about 99% of the work during decryption is performed on TPA.

## V. EXPERIMENTAL RESULT

An application has been implemented using java language on the network (LAN) to achieve the functionalities of the client, TPA and cloud server. We have assumed that the cloud server, TPA and the user are in the same system domain and sharing the uniform system parameters. Through this application the messages can be transferred between these entities and the required result has been achieved.

## VI. CONCLUSION

When a resource constrained mobile device stores its data on the cloud, there is always a big concern of whether the cloud service provider stores the files correctly or not. Security is the main concern in mobile cloud computing. The proposed mechanism provides a security mechanism for securing the data in mobile cloud computing with the help of RSA algorithm and hash function. This research paper has proposed a mechanism to provide confidentiality and integrity to the data stored in mobile cloud. The proposed scheme uses RSA algorithm with other encryption decryption processes to secure the data in such a way that no leakage of data on cloud could be performed. In this scheme encryption is used to provide security to the data while in transmit. Because the encrypted file is stored on the cloud, so user can believe that his data is secure. In the scheme file, only in encrypted form is transferred over the channel, which reduces the problem of information disclosure. No, third person or intruder can get the file because that person do not knows the key of data owner. There is always a scope for improvement in every field of work, so here also. One of the assumption made in all the models of security are that the TPA is neutral. All the computations and verifications are offloaded to TPA so there is a need to do some work for making TPA more secure. Future work could be exploring the applications of other frameworks applied in secure storage services of mobile cloud environment. Some work can also be done to reduce the overhead of mobile terminal.

## REFERENCES

[1] G. Eason, B. Noble, and I. N. Sneddon, “On certain integrals of Lipschitz-Hankel type involving products of Bessel functions,” Phil. Trans. Roy. Soc. London, vol. A247, pp. 529–551, April 1955. *(references)*

[2] Hoang T. Dinh, Chonho lee, Dusit Nivato and Ping Wang, “A survey of mobile cloud computing: architecture, applications, and approaches,” Wireless Communication Mobile Computing, 2011.

[3] Rafik Jamil Shaikh, "Mobile cloud computing Application," International Conference on Technology and Business Management, March 28-30, 2011.

[4] Abdul Nasir Khan, M.L. Mat Kiah , Samee U. Khan, Sajjad A. Madani, "Towards secure mobile cloud computing: A survey," Future Generation Computer Systems, 2012.

[5] G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson and D. Song, ”Provable data possession at untrusted stores”, Proceedings of the 14th ACM conference on Computer and communications security, CCS’07 ,New York, USA, ACM, 2007, pp. 598-609.

[6] Q. Wang, C. Wang, K. Ren, W. Lou and J. Li , ”Enabling Public Auditability and Data Dynamics for Storage Security in Cloud Computing”, IEEE Transactions On Parallel And Distributed Systems, Vol. 22, No. 5, May 2011.

[7] Shashi Mehrotra Seth, Rajan Mishra, " Comparative Analysis Of Encryption Algorithms For Data Communication," IJCST Vol. 2, Issue 2, June 2011 .

[8] Abbas Amini, "Secure Storage in Cloud Computing," 2012.

[9] Behrouz A. Forouzan, "Cryptography and Network Security" special Indian edition, The McGraw-Hill Companies.

[10] Ayesha Malik, Muhammad Mohsin Nazir, "Security Framework for Cloud Computing Environment: A Review," Journal of Emerging Trends in Computing and Infonnation Sciences ,2012.

[11] W. Itani, A. Kayssi, and A. Chehab "Energy-Efficient Incremental Integrity for Securing Storage in Mobile Cloud Computing "in Proceedings of the First Annual International Conference on Energy Aware Computing, December 2010.

[12] P. Cox, "Mobile Cloud Computing: Devices, Trends, Issues, and the Enabling Technologies" in IBM developer Works, March 2011 .

[13] Xiaojun Yu, Qiaoyan Wen, "Design of Security Solution to Mobile Cloud Storage," Springer Volume 135, 2012, pp 255-263.

[14] W. Jia, H. Zhu, Z. Cao, L. Wei, X. Lin, "SDSM: a secure data service mechanism in mobile cloud computing," IEEE Conference on Computer.

[15] Communications Workshops, INFOCOM WKSHPS, Shanghai, China, Apr. 2011.

[16] W. Ren, L. Yu, R. Gao, F. Xiong, "Lightweight and compromise resilient storage outsourcing with distributed secure accessibility in mobile cloud computing," Journal of Tsinghua Science and Technology 16 (5) (2011) 520-528.

[17] Z. Hao, S. Zhong and N. Yu ,”A Privacy-Preserving Remote Data Integrity Checking Protocol with Data Dynamics and Public Verifiability”, IEEE Transactions On Knowledge And Data Engineering, Vol. 23, No. 9, September 2011.

[18] Jeff Sedayao, ”Enhancing cloud security using data anonymization”, Intel white paper on Cloud computing and information security, June 2012.

[19] S.c. Hsueh, lY. Lin, M.Y. Lin, "Secure cloud storage for conventional data archive of smart phones," 15th IEEE Int. Symposium on Consumer.

[20] Electronics, ISCE ' I I , Singapore, June 2011 .

[21] S.Poonkodi, VKavitha, KSuresh, "Providing a secure data forwarding in cloud storage system using threshold proxy re-encryption scheme," in International Journal of Emerging Technology and Advanced Engineering ISSN 2250-2459,Volume 3, Special Issue I, January 2013 .

[22] G D. Fabbrizio, T. Okken, and l G Wilpon, "A speech mashup framework for multimodal mobile services," in Proceedings of the 2009 international conference on Multimodal interfaces (ICMI-MLMI), pp. 71-78, November 2009.