

Two Way Encryption Process in Cloud by using Secure Hash Algorithm

Mr. Kotresh Naik¹

Department of Information Science and Engineering, Channabasaveshwara Institute of Technology, Gubbi, Karnataka, India.

Kotresh.cit@cittumkur.org

Mr. Anil Kumar²

Department of Information Science and Engineering, Channabasaveshwara Institute of Technology, Gubbi, Karnataka, India.

anilkumar9964@gmail.com

ABSTRACT

In this paper, a new scheme has been introduced, “Two Way Integrity Algorithm”. The paper presents a way to implement Third Party Auditor (TPA) which not only checks the reliability of Cloud Service Provider (CSP) but also checks the consistency and accountability of data. This paper addresses this challenging open issue of integrity, confidentiality and authentication of data. It provides an efficient data integrity mechanism between the client and the cloud service provider by applying RSA with digital signature on the message digest instead of on the whole data to make the computations faster. It then uses TPA for auditing the data stored in the cloud and finally provides an algorithm to check data integrity between the client and the TPA.

Index Terms: Two way integrity check algorithm, cloud service provider, Third party auditor, message digest.

I. INTRODUCTION:

Cloud computing is an emerging commercial infrastructure paradigm that promises to eliminate the need for maintaining expensive computing hardware. Through the use of virtualization and resource time-sharing, clouds address with a single set of physical resources a large user base with different needs. Thus, clouds promise to enable for their owners the benefits of an economy of scale and, at the same time, reduce the operating costs for many applications. For example, clouds may become for scientists an alternative to clusters, grids, and parallel production environments [1]. The ever cheaper and more powerful processors, together with the —software as a service (SaaS) computing architecture, are transforming data centres into pools of computing service on a huge scale. Meanwhile, the increasing network bandwidth and reliable yet flexible network connections make it even possible that clients can now subscribe high-quality services from data and software that reside solely on remote data centres. The capability provided to the consumer is to use the provider's applications running on a cloud infrastructure. The applications are accessible from various client devices through a thin client interface such as a web browser (e.g., web-based email). The consumer does not manage or control the underlying cloud infrastructure including network, servers, operating systems, storage, or even individual application capabilities, with the possible

exception of limited user-specific application configuration settings [10].

Several trends are opening up the era of cloud computing, which is based on internet. A lot of research introduces many solutions to decrease the threat of the data integrity and privacy. From the prospect of data safety, which has always been an important aspect of quality of services, cloud computing inevitably poses new challenging security threats for the number of logics. Initially, traditional cryptographic primitives for the purpose of data security protection cannot be directly adopted due to the users' loss control of data under cloud computing. Therefore, authentication of correct data storage in the cloud must be conducted without explicit knowledge of the whole data considering various kinds of data for each user stored in the cloud and the demand of long term continuous assurance of their data safety, the problem of verifying exactness of data storage in the cloud becomes even more challenging. Then, cloud computing is not just a third party data warehouse [5]. The data stocked in the cloud may be regularly updated by the users, including Insertion, modification, deletion, recording, appending, etc. To ensure depository correctness under progressive data update is hence of paramount importance. However, this feature also makes traditional completeness insurance techniques futile and entails new solutions. The deployment of cloud computing is powered by data

centres running in a simultaneous, cooperates and delivered manner.

II. proposed system

In our scheme, The client asks the CSP to provide service where CSP authenticates the client. RSA with Digital signature part will be done by the user to provide data authentication, data integrity and non-repudiation. This is done by first encrypting the user's data using symmetric encryption. The secret key involved is also encrypted using RSA algorithm (by receiver's public key). Then the message digest is created using MD5 algorithm and then the message is signed. After that the message and the signature is sent to the cloud service provider. Thereafter the CSP uses the receiver's private key to retrieve the digest. CSP uses the receiver's private key on the signature to retrieve the digest D' and then it applies the hash algorithm on the encrypted data to get the digest D. CSP now compares the two digests. If they are equal the message is accepted otherwise it informs the user that the data has been intruded. Digital signature will be used as a client's or data owner's identity and message digest helps in ensuring integrity of the data [1]. To enable cloud data storage security under the aforementioned model.

Our protocol design should achieve the following security and performance guarantee:

1. Public auditability: to allow TPA to verify the correctness of the cloud data on demand without retrieving a copy of the whole data or introducing additional on-line burden to the cloud users[2].

2. Storage correctness: to ensure that there exists no cheating cloud server that can pass the audit from TPA without indeed storing users' data intact.

3. Privacy-preserving: to ensure that there exists no way for TPA to derive users' data content from the information collected during the auditing process;

Fig.1. explains the system model it consists of User, TPA and cloud, we are providing integrity for the user from cloud as well as from TPA. Where user is sending the encrypted message to the cloud via TPA and cloud will receive the data and it checks the data is intruded or not while transmitting. After that TPA will verify the cloud by checking the data in cloud is modified or not then it informs the user that cloud is intrude. After verifying the cloud now cloud look after the TPA by checking the data in TPA if the data in TPA is modified means cloud will inform the user that TPA is intrude.

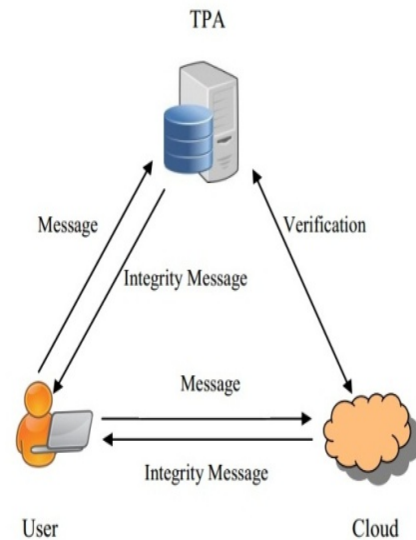


Fig. 1. Integrity mechanism architecture.

Our scheme consists of 4 parts:

1. RSA with Digital signature part will be done by the user.
2. The CS verifies over the user data in the cloud to check over the manipulation or intrusions in the cloud data.
3. The TPA verifies over the cloud server part to check if the cloud server was manipulating in the user data or not. Now, the each entity function in the proposed model is mentioned below:

1. User: User first uses RSA scheme to construct the public and the private keys then he/she will sign the data using the private key to form the digital signature to be uploaded to the cloud. After that the user sends the digital signature and the data to the cloud server and deletes its local copy.

- 2- CS: CS will compute a hash value from the original data to send it to the TPA, and then takes this hash value along with the data signed in the cloud for verification using the public key. At the end, the CS will inform the user if the data in the cloud intruded or not.

- 3- TPA: After the cloud server finishes its role, the TPA will be initiated to verify over the cloud server work by taking the hash value from the cloud server. TPA will take the data signed from the cloud and decrypt it with the public key. The decryption will result a hash value that will be compared along with the hash value that the cloud server compute it in his part. After finishing the verification, the TPA will inform the user if the CS was trusted or not.

III. Three way integrity algorithm

Our proposed algorithm consists of three algorithms:

A. Integrity check mechanism between client and CSP

The steps are:

- 1) The sender first encrypts the data using symmetric encryption using a shared key.
- 2) The shared key is also encrypted using RSA algorithm (Receiver's public key).
- 3) Then the message digest is created using md5 algorithm, $D=h(M')$.
- 4) Then the message is signed, $S=D^d \text{ mod } n$.
- 5) Then the encrypted message and signature is sent to the cloud service provider.
- 6) Then CSP uses the receiver's private key on the signature to retrieve the digest, $D'=S^e \text{ mod } n$.
- 7) Then it applies the hash algorithm on the encrypted data to get the digest D.
- 8) CSP now compares the 2 digests D and D'. If they are equal, the message is accepted, otherwise it informs the user that the data in the cloud is intruded.

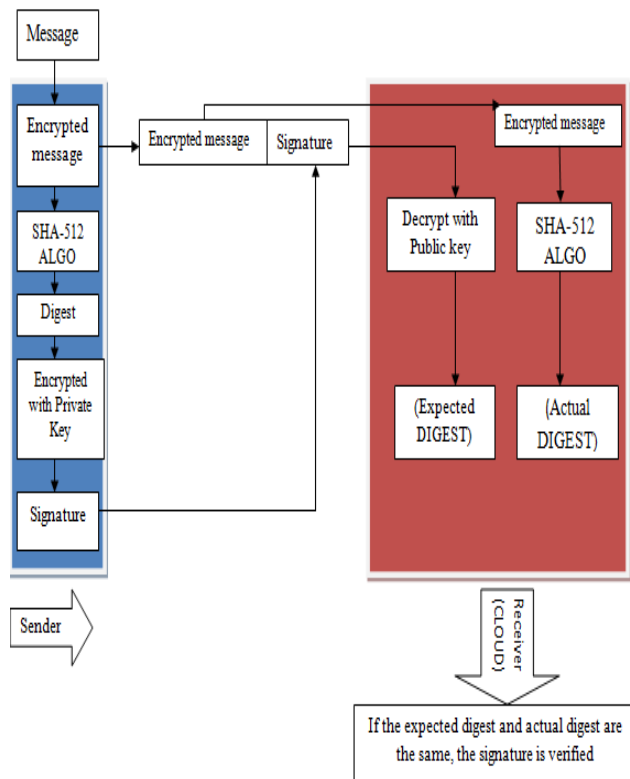


Figure 2: Integrity checks mechanism between client and CSP.

B. Integrity check mechanism between client and third party auditor.

The steps are:

- 1) After the CSP finishes its role, the TPA will be initiated to verify over the cloud server work by taking the hash value(digest) from the CSP (i.e D). TPA will take the data signed from the cloud and decrypt it with the public key and finds the Message digest.

- 2) The decryption will result a hash value that will be compared along with the hash value that the cloud server compute it in his part. After finishing the verification, the TPA will inform the user if the CSP was trusted or not.
- 3) $D(\text{computed by CSP}) = D'(\text{computed by TPA})$ then it means that the CSP is reliable and the data is secured

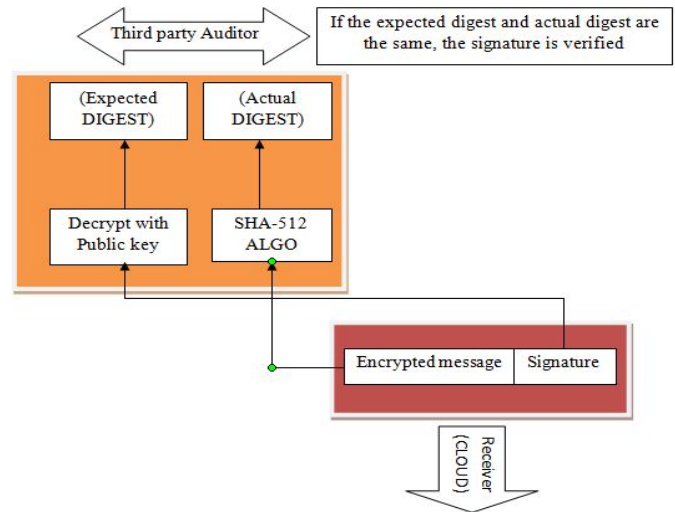


Figure 3: Integrity checks mechanism between client and third party auditor.

IV. Mathematical model

A public auditing scheme consists of four algorithms (KeyGen, SigGen, Genproof, VerifyProof). KeyGen is a one form of key generation algorithm that is run by the user to setup the scheme. SigGen is used by the user to generate analysis metadata, which may consist of MAC and Signatures. GenProof is run by the cloud server to generate a proof of data storage correctness. VerifyProof is run by the TPA and CSP to audit the proof from the cloud server and to audit the proof from the TPA respectively.

Our public auditing system can be constructed from the above auditing scheme in two phase, setup and audit:

A. Setup-phase:

The user initializes the public and secret parameters of the system by executing KeyGen, and preprocesses the data file F by using SigGen to generate the analysis metadata. The user then supply the data file F at the cloud server, and publish the analysis metadata to TPA for later audit. As a part of pre-processing, the user may alter the data file F by expanding it or including additional metadata to be stored at server.

1) KeyGen

RSA: Initially, we describe the parameters involved in a standard RSA signature scheme.

Each sender includes a public key $PK = (e, n)$ and private key $key = (d, n)$ where n is a K -bit modules generated as the product of two random $k/2$ -bit primes p, q and $n = p * q$ where, $p, q \in$ discrete prime numbers.

2) SigGen

All the message digests is formed using the SHA-512 algorithm. $D = H(M)$, where M is the user's message, $H()$ is the applied hash algorithm SHA-512 and D is the message digest involved. Then, digital signature is obtained by encrypting the message digest using the private key (d, n) .

INPUT: Sender has the private key (d, n) , receiver has the public key (e, n) , and message to be signed, M .

OUTPUT: S , signature of M

a) $D = h(M)$.

b) $S = D^d \text{ mod } n$.

c) Return(s).

B. Audit-phase

It consists of GenProof and VerifyProof:

1) GenProof:

The server uses GenProof to generate a response proof of data storage correctness.

INPUT: public key of sender (e, n) , message M , signature S .

OUTPUT: D, D' .

a) $D' = S^e \text{ mod } n$.

b) $D = h(M)$.

If $D = D'$ the received data is valid else it informs user that the data is modified [3].

2) VerifyProof:

With response from the server, the TPA and CSP runs VerifyProof to validate the response run by TPA to check whether the CSP is reliable or not and CSP to check whether the TPA is reliable or not

INPUT: signature S , public key of sender (e, n) .

OUTPUT: D, D'' .

a) $D'' = S^e \text{ mod } n$.

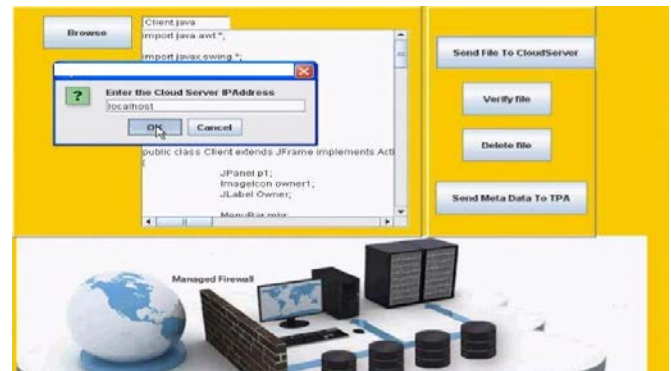
b) $D = h(M)$ (from the CSP).

If $D = D''$ the CSP is genuine else it is not genuine.

The decryption will result a hash value that will be compared along with the hash value that will be compared along with the hash value that the cloud server compute it in his part. After finishing the verification of hash value, the TPA will bring out the user if the CSP was trusted or not and CSP will inform the user if the TPA was trusted or not

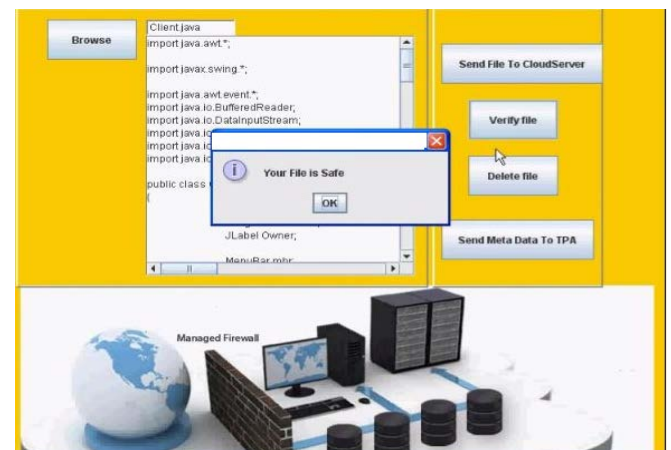
V. Results

1. Sending data to cloud from the user.



First user has log on to the system, by using the IP address of cloud user has to send the data to cloud.

2. After sending the data to the cloud.



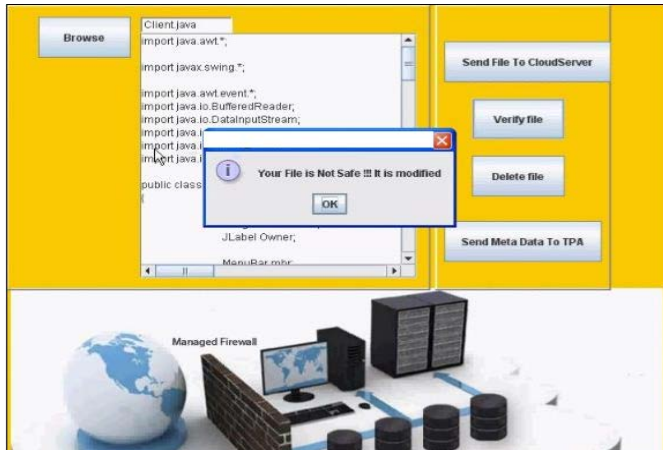
Now CSP will check the hash value (digest) of data i.e. if the expected digest and the actual digest are equal then it informs user that the data is not modified.

3. Data is intruded by the proxy server.



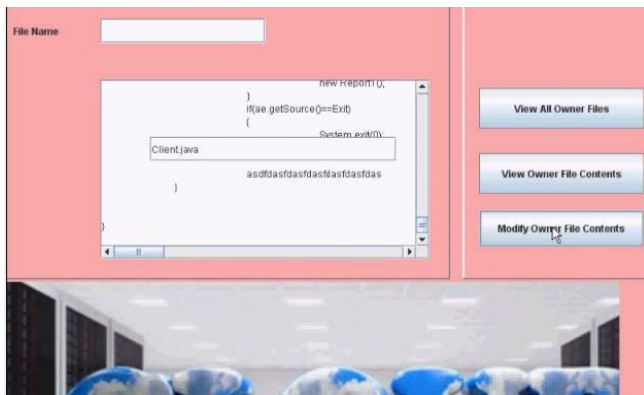
While sending the data to cloud if any proxy server modified the data then MAC address (digest) will change.

4. Modified data reached cloud.



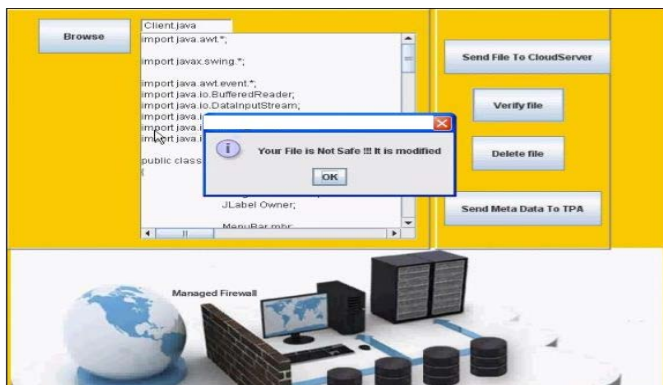
Now the cloud will verify the data. If it finds difference in the expected digest and actual digest then CSP will inform the user that data is modified.

5. Data is modified by the cloud owner itself.



If CSP modified the data in cloud, then cloud is to be considering as intruder.

6. TPA will be initiated to verify over the cloud.



If the data is modified by the cloud, then TPA will verify the hash value, After verification it inform the user that CSP was trusted or not.

Finally the cloud will be initiated to verify over the TPA, by taking the hash value from TPA and CSP. After finishing the verification, the CSP will inform the user if the TPA was trusted or not.

VI. Conclusion

In this paper, we design a system showing cloud architecture, user and TPA that provide integrity proofs where TPA helps in interacting with the cloud service provider and cloud service provider interacting with the TPA on behalf of the client in order to check the data security at the server is presented.

Then, an efficient scheme for checking the data integrity between the client, TPA and server is introduced. Also an algorithm is proposed to check for the reliability of the CSP and TPA i.e. a mechanism checking data integrity between the client, TPA and CSP.

We believe that security in cloud computing is very much needed as data in the cloud storage are not secure and require lots of attention of user. The algorithm that includes data integrity check mechanism between the third party auditor and

CSP is helping in detecting the leakage of data or intrusion done by the cloud service provided itself.

VII. References

1. Cong Wang, Qian Wang, Kui Ren, "Ensuring data storage security in cloud computing", IEEE 2010
2. O Rajitha, Murali Krishna, "Secure dynamic data support and trusted third party auditor in cloud computing", International Journal of science & Engineering Research, Volume 4, Issue 10, October-2013.
3. Garima, "Ensuring data storage security in cloud using two way integrity check algorithm", International Journal of Advanced Research in Computer Science and Software Engineering, Volume 3, Issue 11, November-2013.
4. Xuefeng Liu, Yuqing Zhang, Boyang Wang and Jingbo Yan, "Mona: secure multi-owner data sharing for dynamic groups in the cloud", IEEE transactions on parallel and distributed system, Volume 24, NO. 6, June- 2013.
5. Kapila Sharma, Kavita Kanwar, Chanderjeet Yadav, "Data Storage Security in Cloud Computing", International Journal of Computer Science and Management Research, Volume 2 Issue 1 January 2013
6. M.Sudha, M.Monica Enhanced Security Framework to Ensure Data Security in Cloud Computing Using Cryptography, presented at Advances in

- Computer Science and its Applications, Vol. 1, No. 1, March 2012.
7. Cong Wang and Kui Ren, Wenjing Lou, Jin Li, -Toward Publicly Auditable Secure Cloud Data Storage Services, presented at IEEE Network ,July/August 2010
 8. M. Sudha Dr.Bandaru Rama Krishna Rao,M. Monica,|| A Comprehensive Approach to Ensure Secure Data Communication in Cloud Environment, International Journal of Computer Applications (0975 – 8887) Volume 12– No.8, December 2010.
 9. Akhil Behl, Emerging Security Challenges in Cloud Computing,lpresented at World Congress on Information and Communication Technologies,2011.
 10. Wang Junxiang, Liu Shengli, Dynamic Provable Data Possession with Batch-Update Verifiability.
 11. Sandeep K. Sood –A combined approach to ensure data security in cloud computing lin Journal of Network and Computer Applications , ELSEVIER, 3 July 2012.
 12. Kan Yang · Xiaohua Jia –Data storage auditing service in cloud computing: challenges, methods and opportunities in Springer Science+Business Media, LLC 2011.
 13. S. Subashini , V.Kavitha –A survey on security issues in service delivery models of cloud computing , in Journal of Network and Computer Applications , Elsevier, 11 July 2010.