ijecs open access www.ijecs.in
International Journal Of Engineering And Computer Science ISSN:2319-7242
Volume 3 Issue 5 may, 2014 Page No. 6033-6037

Integrity Check Mechanism in Cloud Using SHA-512 Algorithm

Mrs.Shantala C P[1], Mr.Anil Kumar[2]
Department of Computer Science and Engineering
Channabasaveshwara Institute of Technology
Gubbi, Karnataka, India.
shantala.cp@cittumkur.org.

Department of Computer Science and Engineering,
Channabasaveshwara Institute of Technology
Gubbi, Karnataka, India.
anilkumar9964@gmail.com.

Abstract— **Cloud computing is an alternative to traditional information technology due to its services. In this paper, a new theory has been introduced three way integrity algorithm. Here we are checking the accuracy of cloud service provider (CSP) and third party auditor (TPA). It gives an efficient data integrity mechanism between the client and the cloud by using RSA with digital signature on the message digest instead of on the whole data to make computations faster. Our public auditing system can be constructed from the above auditing scheme in two phases, setup and audit phase. The problem of verifying correctness of data storage in the cloud becomes even more challenging. However, this dynamic feature also makes traditional integrity insurance techniques futile and entails new solutions. We propose an extensible and emphatic distributed scheme with explicit dynamic data support to ensure the correctness of users' data in cloud.**

Index Terms— **Three way integrity check algorithm, cloud service provider, Third party auditor, message digest**.

I. INTRODUCTION

Over 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. Individual user's data is redundantly stored in multiple physical locations to further decreases the data integrity threats. Hence, distributed protocols for storage correctness assurance will be of most importance in achieving a robust and secure cloud data storage system in the real world [1]

Some techniques can be useful to ensure the storage correctness without having user's involvement to check local data they are all focusing on single server scenario [2]. In this paper, we propose an effective and flexible distributed scheme with explicit dynamic data support to ensure the correctness of users' data in the cloud.

II. PROPOSED SYSTEM

In our scheme, the client asks the CSP and TPA to provide services where CSP and TPA authenticate the client. RSA with digital signature part will be done by the user to provide data correctness, non-repudiation and data authentication. This is done by first encrypting the user's data using symmetric encryption. The secrete key involved is also encrypted using RSA algorithm (by receiver's public key). Then the message digests created using SHA-512 algorithm and then the message is signed. After that the signed message and the signature is sent to the cloud service provider. There after 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 (SHA-512) algorithm on the encrypted data to get the digest D. CSP now compares the two digits. If they are equal the message is accepted otherwise it informs the user that the data has been intruded.

Fig.1. explains the system model it consist 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.

Digital signature will be used as a client's or data owner's identity and message digest helps in ensuring integrity of the data [3]. To enable cloud data storage security under the preceding model, our protocol architecture should achieve the following security and performance guarantee:

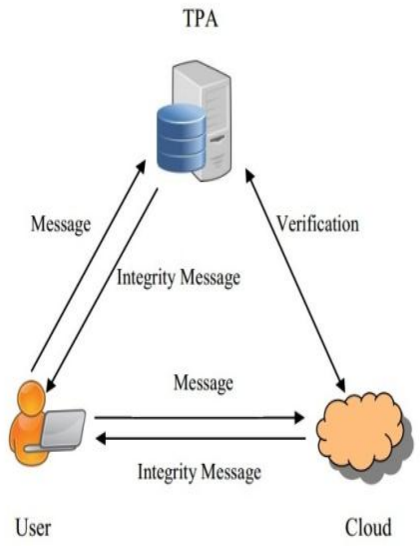


Fig. 1. Integrity mechanism architecture.

Our scheme consists of 4 parts:

1) Applying RSA with digital signature will be done by the user.
2) The CS verify over the user data in the cloud to check over the manipulating in the user data or not.
3) The TPA verify over the cloud server part to check if the cloud server was manipulating in the user data or not.
4) The CSP verify over the TPA to check if the TPA was manipulating in the user data or not.

Now the each concept function in the proposed model is mentioned below:

User: User first uses RSA scheme to construct the public and 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 posts the digital signature and the data to the cloud server and deletes its local copy.

CS: CS will compute a hash value from the original data, this hash value along with the data signed in the cloud for authentication using the public key. At the end, the CS will notify the user if the data in the cloud intrude or not and one more role of CS is to verify over the TPA by taking the hash value from the TPA. CS will take the data signed from the TPA and decrypt it with the public key. The decryption will result a hash value that the TPA compute it in his part. After completing the Verification. The CS will inform the user if the TPA was trust or not.

TPA: After the cloud server finishes its role, the TPA will be brought into verify over the cloud server work by taking the hash value from the cloud server. TPA takes the data signed from the cloud and decrypts 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 correctness, the TPA will disclose 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 encryption using shared key.
2) Then the message digest is created using SHA-1 algorithm, D=h (M').
3) Then the message is signed=D^d mod n.
4) Then the encrypted message and signature is sent to the cloud service provider.
5) Then CSP uses the receiver's private key on the signature to retrieve the digest, D'=S^e mod n.
6) It applies the hash algorithm on the encrypted data to get the digest D.
7) CSP now compares the two digests D and D'. If they are not equal, it posts the user that the data in the cloud is modified. Otherwise the message is accepted.

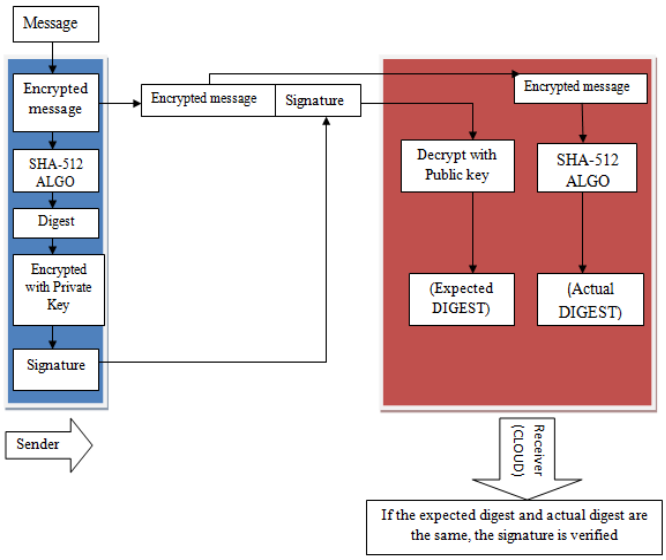


Fig. 2. Integrity checks mechanism between client and CSP.

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

The steps are:

1) After 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 Messages digest.
2) The decryption will result a digest that will be compared along with the digest that the cloud server compute in his part.
3) After finishing the verification, the TPA will let now the user if the CSP was trusted or not. If D (computed by CSP) = D'' (computed by TPA) then it means that the CSP is reliable and data is secured.

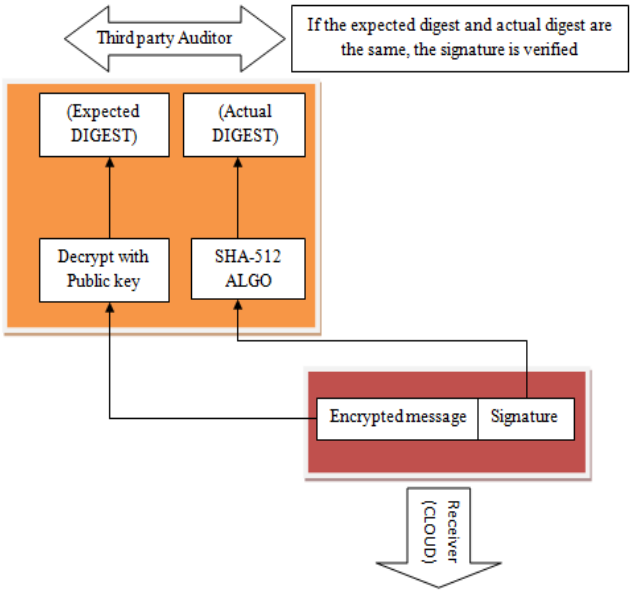


Fig. 3. Integrity checks mechanism between client and third party auditor.

C. *Integrity check mechanism between third party auditor and CSP..*

1) After verifying CSP, now the next task is to verify over the TPA by CSP by taking the hash value (digest) from TPA (i.e. D). CSP will take the data signed from the TPA and decrypt it with public key and find the message digest.
2) The decryption will results a hash value that will be compared along with the hash value that the TPA compute it in his part.
3) After finishing the verification, the CSP will inform the user if the TPA was trusted or not. If D (computed by CSP) =D''' (computed by CSP) then it means that the CSP is reliable and data is secured.

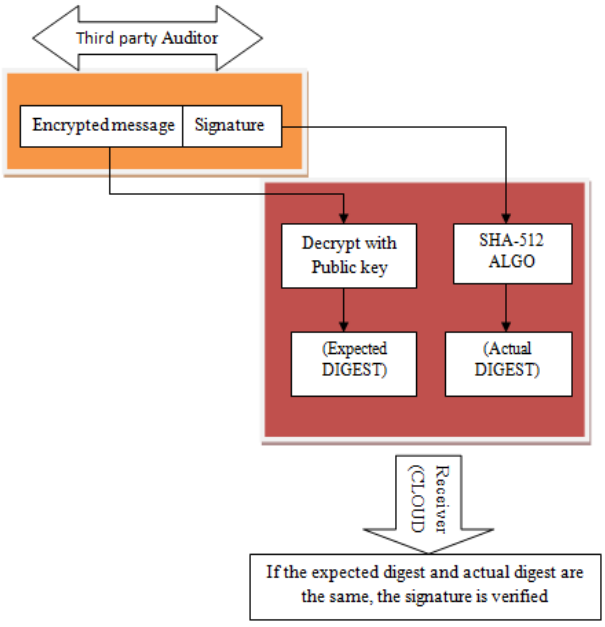


Fig. 4. Integrity check mechanism between third party auditor and CSP.

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=(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 € 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 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 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 mod n.
b) D=h (M) (from the CSP).
If D=D'' the CSP is genuine else it is not genuine.
c) D'''=S^e mod n.
If D=D''' the TPA 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

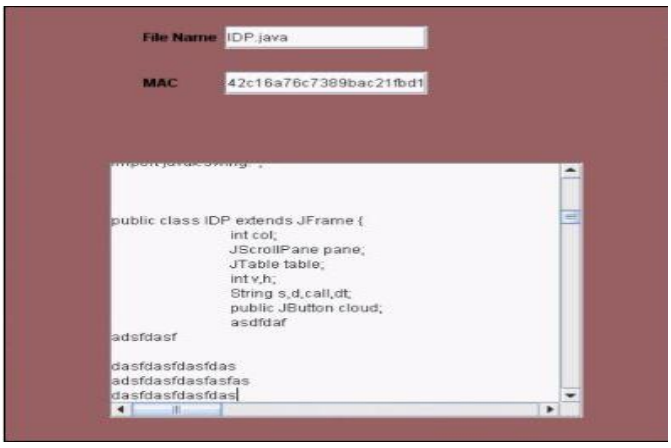
1. Sending data to cloud from the user.

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.

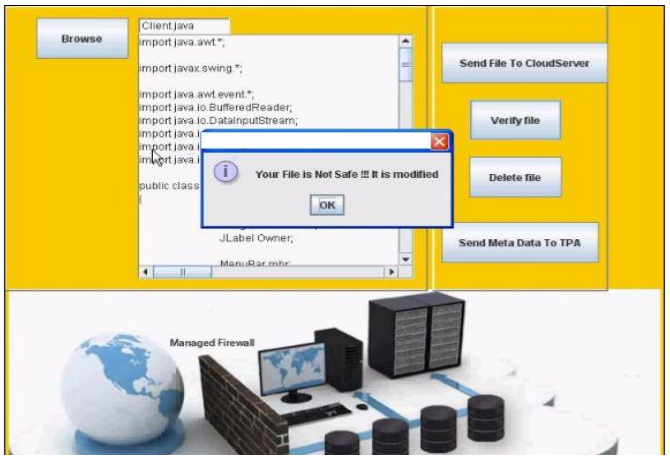
Now CSP will check the hash value (digest) of data i.e. if the expected digest and the actual digest are equal then it informs user that the data is not modified.

3. Data is intruded by the proxy server.



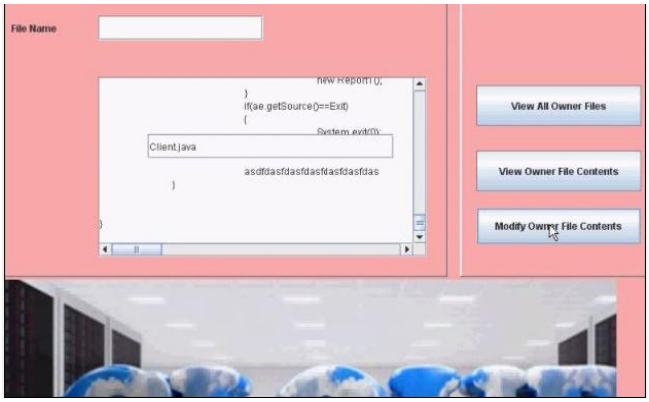
While sending the data to cloud if any proxy server modified the data then MAC address (digest) will change.

4. Modified data reached cloud.



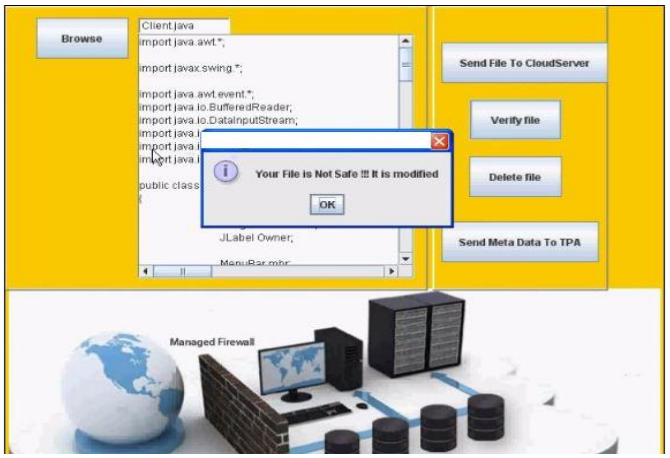
Now the cloud will verify the data. If it finds difference in the expected digest and actual digest then CSP will inform the user that data is modified.

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.

7. 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.

ACKNOWLEDGMENT

I express my sincere gratitude to Dr. D. S.Suresh Kumar, Director, CIT, I also thanks to HOD Prof. C. P. Shantala for continuous guidance without which I would not come up with this paper, I also thanks to all teaching, non teaching staff of CIT College.

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 Rearch, 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