Global Journal of Computer Science and Technology
Cloud and Distributed
Volume 13 Issue 3 Version 1.0 Year 2013
Type: Double Blind Peer Reviewed International Research Journal
Publisher: Global Journals Inc. (USA)
Online ISSN: 0975-4172 & Print ISSN: 0975-4350

Monitoring Data Integrity while using TPA in Cloud Environment

By Jaspreet Kaur & Jasmeet Singh
RIMT-IET, Mandi Gobindghar, India

Abstract - Cloud Computing is the arising technology that delivers software, platform and infrastructure as a service over a network. Cloud minimizes the burden of users by allowing them to remotely store their data and eliminates the need of local storage and maintenance. Even though benefits are higher but while storing data in cloud, correctness of data and security are major concerns as there are many chances for CSP to behave unfaithfully towards users regarding the status of their outsourced data. In order to overcome the threat of integrity, user can entrust third party auditor to assess the risk of outsourced data when needed .For this, in our proposed scheme we are using SHA-2 which is cryptographic hash function to verify integrity of data along with XOR mechanism, Station-to-Station key protocol for key generation and mutual authentication with TPA.

Keywords : *audit, cloud, integrity, station to station protocol, SHA-2, third party auditor, XOR.*

GJCST-B Classification : *C.2.4, H.2.7*

Strictly as per the compliance and regulations of:

Monitoring Data Integrity while using TPA in Cloud Environment

Jaspreet Kaur [α] & Jasmeet Singh [σ]

Abstract - Cloud Computing is the arising technology that delivers software, platform and infrastructure as a service over a network. Cloud minimizes the burden of users by allowing them to remotely store their data and eliminates the need of local storage and maintenance. Even though benefits are higher but while storing data in cloud, correctness of data and security are major concerns as there are many chances for CSP to behave unfaithfully towards users regarding the status of their outsourced data. In order to overcome the threat of integrity, user can entrust third party auditor to assess the risk of outsourced data when needed .For this, in our proposed scheme we are using SHA-2 which is cryptographic hash function to verify integrity of data along with XOR mechanism, Station-to-Station key protocol for key generation and mutual authentication with TPA.

Keywords : audit, cloud, integrity, station to station protocol, SHA-2, third party auditor, XOR.

I. Introduction

Cloud computing is a forthcoming revolution in IT industry due to its advantages-on demand service, location independent resource pooling, rapid resource elasticity and usage based pricing. Cloud Computing is collection of scalable and virtualized resources and provide service based on pay-as-you-use strategy.

Cloud services are broadly divided into three categories as shown in figure 1:

1. SAAS(Software as a service) – SaaS is a model of software deployment that enables software to be delivered from host source over the network as opposed to installations or implementations.
2. PAAS (Platform as a service) – This is a model in which operating system and middleware services are provided to the consumer by the cloud.
3. IAAS (infrastructure as a service) – This is a model in which service provider delivers the entire infrastructure as a service including storage, hardware, servers and networking components.[7]

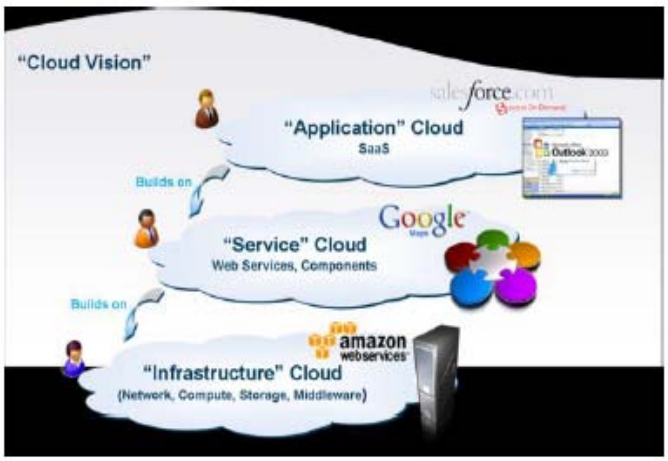

Figure 1 : Cloud Service Models

From user's perspective, including both individuals and enterprises, outsourcing data in the cloud in a flexible on-demand manner as shown in figure 2 brings appealing benefits-

- Relief of the burden of the storage management.
- Universal data access with independent geographical locations.
- Avoidance of capital expenditure on hardware, software and personnel maintenance.

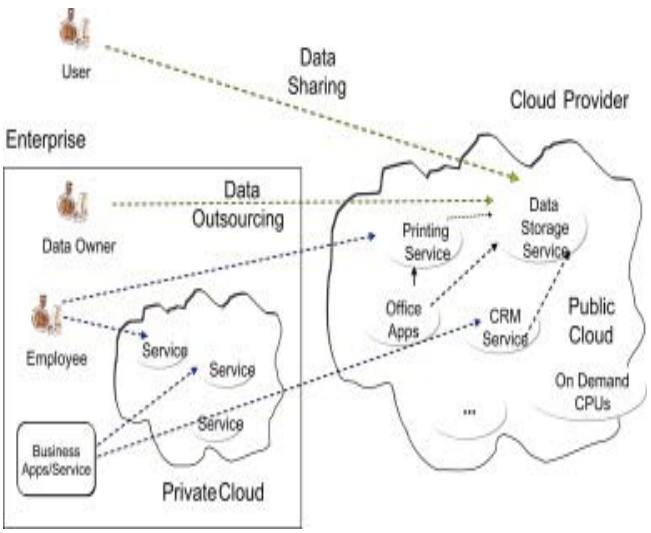

Figure 2 : Data Outsourcing in Cloud

However, outsourcing data brings new challenging security issues.

Author α : M.TECH (Computer Science and Engineering) RIMT-IET, Mandi Gobindgarh, Punjab, India. E-mail : kkhushi12@yahoo.co.in
Author σ : Asst. Professor (CSE Department), RIMT-IET, Mandi Gobindgarh, Punjab, India. E-mail : jasmeetgurm@gmail.com

- Data Integrity
- Unfaithful cloud service provider

The first issue is data integrity. In computer security, data integrity can be defined as "the state that exists when computerized data is the same as that in the source document and has not been exposed to accidental or malicious alterations or destruction". Integrity of data stored at the entrusted cloud server is not guaranteed.

The second issue is unfaithful cloud server providers (CSP). There are many reasons why CSPs are not always trustworthy like, for saving money and storage space, CSPs may discard the data that has not been accessed for long time (which belongs to ordinary client) or sometimes even hide data losses or corruptions to maintain a reputation.[1]

To ensure integrity of outsourced data, data owner delegate the auditing task to trusted third party auditor. The persona of TPA is listed as follows-

- Reduces the owner burden in managing the data.
- Ensure the client that the data stored in the cloud is intact and data integrity is maintained.
- Aid in achieving high economies of scale through customer satisfaction.[5]

The TPA is an independent authority that has expertise and capabilities to monitor the integrity of cloud data outsourced by the following two fundamental requirements have to be met:

1. TPA should be able to audit the cloud data storage efficiently without asking for the local copy of data thereby reducing the on-line burden of cloud users.
2. The third party auditor informs user about data corruption or loss, if any. To securely introduce an effective third party auditor (TPA), the auditing process should not affect user data privacy.[1]

Figure 3 represents the cloud data storage service which consists of three different entities: Users, Cloud Storage Server, Third Party Auditor (TPA).

1. *User :* Users are the data owners who have the large amount of data to be stored in the cloud and relies on cloud storage server for data computation and maintenance, can be either individual consumers or organizations.
2. *Cloud Storage Server :* A cloud storage server (CSS) is an entity that is managed by cloud service provider (CSP). It provides space to the user for data storage and computation.
3. *Third Party Auditor (TPA) :* An TPA is an entity who has expertise and capabilities that user don't have, is trusted to access or expose the risk of cloud storage services on behalf of client upon request.[8]

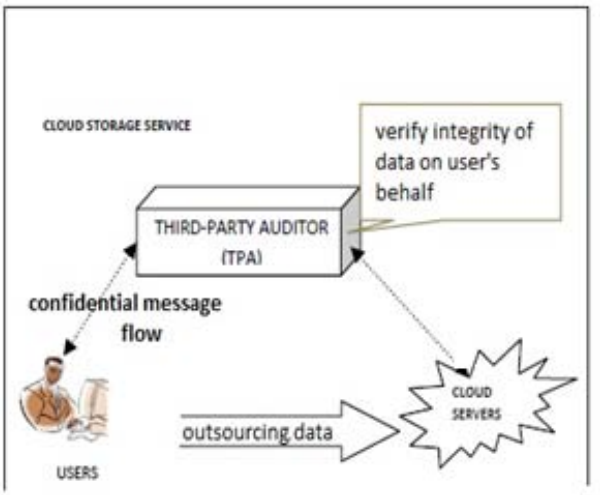

Figure 3 : Architecture of cloud data storage service

II. Problem Formulation

When we talk about Cloud Security, maintaining data integrity is one of the most important and difficult task. In this case, user cannot trust cloud service provider to handle the data by himself as he himself can modify the original data and integrity may be lost. If any intruder attacks and steals the data and modifies it then in some cases the modification is not even noticed by the cloud server. So in this case, user can rely on a trusted (authenticated) third party auditor to check for the integrity of his data. For this, strong cryptographic hash function is required by the auditor to check for integrity of cloud data.

III. Proposed Work Plan

For ensuring the integrity of the data we will be using combination of three approaches-

- Station-to-Station protocol (based on Diffie-Hellman key exchange algorithm) generates mutual key which is known to both user and auditor. It also provides entity authentication to both.
- Exclusive-OR (XOR) to perform a xor operation between the message and the key generated using Station-to-Station protocol.
- Secure Hashing Algorithm (SHA-2) to generate a digest by passing the original message to the hash function. This is done by both the user and the auditor and the value obtained from the hash function by both of them is compared and hence the data integrity is verified.

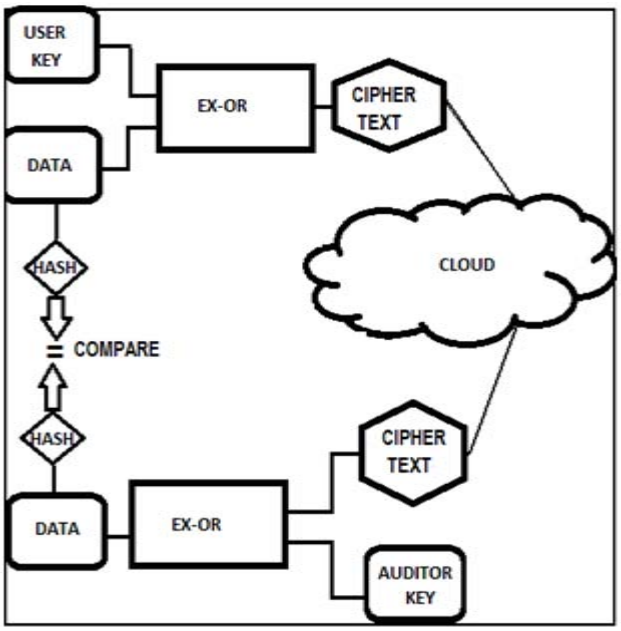

Figure 4 : Methodology

According to figure 4, the steps are as follows:

1. A secret key generated using STS protocol(that is known to both user and auditor).Also, mutual authentication is done using this protocol.
2. XOR operation is done between the data and the key generated to create cipher text which is stored in cloud.
3. Separately the data is passed in a hash function (using SHA-2) and the hash value is obtained by the user.
4. Auditor gets the cipher text from the cloud and performs an XOR operation with the secret key generated by the station to station protocol and gets a plain text.
5. Auditor passes this plain text to the same hash function (using SHA2) and obtains a hash value.
6. He then compares this hash value with the hash value received from the user .If both the values are identical then the data integrity is maintained else data is tampered.

a) Algorithms

i. *STS Protocol*

Key exchange should be linked to authentication so that a party has assurances that an exchanged key (which might be used to facilitate privacy or integrity and *thus keep authenticity alive*) is in fact shared with the authenticated party, and not an impostor. *Station-to-Station* (STS) *Protocol* is efficient authenticated key exchange protocol. The STS protocol consists of Diffie-Hellman key establishment followed by an exchange of authentication signatures.

Set Up-

- In the basic version of the protocol, we assume that the parameters used for the key establishment (i.e., the specification of a particular cyclic group and the corresponding primitive element α) are fixed and known to all users.
- Also, the public portion of this key pair may be shared prior to session establishment.

Steps-

1. The protocol begins with one party, Alice, creating a random number x and sending the exponential α^x to the other party, Bob.
2. Bob creates a random number y and uses Alice's exponential to compute the exchanged key $K = (\alpha^x)^y$. Bob responds with the exponential α^y and a token consisting of his signature on the exponentials, encrypted with K using a suitable symmetric encryption algorithm E (i.e., E_K (s_B { α^y, α^x })).
3. Alice computes K, decrypts the token using K, and verifies Bob's signature using Bob's public key. Alice sends to Bob her corresponding encrypted signature on the exponentials, $E_K(s_A\{\alpha^x, \alpha^y\})$.
4. Finally, Bob similarly verifies Alice's encrypted signature using K and Alice's public key.

Alice and Bob are now mutually authenticated and have a shared secret. This secret, K, can then be used to encrypt further communication.[6] The basic form of the protocol is formalized in the following three steps:

1. Alice → Bob: α^x
2. Alice ← Bob: α^y, $E_K(s_B\{\alpha^y, \alpha^x\})$
3. Alice → Bob: $E_K(s_A\{\alpha^x, \alpha^y\})$

a. *Exclusive or Operation*

XOR is an operation in which same bits produce a resultant bit as 0 whereas different bits produce a resultant 1. In this, a XOR operation is done at original text along with secret value which gives cipher text. The original text can be obtained by performing an XOR operation between the secret key and resultant cipher text.[4]

b) SHA-2

This family of hashing algorithms known as SHA-2, use larger digest messages than SHA-1, making them more resistant to possible attacks and allowing them to be used with larger blocks of data, up to 2^{128}bits, e.g. in the case of SHA512. The SHA-2 hashing algorithms are same for SHA256, SHA224, SHA384, and SHA512 hashing functions, differing only in the size of the operands, the initialization vectors, and the size of the final digest message.

The figure 5 describes the SHA-2 algorithm applied to the SHA256 hash function.

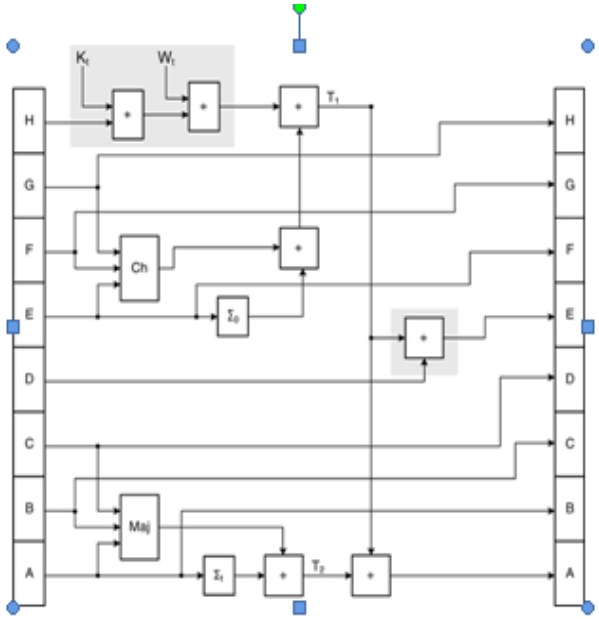

Figure 5 : SHA-2 Round Calculation

The SHA256 Hash function produces afinal digest message of 256 bits, that is dependent of the input message, composed by multiple blocks of 512 bits each. [3]

i. *Message Padding and Parsing*

The binary message to be processed is appended with a '1' and padded with zeros until its length ≡ 448 mod 512. The original message length is then appended as a 64-bit binary number. The resultant padded message is parsed into *N* 512-bit blocks, denoted $M^{(1)}$, $M^{(2)}$, ……. $M^{(N)}$. These $M^{(i)}$ message blocks are passed individually to the message expander.

ii. *Message Expander*

The message expander (also called the message scheduler) takes each $M^{(i)}$ and expands it into 64 32-bit W_t blocks.[2]

iii. *Message Compression*

The W_t words from the message expansion stage are then passed to the SHA compression function, or the 'SHA core'. The input block is expanded and fed to the 64 cycles of the SHA256 function in words of 32 bits each .In each round of the SHA-2 algorithm the introduced data is mixed with the current state. This data scrambling is performed by additions and logical operations, such as bitwise logical operations and bitwise rotations as shown in Figure 5.

The 32-bit values of the A to H variables are updated in each round and the new values are used in the following round. The initial values of these variables is given by the 256-bit constant value, this value is only set for the first data block. The consecutive data blocks use the intermediate hash value, computed for the previous data block. Each 512 data block is processed for 64 rounds, after which the values of the variables A to H are added to the previous digest message, in order to obtain partial digest message. To better illustrate this algorithm a pseudo code representation is depicted in table 1.

```
for each data block  i do
W  = expand(data block)
A = DM0; B = DM1; C = DM2; D = DM3
E = DM4; F = DM5; G = DM6; H = DM7
for t= 0, t≤ 63 {79}, t=t+1 do
T1= H +    1(E) + Ch(E, F, G) + Kt+ Wt
T2=    0 (A) + Maj(A, B, C)
H = G ; G = F  ; F = E  ;
E = D + T1
D = C  ; C = B  ; B = A
A = T1+ T2
end for
DM0= A + DM0; DM1= B + DM1
DM2= C + DM2; DM3= D + DM3
DM4= E + DM4; DM5= C + DM5
DM6= D + DM6; DM7= E + DM7
end for
```

Table 1 : Pseudocode for SHA-2

The final Digest Message (DM) for a given data stream, is given by the result of the last data block.[3]

IV. Conclusion

Cloud Computing is an emerging commercial paradigm that promises to eliminate the need for maintaining expensive hardware. As market grows threat on data also grows. In this paper to ensure that our data are intact we addressed the construction of an efficient audit service where user can delegate the integrity checking task to third party auditor and be worry-free to use cloud storage services. We believe that advantages of proposed scheme will shed light on economies of scale of cloud computing.

References Références Referencias

1. Jaspreet Kaur, Jasmeet Singh, "Survey on Efficient Audit Service to ensure Data Integrity in Cloud Environment", Global Journal of Computer Science and Technology (GJCST), Vol. 13, Issue 4, 2013.
2. Robert P. McEvoy, Francis M. Crowe, Colin C. Murphy and William P. Marnane, "Optimisation of the SHA-2 Family of Hash Functions on FPGAs", IEEE Emerging VLSI Technologies and Architectures, Vol. 00, 2-3 March 2006.
3. Ricardo Chaves, Leonel Sousa, "Improving SHA-2 Hardware Implementation", International Association for Cryptographic Research, 2008.
4. K. Govinda, E. Sathiyamoorthy, "Data Auditing in Cloud Environment using Message Authentication Code", International Conference on Emerging Trends on Advanced Engineering Research (ICETT), 2012.

5. Miss. M. Sowparnika, Prof. R. Dheenadayalu, "Improving Data Integrity on Cloud Storage Services", International Journal of Engineering Science Invention (IJESI), Vol. 2, Issue 2, February 2013.
6. Whitfield Diffie, Michael J. Wiener, "Authentication and Authenticated Key Exchanges", Appeared in Designs, Codes and Cryptography, 2, 107-125 (1992), Kluwer Academic Publishers.
7. Shobha Rajak, Ashok Verma, "Secure Data Storage in the Cloud using Digital Signature Mechanism", International Journal of Advanced Research in Computer Engineering and Technology, Vol. 1, Issue 4, June 2012.
8. B. Dhiyanesh, A. Thiyagarajan, "A Novel Third Party Auditability and Dynamic Based Security in Cloud Compuitng", International Journal of Advanced Research in Technology(IJART), Vol. 1, Issue 1, 2011.
9. Reenu Sara Georeg, Sabitha S, "Survey on Data Integrity in Cloud Computing", International Journal of Advanced Research in Computer Engineering & Technology (IJARCET) Vol. 2, Issue 1, January 2013.
10. http://en.wikipedia.org/wiki/Station-to-Station_protocol