SECURING THE CLOUD COMPUTING SYSTEMS WITH MATRIX-VECTOR AND MULTI-KEY USING LINEAR EQUATIONS

Shankar M[1], Kannan M[2]

[1]*M.E, Computer Science and Engineering, Mahendra Engineering College, Thiruchengode, Tamil Nadu, India*
[2]*Professor/CSE, Mahendra Engineering College, Thiruchengode, Tamil Nadu, India*

Abstract

Cloud computing systems security is to improve when end-user communicates with the cloud server. In order to make a connection between end-user and the Cloud Server, first end-user or Cloud Server make sure that they are communicating with right counterpart. Multi-Key encryption concept is used to encrypt the request message or data which is sent from end-user or also from Cloud Server. Request message sent by end-user to cloud server should matches with the response received from by end-user from cloud server. Multi-key concept takes more number of iteration when compared to single-key concept when trying to validate the requested message. Multi-Key concept is more secured than single-key.

--***--

1. INTRODUCTION

Cloud computing is the provision of dynamically scalable and often virtualized resources as a services over the internet Users need not have knowledge of, expertise in, or control over the technology infrastructure in the "cloud" that supports them. Cloud computing represents a major change in how we store information and run applications. Instead of hosting apps and data on an individual desktop computer, everything is hosted in the "cloud machine", an assemblage of computers and servers accessed via the Internet.

End-user wants to store their confidential data into the cloud machine, in order to store and access data in the cloud server, first step is to make a secure connection between end-user and cloud machine. Then user can access those data from internet and also from different location across globe. If the user wants to access their data, user has to make a secure connection to the cloud server and then user will send a request message to the cloud server and cloud server will validate the request messages which are received from the end-user. Both the end-user and the cloud server will do the validation and then secure connection will make only when the validation is successful on both sides.

2. ARCHITECTURE

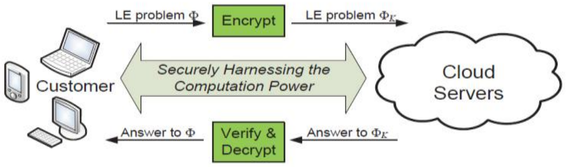


Fig 1 Architecture

Customer will send an encrypted message to cloud server and cloud server will process the request message which is sent by customer and sent back the processed message which is encrypted by cloud server and customer will verify the processed message received from the cloud server and then secure connection will establish between customer and the cloud server.

3. KEY GENERATION

End-user will generate a set of keys and send to cloud server or cloud server will generate a set of keys and send to end-user. Once the key is generated and the generated keys should be shared between the end-user and the cloud server. Then end-user has to send a request message to make a connection with cloud server and vice-versa. End-user will encrypt the request message Encrypt(m1) with the Key1 and send to cloud server through communication channel. Cloud server will decrypt the message sent by end-user [Decrypt(Encrypt(m1))] and also cloud server will encrypt the same encrypted message (before decryption in cloud server side) which is received from the end-user with the Key1 which is available in cloud server and send back the encrypted message to end-user for validation. End-user will decrypt the message which is received from cloud server and compared with the encrypted message which is sent by end-user and should match otherwise someone has interrupted the communication channel or someone has modified the data which is sent through communication channel.

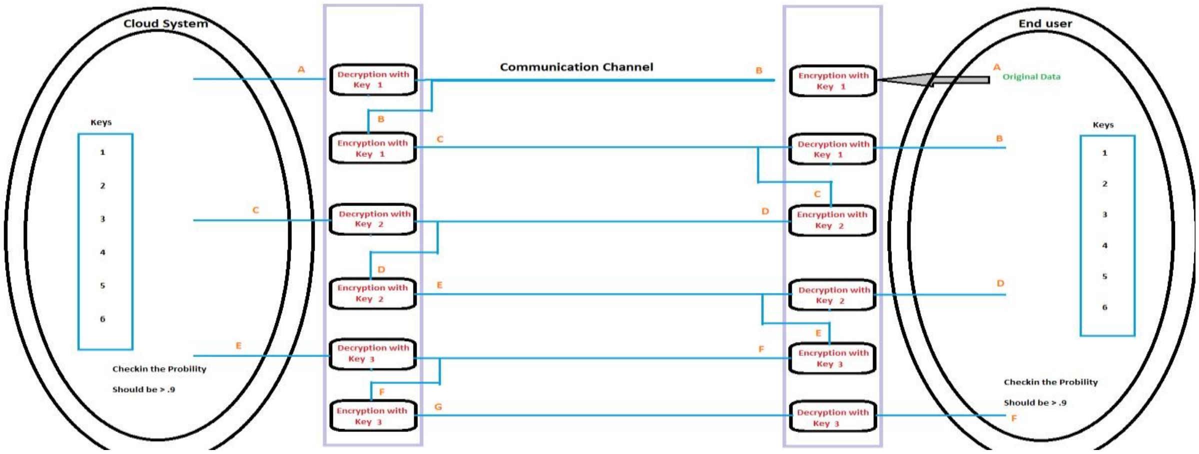


Fig 2 Flow Diagram

Lets take ”A” is the request message to be send from end-user and this message is encrypted with Key1, so the encrypted message will be Encrypt(A,Key1)→B. Message B is sent through communication channel and this has been decrypted with Key1 in the cloud server Decrypt(B,Key1)→A and this message will be stored in cloud server. Again message B is encrypted with Key1 Encrypt(B,key1)→C [Encrypt(Encrypt(A,Key1), Key1)] and C will be sent to end-user and message C will be decrypted with Key1 Decrypt(C,Key1)→B. Output of end-user is B and also the message received from the cloud server is also B, so input message should matches with the output message. Hence the message is communicated securely and no has interrupted the message in the communication channel

4. SOLVING LINEAR EQUATION

This matrix calculation is used to find out the probability of the output and the input message which is communicated between end-user and the cloud server. Let’s take [1 2 3 4 5 6] is the generated keys which is shared between end-user and the cloud server and [9 8 7] is the request message or data which is sent to the cloud server for validation or store data in the cloud server. Once the Keys are generated and need to share between the end-user and the cloud server. Sharing of the keys should be through communication channel. There may be a chance of altering or hacking the secret keys. Suppose the message [9 8 7] is encrypted with the Key1, its giving the some encrypted value and which should be stored in variable and again in the second iteration if we encrypt the same encrypted message with the Key1 which is available in cloud server and send to end-user, it gives the different value and stored in the variable. Here we should find the difference in the variables which is generated in the first and second iterations and the difference should be greater than .9. Suppose, if the key is altered (Key 6 is modified to 7 as shown in the below Fig.3) and encrypt the same message and will produce some result and the difference should be always less than .9 (refer the below Fig.3). Same process will be manipulated on both end-user and the cloud server. If the probability value is less than .9, then the end-user or cloud server will send a request message to generate a new set of keys for a secure communication. Again the same process will be carried out to validate the input and the output message for the secure communication.

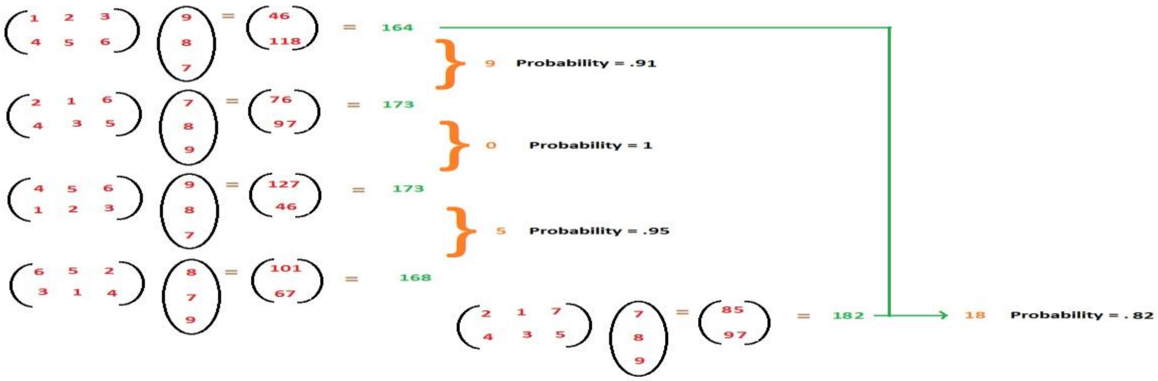
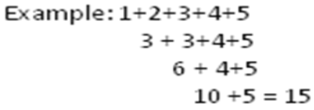


Fig 3 Solving Linear Equation

5. HOMOMORPHIC ENCRYPTION

5.1 Deterministic Method:

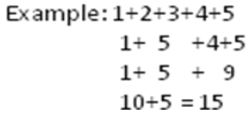
In this method, we are sending the encrypted message in a sequence of message. For e.g. “I am in London” is the message which I need to send to the cloud server. From the end-user, first we will to encrypt the “I” and send to the cloud server through the communication channel and then encrypt the “am” and send to the cloud server and then “in” and then “London”. Hacker can easily identify the sequence, if they identify the sequence “I am __ London”, then hacker can easily identify the message “in”. Then hacker can easily frame the sentence.



```
Example: 1+2+3+4+5
            3 + 3+4+5
               6 + 4+5
                 10 +5 = 15
```

5.2 Non-Deterministic Method:

In this method, we are sending the encrypted message in a sequence of message. For e.g. “I am in London” is the message which I need to send to the cloud server. From the end-user, first we will encrypt the “London” and send to the cloud server through the communication channel and then encrypt the “in” and send to the cloud server and then “am” and then “I”. Hacker cannot easily identify the sequence.



```
Example: 1+2+3+4+5
           1+ 5  +4+5
           1+ 5  +  9
           10+5 = 15
```

We formulate the problem of securely outsourcing large-scale systems of LE via iterative methods, and provide mechanism designs fulfilling input/output privacy. Provide mechanism designs fulfilling input/output privacy, cheating resilience, and efficiency.

6. LIMITATIONS

Here we are using Deterministic Method using Multi-Key. Arranging the received message is easy in Deterministic method. But, in Non-Deterministic with multi-key, the data will be sent in different sequence of message as discussed in section 5.2. This Multi-key concept requires more iteration to retrieve the original message from the encrypted message. Increase in the number of keys leads to increase in the more number of iteration and also to improve the security and also decrease the chance of intruder to decrypt the message.

7. KEY CHARACTERISTIC

Agility improves with users' ability to re-provision technological infrastructure resources.

Cost is claimed to be reduced and in a public cloud delivery model capital expenditure is converted to operational expenditure. This is purported to lower barriers to entry, as infrastructure is typically provided by a third-party and does not need to be purchased for one-time or infrequent intensive computing tasks. Pricing on a utility computing basis is fine-grained with usage-based options and fewer IT skills are required for implementation. The e-FISCAL project's state of the art repository contains several articles looking into cost aspects in more detail, most of them concluding that costs savings depend on the type of activities supported and the type of infrastructure available in-house.

Virtualization technology allows servers and storage devices to be shared and utilization be increased. Applications can be easily migrated from one physical server to another.

Multi tenancy enables sharing of resources and costs across a large pool of users thus allowing for:

Centralization of infrastructure in locations with lower costs (such as real estate, electricity, etc.)

Utilization and efficiency improvements for systems that are often only 10–20% utilized.

Reliability is improved if multiple redundant sites are used, which makes well-designed cloud computing suitable for business continuity and disaster recovery.

Performance is monitored and consistent and loosely coupled architectures are constructed using web services as the system interface.

Security could improve due to centralization of data, increased security-focused resources, etc., but concerns can persist about loss of control over certain sensitive data, and the lack of security for stored kernels. Security is often as good as or better than other traditional systems, in part because providers are able to devote resources to solving security issues that many customers cannot afford. However, the complexity of security is greatly increased when data is distributed over a wider area or greater number of devices and in multi-tenant systems that are being shared by unrelated users. In addition, user access to security audit logs may be difficult or impossible. Private cloud installations are in part motivated by users' desire to retain control over the infrastructure and avoid losing control of information security.

Maintenance of cloud computing applications is easier, because they do not need to be installed on each user's computer and can be accessed from different places.

8. CONCLUSIONS

This paper describes about to improve the security issue in Cloud Server when end-user trying to access the cloud server data from across the globe. Accessing the cloud data from public place is prone to change or modify the original message. So in order to improve the security obviously we need to increase the keys to encrypt and also increase the computation to validate the message.

REFERENCES

[1] Harnessing the Cloud for Securely Outsourcing Large-Scale Systems of Linear Equations, Cong Wang, Member, IEEE, Kui Ren, Senior Member, IEEE, Jia Wang, Member, IEEE, and Qian Wang, Member, IEEE - 2013

[2] C. Gentry, “Computing Arbitrary Functions of Encrypted Data,” Comm. ACM, vol. 53, no. 3, pp. 97-105, 2010.

[3] K. Forsman, W. Gropp, L. Kettunen, D. Levine, and J. Salonen, “Solution of Dense Systems of Linear Equations Arising from Integral-Equation Formulations,” IEEE Antennas and Propagation Magazine, vol. 37, no. 6, pp. 96-100, Dec. 1995.

[4] A. Edelman, “Large Dense Numerical Linear Algebra in 1993: The Parallel Computing Influence,” Int’l J. High Performance Computing Applications, vol. 7, no. 2, pp. 113-128, 1993.

[5] B. Carpentieri, “Sparse Preconditioners for Dense Linear Systems from Electromagnetic Applications,” PhD dissertation, CERFACS, Toulouse, France, 2002

[6] R. Cramer and I. Damga°rd, “Secure Distributed Linear Algebra in a Constant Number of Rounds,” CRYPTO: Proc. Ann. Int’l Cryptology Conf. Advances in Cryptology, 2001.

[7] P. Mohassel and E. Weinreb, “Efficient Secure Linear Algebra in the Presence of Covert or Computationally Unbounded Adversaries,” CRYPTO: Proc. 28th Ann. Int’l Cryptology Conf., pp. 481-496, 2008.

[8] J.R. Troncoso-Pastoriza, P. Comesan˜ a, and F. Pe´rez-Gonza´lez, “Secure Direct and Iterative Protocols for Solving Systems of Linear Equations,” Proc First Int’l Workshop Signal Processing in the EncryptEd Domain (SPEED), pp. 122-141, 2009.

[9] W. Du and M.J. Atallah, “Privacy-Preserving Cooperative Scientific Computations,” Proc. IEEE 14th Computer Security Foundations Workshop (CSFW), pp. 273-294, 2001