IMPACT: International Journal of Research in Engineering & Technology (IMPACT: IJRET)
ISSN(E): 2321-8843; ISSN(P): 2347-4599
Vol. 2, Issue 6, Jun 2014, 203-210

TRUSTED THIRD PARTY SERVICE FOR SECURE CLOUD DATA

R. S. NEJKAR[1] & G. A. PATIL[2]

[1]Student, ME (CSE)-II, D.Y. Patil College of Engineering and Technology, Kolhapur, Maharashtra, India

[2]Professor, D.Y. Patil College of Engineering and Technology, Kolhapur, Maharashtra, India

ABSTRACT

Data security and access control is one of the most challenging ongoing research work in cloud computing, because users are outsourcing their sensitive data to cloud providers. Existing solutions uses pure cryptographic techniques to mitigate these security and access control problems. These solutions suffer from heavy computational overhead on the data owner as well as the cloud service provider for key distribution and management. We address this challenging issue using capability based access control with trusted third party to ensure valid users will access the outsourced data.

KEYWORDS: Cloud Computing Secure Data Access, Third Party in Cloud

INTRODUCTION

Cloud computing is “a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g. networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction”. Data owners store their data on external servers, there have been increasing demands and concerns for data confidentiality, authentication and access control.

Besides confidentiality and privacy breaks, the external servers could also use part of the data or whole for their financial gain and hence lose the data owners market or even bringing economic losses to the data owner. These concerns originate from the fact that cloud servers are usually operated by commercial providers which are very likely to be outside of the trusted domain of users.

RELATED WORK

A number of different mechanisms have been proposed for security aspects in cloud computing. Some of the researchers have suggested the following strategies to support secure access in cloud computing.

Sunil Sanka, Chittaranjan Hota, Muttukrishnan Rajarajan [1] address capability based access control technique that ensures only valid users will access the outsourced data. The authors also propose a modified Diffie-Hellman key exchange protocol between cloud service provider and the user for secretly sharing a symmetric key for secure data access but key management and distribution scenarios are not mentioned.

Zhidong Shen, Qiang Tong [2] proposed a method to build a trusted computing environment for cloud computing system by integrating the trusted computing platform into cloud computing system. In this model, some important security services, including authentication, confidentiality and integrity are proposed in cloud computing system but there is no mention for data integration.

Weichao Wang, Z. Li, R. Owens, and B. Bhargava [3] propose to encrypt every data block with a different key so that flexible cryptography-based access control can be achieved. Through the adaptation of key derivation methods, the owner needs to maintain only a few secrets but scheme for key management is not mentioned.

PRESENT SCHEME

Model consists of three participants Data Owner (DO), Cloud Service Provider (CSP) and users. The DO places the data on the CSP which user wants to access. The data owner computes a message digest using MD5 for every file belonging to the data set available with it. DO then updates the capability list with a new entry for every user and the data item that can be accessed by the user. DO then send everything encrypted using its private key first and then using public key of the CSP for the purpose of authentication and confidentiality between CSP and DO. CSP uses its own private key and the public of DO to decrypt the message and store the encrypted data files and capability list in its' storage.

When a new user is to be added, it needs to send a registration request to the data owner. After receiving a request, data owner adds an entry into the capability list if it is a valid request. For simplicity we assume that the DO has a separate procedure for verifying the genuineness of the client request. DO now sends the capability list and an encrypted message intended for user with all the key parameters needed at user for decrypting the data files to CSP. CSP now updates its capability list and sends a registration reply to user. After the keys are made available to the user, now that the actual data access request goes from a user to the CSP. If request is valid, CSP send encrypted data to the user through TTP.

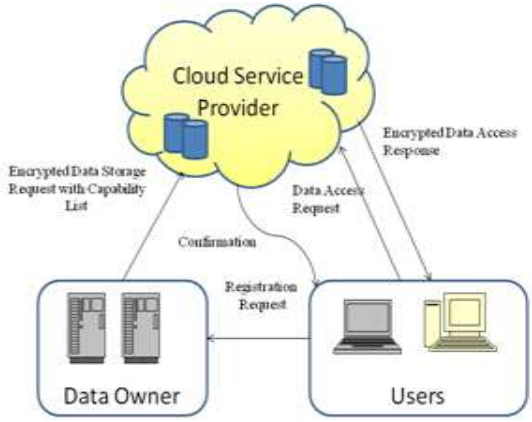

Figure 1: Present Scheme

The user upon receiving an encrypted response from the CSP can decrypt the message and calculates the digest by using the hash function. The newly calculated digest is then compared with the digest that is attached with the message to check the integrity of the message.

The problems with secure data access in cloud computing model can be summarized as below.

- Difficulty in knowing key of appropriate Data Owners, if multiple Data Owners exists.
- Issues in remembering Key to the Users if users loose the key due to system crash, as it is required to access data from CSP.
- Data Owner with poor computing capability becomes bottleneck if multiple users send request to single Data Owner simultaneously.

To overcome the above problems we propose a system which will do the task of system manager. This task can be got done by third party auditor who must be trusted.

PROPOSED SCHEME

Figure 2 Shows Secure Data Access in Cloud Computing Model Using Trusted Third Party. It Has Four Participants Data Owner (do), Cloud Service Provider (csp), Trusted Third Party (ttp) and Users.

The proposed system will be designed and implemented in the following manner,

- Calculating message digest and storing object to CSP.
- Registration, Updating Capability lists and Response.
- Secure Data Access Mechanism.

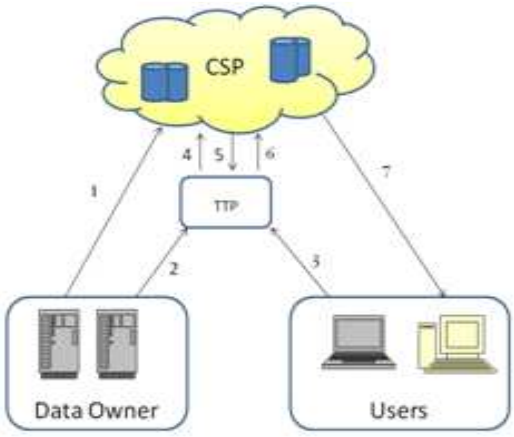

Figure 2: Proposed System Using TTP

Calculating Message Digest and Storing Object to CSP

The proposed system places encrypted data on the CSP which user wants to access. It first computes a message digest using MD5 for every file belonging to the data set available with it. Then encapsulates this digest along with the file using a symmetric key. This in turn gives cryptographic strength much more than using the SHA-1. This ensures data confidentiality and integrity between owners and users.

Registration, Updating Capability Lists and Response

Figure 3 depicts the process of registration, updating Capability lists and Response to Users. First Participants need to register themselves to the TTP. TTP contains list of all DO's and users already registered. After registration, our system gives authority to TTP to provide data to user as per request. Whenever user wants to access data, he needs to contact TTP for the appropriate DO. If multiple DO's are able to provide data and users don't know appropriate DO, then the system will provide list of DO's to the user.

From the available list, user will choose any one DO and send a request for data to the system. The system will update the capability list if it is a valid request. TTP will send the capability list and an encrypted message intended for user with all the key parameters needed at user side for decrypting the data files coming from CSP. CSP will also update its capability list and sends a reply to user which is coming from TTP.

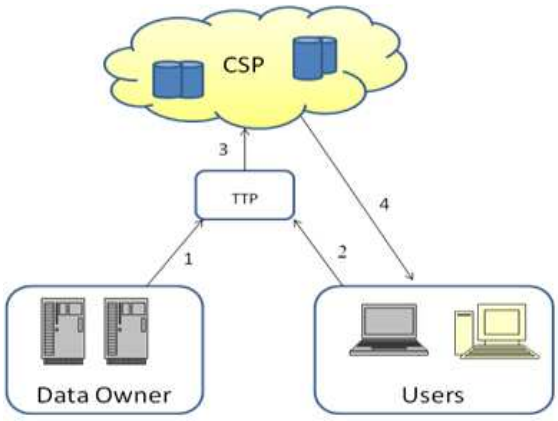

Figure 3: Registration, Updating Capability Lists and Response to Users

Secure Data Access Mechanism

Figure 4 depicts secure data access mechanism. User will send data access request to CSP with the response sent by the system. When request goes from user to TTP, TTP will check validity of user. If user is valid, encrypted data send from CSP to the user through TTP. After receiving an encrypted response from the CSP user will decrypt the message using AES algorithm and then calculates the message digest using MD5 algorithm. User will check message integrity by comparing calculated digest with the attached digest to the message. In this way user can access data from CSP.

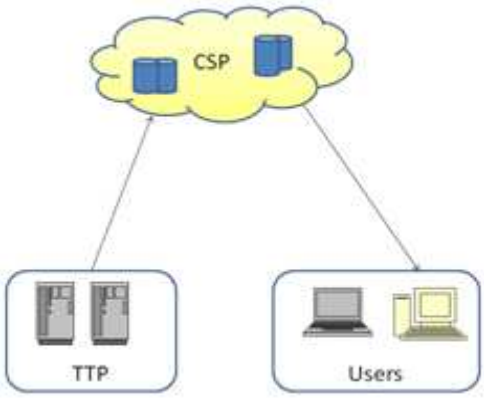

Figure 4: Secure Data Access Mechanism

WORK DONE SO FAR

Calculating Message Digest and Storing Object on CSP

Our system consists of four components Data Owner (DO), Trusted Third Party (TTP), Cloud Service Provider (CSP) and User. In this module, the functionality of DO is developed to place encrypted data on the CSP which user can access. DO first computes a message digest using MD5 and then encapsulates this digest along with the file using AES algorithm. Once encapsulated DO places encrypted data on the CSP which any user can access.

The details of MD5 Message Digest and AES used for this purpose are as mentioned below:

- **Message Digest**

The **MD5 Message-Digest** tool is a widely used cryptographic hash function that produces a 128-bit (16-byte) hash value. An MD5 hash is typically expressed as a 32-digit hexadecimal number.

Algorithm

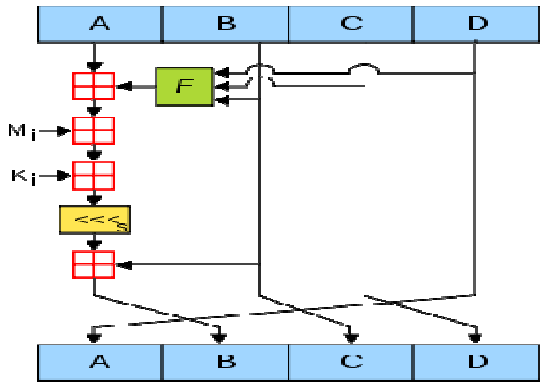

Figure 5: One MD5 Operation

MD5 consists of 64 of these operations, grouped in four rounds of 16 operations. F is a nonlinear function; one function is used in each round. M_i denotes a 32-bit block of the message input, and K_i denotes a 32-bit constant, different for each operation. $\lll_s$ denotes a left bit rotation by s places; s varies for each operation. $\boxplus$ denotes addition modulo 2^{32}.

- **AES**

AES is based on a design principle known as a substitution-permutation network, and is fast in both software and hardware. AES operates on a 4×4 column-major order matrix of bytes, termed the state.

The key size used for an AES cipher specifies the number of repetitions of transformation rounds that convert the input, called the plaintext (original data + message digest), into the final output, called the cipher text (encrypted data). The number of cycles of repetition are as follows: 10 cycles of repetition and 128-bit keys.

Each round consists of several processing steps, each containing four similar but different stages, including one that depends on the encryption key itself. A set of reverse rounds are applied to transform ciphertext back into the original plaintext using the same encryption key.

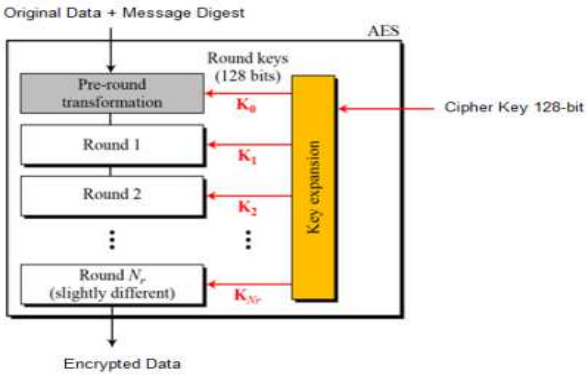

Figure 6: AES Encryption Cipher

- **Output**

When user wants to upload his data, he needs to register as a Data Owner. The new data owner can subscribe to the application by signing up. The Data Owner can create and login into the account. Data Owner can upload his encrypted data on Cloud.

Figure 7 Shows The Process to Upload Encrypted Data:

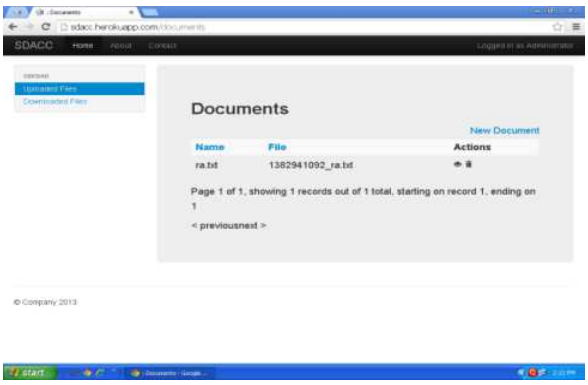

Figure 7: Process to Upload Encrypted Data

Registration, Updating Capability Lists and Response

- **Registration & Updating Capability Lists**

In our system, if user wants to access data, he needs to register to the TTP. New user can subscribe to the application by signing up. After user register to the TTP, he contact to the TTP for the appropriate DO. Our system will provide capability list for users which contain list of DO's, there data and actions to be taken out for the data. Our system has also updating capability lists automatically. When DO upload new data to the CSP or delete his data available to the CSP then our system automatically update capability list.

DO (admin108) add new data as a file (rahul1.txt) and (rahul2.txt) on CSP. After uploading data to CSP, capability lists updated automatically. Figure 8 shows process for updating capability lists.

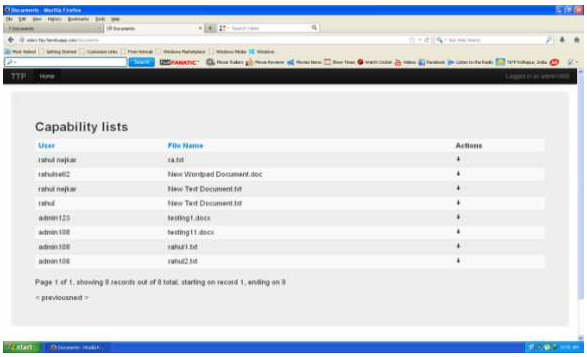

Figure 8: Process for Updating Capability Lists

- **Response**

From the available capability lists, user will choose data from one DO and send a request for data to system. When request send from user, he waits until request is accepted on CSP.

Request on CSP contains name of user who send request, file name which user want to access and request status. For accept request our system change status as approved "no" to "yes". Once System will accept request, system will send reply to user for data access.

Figure 9 Shows Request for Data From User. As Shown in Figure User Select Data (rahul2.txt) from do (admin108).

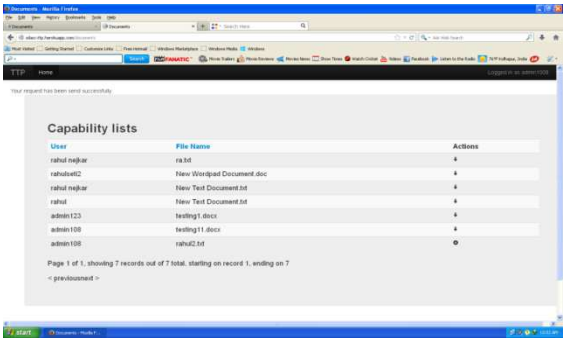

Figure 9: Request for Data from User

As request send from user, request goes to CSP for approval. Figure 4 shows request on CSP from user. User (admin1008) send request for file (rahul2.txt) and wait for approve request from “no” to “yes”.

As soon as request is accepted on CSP, response will generated to the user. Figure 10 shows response to the user on TTP.

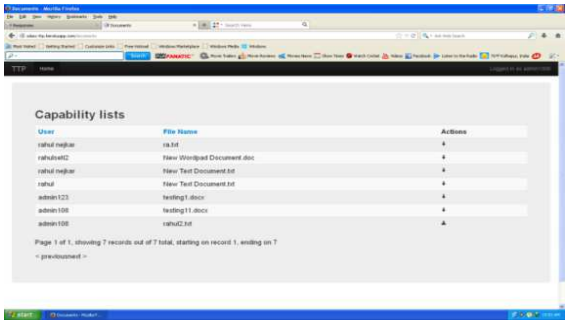

Figure 10: Response to the User on TTP

- **Secure Data Access Mechanism**

Once response is generated, User will send data access request to CSP with the response sent by the system. When request goes from user to TTP, TTP will check validity of user. If user is valid, encrypted data send from CSP to the user through TTP.

After receiving an encrypted response from the CSP user will decrypt the message using AES algorithm and then calculates the message digest using MD5 algorithm. User will check message integrity by comparing calculated digest with the attached digest to the message. In this way user can access data from CSP. Figure 11 shows data available to the user.

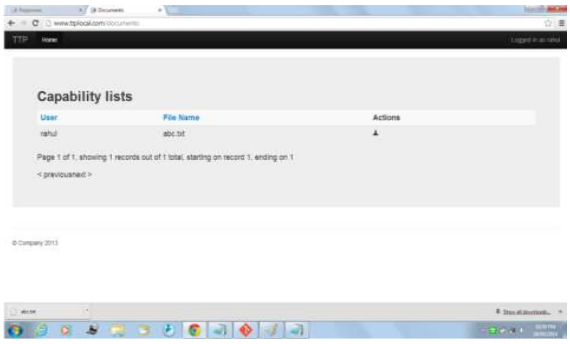

Figure 11: Data to the User

CONCLUSIONS

Data security is important in cloud computing. To achieve this we are using secure data access model which has some problems. So we are proposed one system which will do the task of system manager to sustain the advantages of proposed approach. So this task is done by third party auditors.

Our Proposed scheme empowers the data owner to outsource the security enforcement process on the outsourced data files without losing control over the process. Moreover, our scheme can also delegate most of the computation overhead to cloud servers.

REFERENCES

1. Sunil Sanka, Chittaranjan Hota, Muttukrishnan Rajarajan, "Secure data Access in Cloud Computing", in 4th International conference on Internet Multimedia Services Architecture and Application(IMSAA), 2010 IEEE.
2. Zhidong Shen, Qiang Tong, "The Security of Cloud Computing System enabled by Trusted Computing Technology", in second International Conference On Signal Processing Systems, 2010 IEEE.
3. Weichao Wang, Z. Li, R. Owens, and B. Bhargava, "Secure and efficient access to outsourced data," in Proc. of ACM Cloud Computing Security Workshop, pp. 55-65, 2009.
4. Peter Mell, and Tim Grance, "Draft NIST Working Definition of Cloud Computing," 2009, from http://csrc.nist.gov/groups/SNS/cloud-computing/
5. R. Buyya, C. S. Yeo, and S. Venugopal, "Market oriented cloud computing: vision, hype, and reality, for delivering IT services as computing utilities," in Proc. of 10th IEEE International Conference on High Performance Computing and Communications, 2008.
6. M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. H. Katz, A. Konwinski, G. Lee, D. A. Patterson, A. Rabkin, I. Stoica, and M. Zaharia, "Above the clouds: A berkeley view of cloud computing, " University of California, Berkeley, Tech Rep USB-EECS-2009-28, Feb 2009.
7. David Chappell, "Introducing the Azure Service Platform," White paper, Oct 2008.
8. Amazon EC2 and S3, Online at http://aws.amazon.com/
9. S. D. C. di Vimercati, S. Foresti, S. Jajodia, S. Paraboschi, and P. Samarati, "A Data Outsourcing Architecture Combining Cryptography and Access Control," in Proc. of ACM Workshop on Computer Security Architecture (CSAW'07), Nov 2007, USA.