International Journal of Scientific Research in Computer Science, Engineering and Information Technology
© 2018 IJSRCSEIT | Volume 3 | Issue 5 | ISSN : 2456-3307

Secure File Sharing between Multiple Clouds using Encrypted Mechanism

N. Meena[1] , Pydimani Lakshmi[2]

[1]Student, Department of Computer Science, Sri Venkateswara College of Engineering For Women, Karakambadi Road, Tirupati, Andhra Pradesh, India

[2]Assistant Professor, Department of Computer Science And Engineering,Sri Venkateswara College of Engineering For Women, Karakambadi Road, Tirupati, Andhra Pradesh, India

ABSTRACT

A large amount of the data is stored in the file one can retrieve or access the stored data easily. Now-a-days most of the people are concentrate on the file sharing for sharing the important data. But many techniques are concentrating on owner to single user filesharing. In this paper, we propose File share for the secure data sharing between the multiple organizations in the different clouds for the multiple users. The proposed protocol is based on the revocable key- policy Attribute- Based Encryption schemes and it allows a lot of users to share the data in the cloud based on the policy which is defined by the data owner . Furthermore, access to a malicious or compromised user/organization can be easily revoked without the need to generate fresh encryption keys. The file is shared to the multiple users of the particular category by the owner. By this a lot of time is saved as the data corresponding to the particular category are shared to all the users registered with that particular category.

Keywords : Security, Data sharing, Cloud computing, key policies, SHA (Secure Hash Algorithm1).

I. INTRODUCTION

The term Cloud refers to a Network or Internet. In other words, we can say that Cloud is something, which is present at remote location. Cloud can provide services over public and private networks, i.e., WAN, LAN or VPN. Cloud computing enables the clients to store their information in the cloud and can retrieve it whenever by utilizing the internet. It provides three types of services. The main service is IaaS(Infrastructure as a Service).

Generally files are used to store the data securely and the stored data is also secured by using the keys for the different files. In some conditions the stored need to be shared to the other users. In cloud computing the data is stored in the server side which is uploaded by the owner. The data which is stored in the files is shared to the authorized users of the cloud. While trasfering the file, the data security is maintained by using keys and different encryption or decryption policies.

In existing system we propose file share for sharing the data through files via cloud and it is based on a Revocable-Key Policy Attribute- Based Encryption scheme and allow users to share the encrypted data based on a policy that has been defined by the data owner. Furthermore, access to a malicious or compromised user/organization can be easily revoked without the need to generate fresh encryption keys. By this file sharing is done between single user and the cloud. This case may fails in several situations like sending same messages or file to the category of the users.

To eliminate the above drawbacks we are proposing a system in which the cloud service provider can send the same files or messages to the users that corresponds to the particular category. In this paper during the registration phase the user has to select the particular category he want from the cloud service provider. The Cloud service provider may send some information or files regarding the particular category to all the users of that particular category. If the user is interested in the file which is send by the cloud service provider the user can send the request to the cloud service provider for the generation of the key. The cloud service provider can see the request which is send by the user and can generate key for the particular file of the user. The user can get the key and can download the file by using the key. And the data in the file is also encrypted by the cloud service provider and the user can view only the encrypted data before downloading the file. Once he got the key the user can download the file which is in plain text. For this encryption and decryption of the file SHA (Secure Hash Algorithm1) is used.

The proposed protocol is based on the revocable key- policy Attribute- Based Encryption schemes and it allows a lot of users to share the data in the cloud based on the policy which is defined by the data owner.

Revocable Key-Policy ABE: A revocable KP-ABE plot is a tuple of the accompanying five calculations:

1. Setup is a probabilistic calculation that takes as information a security parameter λ and yields an open key pk and an ace key MSK. We indicate this by
(pk,MSK) ← Setup(1λ).

2. Gen is a probabilistic calculation that takes as info an ace key, an arrangement P ∈ P and the interesting identifier of a client and yields a mystery key which is tie both to the relating strategy and client. We indicate this by
(skP,ID) ← Gen(MSK, P, ID).

3. Enc is a probabilistic calculation that takes as info an open key, a message m, an arrangement of traits S ∈ Ω and a timestamp t. After a legitimate run, the calculation yields a ciphertext cS,t which is tie both to the arrangement of traits and the time. We indicate this by
(cS,t) ← Enc(pk,m, S, t).

4. KeyUpdate is a probabilistic calculation that takes as info an ace key, a repudiation list rl and a timestamp t and yields a key refresh data for time t. We indicate this by (Kt) ← KeyUpdate(MSK, rl, t).

5. Dec is a deterministic calculation that takes as information a mystery key, a key refresh Kt1 and a ciphertext and yields the first message m iff the arrangement of traits S that are tie to the ciphertext fulfills the approach P, t1≥ t and the ID of the relating client was not disavowed at time t.We denote this by Dec(skP,ID,K t^1, cS,t) → m.

II. ALGORITHM

SHA (Secure Hash Algorithm1):

In cryptography, SHA-1 (Secure Hash Algorithm 1) is a cryptographic hash work which takes an info and produces a 160-piece (20-byte) hash esteem known as a message process - commonly rendered as a hexadecimal number, 40 digits in length. Inside the group of secure hash calculations, there are a few cases of these instruments that were set up to encourage better advanced security. The first, SHA-0, was produced in 1993. Like its successor, SHA-1, SHA-0 highlights 16-bit hashing.

These protected hash calculations are a piece of new encryption gauges to keep delicate information safe and avert diverse kinds of assaults. Albeit a portion of these were created by offices like the National

Security Agency, and some by free designers, every one of them are identified with the general elements of hash encryption that shields information in certain database and system situations, advancing cybersecurity in the computerized age.

SHA-1 delivers a message process in light of standards like the outline of the MD4 and MD5 message process calculations, however has a more preservationist plan. SHA-1 frames some portion of a few generally utilized security applications and conventions, including TLS and SSL, PGP, SSH, S/MIME, and IPsec. Those applications can likewise utilize MD5; both MD5 and SHA-1 are slipped from MD4. The calculation has likewise been utilized on Nintendo's Wii gaming console for signature check while booting, yet a huge blemish in the main executions of the firmware took into account an assailant to sidestep the framework's security plot. Nobody has been able to break SHA-1, but the point is the SHA-1, as far as Git is concerned, isn't even a security feature. It's purely a consistency check. The security parts are elsewhere, so a lot of people assume that since Git uses SHA-1 and SHA-1 is used for cryptographically secure stuff.

The SHA-1 algorithm steps are as follows:

1. Intialize the variables.
2. Apply pre-processing.
3. Process the message in successive 512 bits.
4. Extend the sixteen 32-bit words into eight 32 words.
5. Intialize hash value for each chunk.
6. Apply chunk hash value to the result.
7. Produce the final hash value as 160 bit number.

Note 1: All variables are unsigned 32-bit quantities and wrap modulo 232 when calculating, except for ml, the message length, which is a 64-bit quantity, and hh, the message digest, which is a 160-bit quantity.

Note 2: All constants in this pseudo code are in big endian.

Within each word, the most significant byte is stored in the leftmost byte position

III. CONCLUSION

In this paper we have shown the file sharing in a secure manner by encrypting the data in the file and the key used for the file sharing. We have shown that the user can select the category while registering to the cloud. The Cloud service provider then send all the information regarding to that category to the users who are under that particular category that too in the encrypted format. Once the user is interested in the data which CSP has sent then he can send the request to the CSP to provide the key. Once the Cloud Service Provider send the key then the user can download the file. And the data in the file is also encrypted by the cloud service provider and the user can view only the encrypted data before downloading the file. Once he got the key the user can download the file which is in plain text. For this encryption and decryption of the file SHA (Secure Hash Algorithm1) is used.

IV. REFERENCES

1. R Dowsley, A. Michalas, and M. Nagel, "A report on design and implementation of protected searchable data in iaas," tech. rep., Swedis Institute of Computer Science (SICS), 2016.
2. Y Verginadis, A. Michalas, P. Gouvas, G. Schiefer, G. Hbsch, and I. Paraskakis, "Paasword: A holistic data privacy and security by design framework for cloud services," in Proceedings of the 5th International Conference on Cloud Computing and Services Science, pp. 206 213, 2015.
3. A Michalas and K. Y. Yigzaw, "Locless: Do you really care your cloud files are?," in 2016 IEEE/ACM 9th International Conference on Utility and Cloud Computing (UCC), pp. 618-623, Dec 2015.

4. A Michalas, "Sharing in the rain: Secure and efficient data sharing for the cloud," in 2016 International Conference for Internet Technology And Secured Transactions, pp. 589-595, Dec 2016.
5. N Paladi, A. Michalas, and C. Gehrmann, "Domain based storage protection with secure access control for the cloud," in Proceedings of the 2014 International Workshop on Security in Cloud Computing, ASIACCS 14, (New York, NY, USA), ACM, 2014.
6. Y Verginadis, A. Michalas, P. Gouvas, G. Schiefer, G. Hubsch, and I. Paraskakis, "Paasword: A holistic data privacy and security by design framework for cloud services," pp. 1-16, 2017.
7. A Sahai and H. Seyalioglu, "Dynamic credentials and ciphertext delegation for attribute-based encryption," in in Proceedings of the 32nd Annual International Cryptology Conference: Advances in Cryptology - CRYPTO2012, pp. 199-217, Springer, 2012.
8. D Dolev and A. C. Yao, "On the security of public key protocols," Information Theory, IEEE Transactions on, vol. 29, no. 2, 1983.
9. A Michalas, N. Komninos, N. R. Prasad, and V. A. Oleshchuk, "New client puzzle approach for dos resistance in ad hoc networks," in Information Theory and Information Security (ICITIS), 2010 IEEE International Conference, pp. 568-573, IEEE, 2010.
10. A. Michalas, N. Komninos, and N. R. Prasad, "Mitigate dos and ddos attack in mobile ad hoc networks," International Journal of DigitaCrime and Forensics (IJDCF), vol. 3, no. 1, pp. 14-36, 2011.
11. A. Michalas, N. Komninos, and N. Prasad, "Multiplayer game for ddos attacks resilience in ad hoc networks," in Wireless Communication, Vehicular Technology, Information Theory and Aerospace Electronic Systems Technology (Wireless VITAE), 2011 2nd International Conference on, pp. 1-5, Feb 2011.
12. A. Michalas, N. Komninos, and N. R. Prasad, "Cryptographic puzzles and game theory against dos and ddos attacks in networks," International Journal of Computer Research, vol. 19, no. 1, p. 79, 2012.
13. T. Dimitriou and A. Michalas, "Multi-party trust computation in decentralized environments," in 2012 5th International Conference on New Technologies, Mobility and Security (NTMS), pp. 1-5, May 2012.
14. T. Dimitriou and A. Michalas, "Multi-party trust computation in decentralized environments in the presence of malicious adversaries," Ad Hoc Networks, vol. 15, pp. 53-66, Apr. 2014