World Applied Sciences Journal 29 (Data Mining and Soft Computing Techniques): 146-148, 2014
ISSN 1818-4952

DOI: 10.5829/idosi.wasj.2014.29.dmsct.27

Protection of User Sensitive Data Using Automatic Slaughter Scheme

P. Kayalvizhi, S. Jenifer, R. Vijay Sai and M. Lavanya

School of Computing, SASTRA University, Tamil Nadu, India

Abstract: Technology in this century has improved a great deal. It cannot be assured everything that the technology will facilitate us to build our life more comfortable. More diligences are coming forth along the internet to get our information unsafe and with the evolution of technology it becomes easier. So we are proposing a scheme which does not rely on a third party. The most comfortable path to protect our data is to cipher the data in prior and spill the public key to the licensed users. Nevertheless, the information can be exposed while the public key is revealed. Here, we present a scheme called Automatic Slaughter Scheme (ASS). This is achieved by ciphering the data and by placing both the public key and the cipher text into the distributed hash table network. This scheme facilitates the users to control the lifetime of the data.

Key words: Third Party · Cipher · Distributed Hash Table

INTRODUCTION

As people depend more on the Internet, the protection of their privacy becomes risky. The faster growth of computer network and communication technology creates an easy means for exposing user sensitive information. If this problem is not considered, it will lead to leakage of messages.

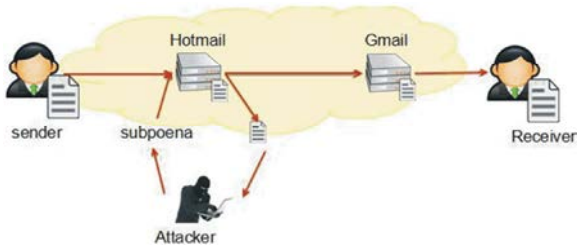
Considering the scenario [fig 1], the sender sends the user sensitive data to the receiver through server like Gmail. The sender and receiver trust their confidentiality of their emails. The most comfortable path to cover our data is cipher the data in prior and spills the public key to the licensed users. Even though users sanitize their files, its duplicates may be reserved for an elongation time by its providers. By the mean time attacker can attack the sensitive information.

To resolve this issue, it required a well-organized scheme, which empowers the perceptive end user data to be automatically slaughtered. More generally, self-destructing data are generally applicable in today's Web- focused universe, where user sensitive information can stay in the internet indefinitely. The automatic slaughtering scheme which enables end users to reduce the lifespan of their perceptive data is suitable in up-to-date web-focused universe. A method proposed by Geambasu *et al* [2] saves the part of the decryption key in a wide distributed hash table (DHT) network. Without using any elimination operation by the end user, it enhances the electronic data to be slaughtered automatically after a period of fourth dimension. But, there is a clear deficiency in their system, where only the public key is deleted and the whole encrypted data can still be accessed.

The conventional method such as brute force [3] can be utilized by the opponents which offend the encrypted text, even when the decoding key is not made to be available. To evade this situation, we offer an automatic slaughter scheme (ASS) for electronic data which enables both the public key and the encrypted text destructed to maintain the information protection. ASS scheme can be used to make the electronic information automatically slaughtered after a point of time and it can resist the attack in the DHT network.

Related Works: To thwart the data leakage in the internet, one way is to destruct the data manually. Some of the destruction tools used for manual deleting such as a software erasure program, shredding equipments and hence as mentioned in papers [3, 4]. Electronic mail and other electronic information are stacked away on providers where end user cannot control the process.

Corresponding Author: R. Vijay Sai, School of Computing, SASTRA University, Tamil Nadu, India.



The user can delete the email hoping that it is destroyed correctly. But he has no idea on whether the data is really destructed. Only licensed users can follow alternate approach by applying conventional encryption methods to reveal the decryption key. This system can be ineffective since the decryption key may be revealed to unlicensed users. The Ephemerizer family of solutions can be suited for the above scenario [7, 8].The main disadvantage for this is that the reliable third parties may mislead in certain instances. They may expose it for creating profits.[9].Another major approach is the SSDD scheme, which encrypts information and control the life span of the decryption key.

Geambasu extended this scheme where electronic information is automatically killed after some time [2]. Applying the symmetric encryption and storing the shares of the public key in a DHT network is the objective of their scheme to enable encryption.

In their strategy, the decryption key is destroyed and the whole encrypted information is still accessible. Another scheme (SSDD) as Geambasu et al distributes the decryption key and encrypted information into DHT network. In this scheme the cipher text cannot be decrypted without getting the decryption key and part of the encrypted information [1]. ASS method makes a considerable increase in security by using asymmetric algorithm (RSA) for encryption and decryption of sensitive user data.

Proposed Method: In our proposed algorithm we encrypt the information and share it in distributed hash tables. The algorithm which is applied here is RSA which provides more protection than any symmetric algorithm. In this technique two steps are put into usage. Encapsulation and Decapsulation are the two important stages in this strategy. In encapsulation we will cipher the data first and bonds the cipher with a key and will place it in a distributed hash table and then we will get the data to be disappeared after a particular period of time from the hash table by deleting the part of the cipher text and the key. A DHT network has the property of providing spaces to current information by chucking out older data after some point of time. Protection of data by the unauthorized users is then applied.

In our proposed algorithm more security to the data is assured by going through it asymmetrically which cannot be easily offended by the attempts.

Algorithm: The algorithm described below explains the concept of this paper. The input to algorithm 1 is a random prime number p and q, random encryption key e and random public key d. The encryption key is selected at random and public keys and private keys are generated. The algorithm2 concentrates on encryption and decryption part by performing mod operation on the respective variables. The algorithm 3 is about Distributed Hash Table network, where public key and cipher text are inserted after encryption. Decryption of message is performed using public key. Then self destruction of data is executed.

Algorithm 1: Generation of public key and private key
Input: Random prime number p,q, random encryption key e, random public key d

- Set of two prime numbers at random are obtained.
- Two prime numbers are multiplied and stored in N.
- Encryption key e is selected at random.
- Public key which should be known to everyone puk{e,N} is generated
- Private key prk{d,p,N} is generated

Algorithm 2: Encrypt & Decrypt
Input: Public key puk, modulus mod N, private key prk, message M

- Compute C=M^e mod N
- Compute M=C^d mod N

Algorithm 3: DHT Network
Input: key1 puk, value C, key2 prk

- Insert public key and cipher text in DHT using insert (key1,value) after encryption.

Table 1: Comparison Analysis Table

Comparison item	SSDD Scheme	ASS Scheme
Key length	L1+L2+L3	L
Key space	2^L1+L2+L3	2^L
Key destruction	Can be destructible	Can be destructible
Encrypted data destruction	Can be destructible	Can be destructible
Protection from conventional attacks	YES	YES
Protection from brute force attacks	YES	YES
Protection from the attacks in the DHT table	YES	YES
Security	Can be broken under some circumstances	Cannot be broken by any of the attacks mentioned above

- While decrypting the message use public key to look up in the DHT lookup(key1).
- To self-destruct the data in DHT network use Delete (key1)

Experimental Analysis: Experimental analysis as seen in Table1 shows that our proposed scheme when compared with SSDD scheme, provides more security and offers data protection from third party intruders. It can be seen that under any circumstances, the security of data cannot be broken, thereby offering protection.. Many parameters such as key length, key space, key destruction and encrypted data destruction are compared and the analysis are tabulated.

CONCLUSION

Cryptography plays a crucial part in the development of network security. More diligences are coming forth in the internet to get our data unsafe and with the enhancement of technology it becomes easier. So we present a scheme called automatic-slaughter scheme for electronic data. This scheme will assist us to keep our confidential information from attacks. This is accomplished by encrypting the data using asymmetric encryption technique followed by casting it on a distributed hash table and then deleting the part of the cipher text and the public key from the hash table. We can decrypt the data by using the decrypting key which is known only to the authorized users. An asymmetric algorithm is applied to ensure more protection.

REFERENCES

1. Fengshun Yue, Guojun Wang and Qin Liu, 2010. A Secure Self-Destruction scheme for electronic data. In School of Information Science and Engineering Central South University.
2. Roxana Geambasu,Tadayoshi Kohno, Amit A. Levy and Henry M. Levy.Vanish: Increasing Data Privacy with Self-Destructing Data.
3. Lu, Z., T. Li, X. Hu, K. Zhao, J. Zeng and L. Peng, 2009. Data self-destruction method. In Application Research of Computers, 26(1) 350-355.
4. Dong, H., S. Kun and C. Yu, 2009. Research on secure destruction of digital information. In Proc. of International Conference on Apperceiving Computing and Intelligence Analysis, pp: 356-359.
5. Dong, H., S. Kun and C. Yu, 2009. Research on secure destruction of digital information. In Proc. of International Conference on Apperceiving Computing and Intelligence Analysis, pp: 356-359.
6. Sun, K., J. Choi, D. Lee and S. Noh, 2008. Secure deletion of confidential data in consumer electronics. In Digest of Technical Papers-IEEE International Conference on Consumer Electronics, pp: 385-386.
7. Nair, S.K., M.T. Dashti, B. Crispo and A.S. Tanenbaum, 2007. A hybrid PKI-IBC based ephemerizer system. In Proc. of International Information Security Conference, pp: 241-252.
8. Perlman, R., 2005. The ephemerizer: making data disappear. In Journal of Information System Security, 1(1): 21-32.
9. Singel, R., 2007. Encrypted e-mail company Hushmail spills to feds. http://www.wired.com/threatlevel/ 2007/11/encrypted-e-mai/
10. Rhea, S., B. Godfrey, B. Karp, J. Kubiatowicz, S. Ratnasamy, S. Shenker, I. Stoica and H. Yu, 2005. Open DHT: A public DHT service and its uses. In Proc. of ACM SIGCOMM, pp: 73-84.