Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 4, Issue. 7, July 2015, pg.299 – 304

RESEARCH ARTICLE

A Self-Destructive System for Data Privacy on Web Services

Ganesh Z Bhade[1], Prof. Vikrant Chole[2]

[1]Department of Computer Science and Technology
[1]G.H. Raisoni Academy of Engineering and Technology, Nagpur, India
[2]Department of Computer Science and Technology
[2]G.H. Raisoni Academy of Engineering and Technology, Nagpur, India

Abstract: Now a days more and more services and applications are emerging in the Internet, exposing sensitive electronic data in the internet has become easier. Web services causes personal data to be cached, copied, and archived by third parties, often without our knowledge or control. We propose a secure self-destructing scheme for data, which can protect a user's sensitive data by making the sensitive and important data automatically destructed after particular period of time. Specifically, we first encrypt the data into a cipher text. Then, we associate the cipher text, and extract a part of the cipher text to make it incomplete. There will be one time passwords to decrypt the data. In addition, a time-to-live (TTL) is integrated into the executable to provide an additional layer of security so that the data is only accessible within a defined time period.

Keywords: Data privacy, self destruction, time to leave (TTL), web services.

I. INTRODUCTION

With development of web services and popularization of Internet, web services are becoming more and more important for people's life. People requested to post their private and personal data on internet. People rely on internet service provider their privacy and security takes more risks. When data is being processed, transformed, stored by computer system or network, one of them copy or archive the data. This data is essential for system or network. The copy of data is stored in the system or at the network may leak the privacy of user. Internet service provider can use this data without user knowledge and may misuse this data.

Self destructive data scheme will self destruct the data which get stored at client side in the form of cookies. Additional to this data which is to be stored at server side will be stored in encrypted form. To decrypt the data one key will be generated which is valid for only one time. In addition Time to live (TTL) is provided which will increase security so that data is only accessible within a defined time period.

In this paper, Self Destruction (SeDas) present a solution to implement a self destructing data system, Which consist of two main parts: 1) secret key part-generate a pair of keys through RC4 algorithm. 2) survival time part-specify time limit to each keys. Through this it can meet the following advantages: 1) No explicit delete action by any third party. 2) The keys can be self destructed after user specified time and also reduces the communication overhead as well as network delay. 3)Increase processing speed and it will meet all the privacy preserving goals.

1.1 RC4 Algorithm

RC4 is ideal for storing information that is highly sensitive and highly important. A RC4 method can secure a secret over multiple servers and remain recoverable despite multiple server failure. The dealer may act as several district participants, distributing the shares among the participants. Each share may be stored on a different server, but the dealer can recover the secret even if several servers break down as long as they can recover. The algorithm divides a message into fix sized large blocks and encrypts these blocks concurrently on multi core machine.

1.2 Self Destruction using Time to Live property

Time to Live is a mechanism that limit the lifespan or lifetime of keys stored in cloud. TTL may be implemented as a counter or timestamp attached to or embedded in the keys. Once the prescribed event occur or time span has elapsed, keys can be self destructed without any user intervention. In computing application, TTL is used to improve performance of caching or to improve privacy from the leakages.

II. BACK GROUND

Web services allow user to take benefit from all the technologies, without the need for deep knowledge about or expertise with each one of them. It is not only sharing of resources but can also dynamically reallocating as per demands. User subjectively hope service providers will provide security policy to protect the data from the leakages. To protect these data is important also take more and more risk in web services environment. To provide privacy they perform encryption using key. On the other hand, When data is being processed, transformed and modified by the hackers and intruders. To overcome these problem sedas system provide more security in cloud environment through Shamir algorithm and also reduce the runtime overhead as well as network delay.

III. SYSTEM ARCHITECTURE

RC4 algorithm implemented in web server to generate a pair of keys to the user. The main components are key generation, encryption and decryption using encryption algorithm. The user can also specify the lifetime of each keys. These keys associated with Time to Live (TTL) property. When the keys can be self destructed after user specified time without any user intervention. The encrypt and decrypt procedure uses a algorithm for encryption and decryption. Algorithm that has been utilized in wide variety of security application, and is also commonly used to check data integrity. Keys are stored in server, before download a data in cloud the RC4 algorithm check the keys match with entered key. Then only the given user is authorized to download data in web.

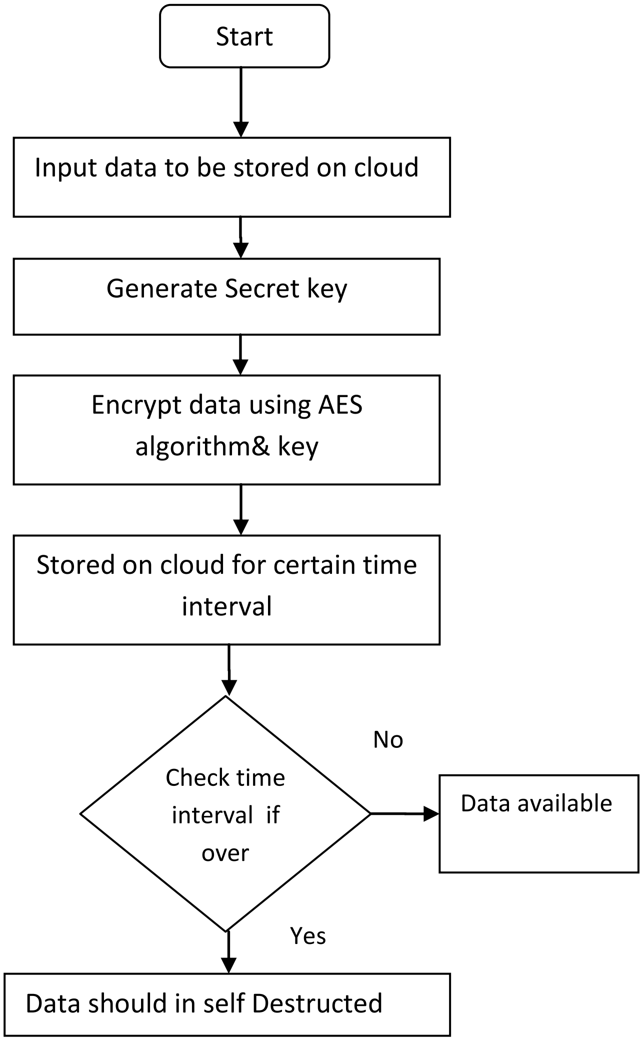

Start
Input data to be stored on cloud
Generate Secret key
Encrypt data using AES algorithm& key
Stored on cloud for certain time interval
Check time interval if over
No
Data available
Yes
Data should in self Destructed

Fig 1.System Model

In Proposed system, sedas method implement automatic self destruction of keys to avoid explicit delete or modification of data in cloud environment by any third party or attackers. It leads several advantages are reduces communication overhead, network delay, generate pair of keys to single file (keys length is 2bytes) , increase processing speed, increase I/O performance and also it will meets all privacy preserving goals.

IV. MODULES

User Authentication

Secret Key Part

File Uploading

Downloading File

Self Destruction Method

4.1 Modules Description

User Authentication:- A new user has to first create a profile. This is done by registration. A user id and password are submitted by the user. The user can login successfully only if user id and password are entered correctly. The login is a failure if the incorrect user id or wrong password is entered by the user. This helps in preventing unauthorized access.

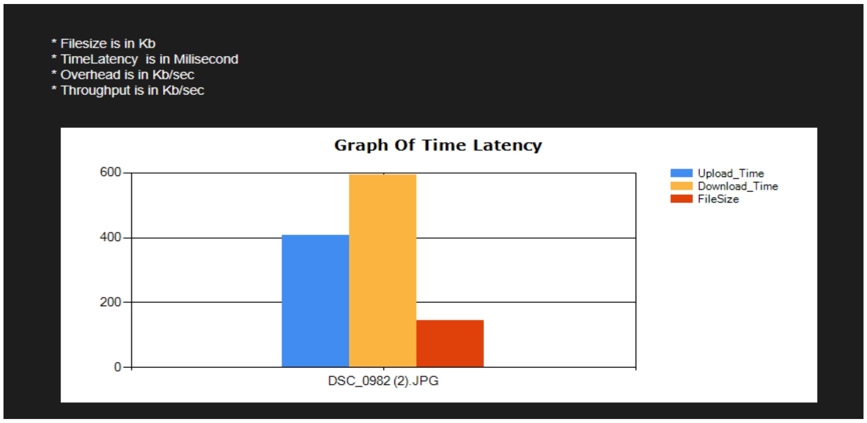
Secret Key Part:- RC4 is a stream cipher algorithm and operates on individual bits to secure the algorithm. Rc4 is the most common algorithm and is used in popular protocols like secure socket layer (SSL) to protect web browsing and in WEP to protect the wireless networks. The algorithm divides a message into fix sized large blocks and encrypts these blocks concurrently on multi core machine.

File Uploading:- Before upload file in cloud the user perform encryption using pair of keys generated by RC4. When a user upload only a encrypted file in cloud and stores his keys using sedas method, it should specify the file, keys and TTL as the arguments for uploading procedure. When the keys are self destructed after a user specified time.

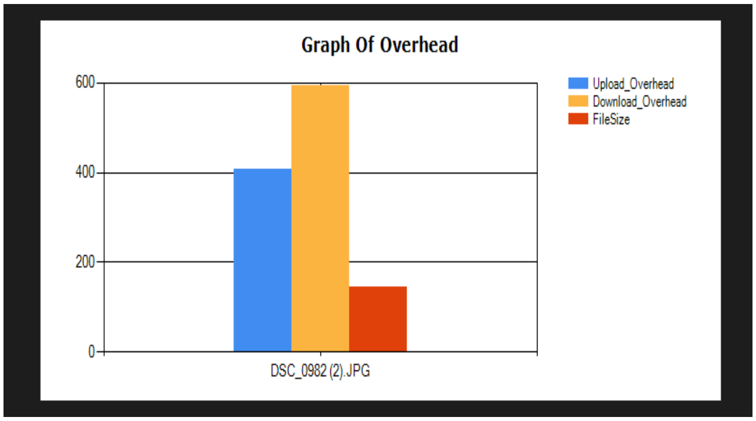
Downloading File:- Users who has permission can download data stored in the server . The data must be decrypted before use. If the self destruct operation has not triggered, the client can get enough key shares to reconstruct the keys successfully. During download process, Shamir algorithm checks the given keys are expired or not. If the keys are not expiring, the user can easily download. Otherwise, Shamir algorithm generates a new pair of keys to the authorized user.

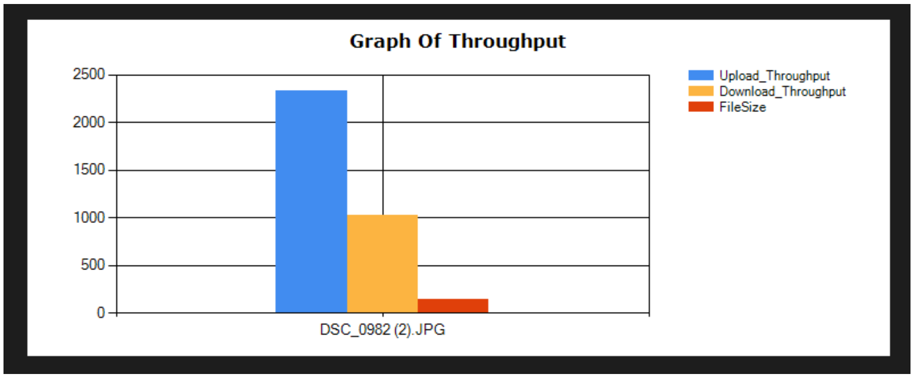
Self Destruction Method:- Self destruction mainly aims at protecting the user data privacy. All the keys become self destructed or unreadable after user specified time. The result demonstrate that the sedas meet all goals of privacy preserving. Sedas does not affect the normal use of storage system and can also meet the requirement of self destructing data under a survival time by user controllable keys. These are multiple storage services for a user to store data. Meanwhile, to avoid problem produced by the centralized “trusted” third party, the responsibility of sedas is to protect the user keys and provide the function of self destructing data.

V. There are multiple storage services for a user to store data. The user application program interacts with the SeDas server through SeDas’ client, getting data storage service. The process to store data has no change, but encryption is needed before uploading data and the decryption is needed after downloading data. In the process of encryption and decryption, the user application program interacts with SeDas. The client mainly runs in kernel mode, and it can mount a remote file system to local. 1st graph shows the latency. 2nd graph plots the overhead of both encryption and decryption processes under different file sizes in SeDas. 3rd graph shows the throughput results for the different schemes. The throughput decreases because upload/download processes require much more CPU computation and finishing encryption/ decryption processes in the SeDas system,



* Filesize is in Kb
* TimeLatency is in Milisecond
* Overhead is in Kb/sec
* Throughput is in Kb/sec





VI. CONCLUSION

Each authenticated user, the RC4 generate a pair of keys also the user specify the lifetime of each keys. After user specified time the keys can be self destructed without user intervention. During the download process the RC4 check the validity of the keys. If the keys are expired, The RC4 generate new pair of keys to the user through this only the sedas system meet all the privacy preserving goals. The future work of the sedas system further to increase the key length to provide user data privacy in web infrastructure

REFERENCES

[1] Y. Tang, P. P. C. Lee, J. C. S. Lui, and R. Perlman, “FADE: Secure overlay cloud storage with file assured deletion,” in Proc. Secure Comm, 2010**.**

[2] C. Wang, Q. Wang, K. Ren, and W. Lou, “Privacy-preserving public auditing for storage security in cloud computing,” in Proc. IEEE INFOCOM, 2010.

[3] R. Geambasu, T. Kohno, A. Levy, and H. M. Levy, “Vanish: Increasing data privacy with self-destructing data,” in *Proc.* USENIX Security Symp., Montreal, Canada, Aug. 2009, pp. 299–315.

[4] Z. Lu, T. Li, X. Hu, K Zhao, J. Zeng, L. Peng. Data self-destruction method. In Application Research of Computers, 26(1), 2009, pages 350-355.

[5] G. Wang, F. Yue, and Q. Liu, “A secure self-destructing scheme for electronic data,” J. Comput. Syst. Sci., vol. 79, no. 2, pp. 279–290, 2013.

[8] J. Xiong, Z. Yao, J. Ma, X. Liu, and Q. Li, “A secure document self-destruction scheme with identity based encryption,” in Proceedings of the 5th International Conference on Intelligent Networking and Collaborative Systems, INCoS 2013. IEEE, 2013, pp. 239–243.

[9] S. Wolchok, O. S. Hofmann, N. Heninger, E. W. Felten, J. A. Halderman, C. J. Rossbach, B. Waters, and E. Witchel, “Defeating vanish with low-cost sybil attacks against large DHEs,” in Proc. Network and Distributed System Security Symp., 2010.

[10] R. Perlman, “File system design with assured delete,” in Proc. Third IEEE Int. Security Storage Workshop (SISW), 2005.

[11] Disha Handa, Bhanu Kapoor PARC4: High Performance Implementation of RC4 Cryptographic Algorithm using ParallelismICROIT 2014, India.