Indian Journal of Science and Technology, *Vol 9(48), DOI: 10.17485/ijst/2016/v9i48/103399, December 2016*
ISSN (Print) : 0974-6846
ISSN (Online) : 0974-5645

A Proficient Two Level Security Contrivances for Storing Data in Cloud

K. Anbazhagan[1]* and R. Sugumar[2]

[1]Department of IT, St. Peter's University, Chennai - 600082, Tamil Nadu, India; rishianbu@gmail.com
[2]Department of CSE, Velammal Institute of Technology, Chennai - 601204, Tamil Nadu, India

Abstract

Objectives: Cloud Computing provides vast storage facility. The requirement of this system is to improve the security and transmission performance in the cloud storage environment. **Methods**: This system provides two level of security for the cloud data. The Client Data Security Contrivance (CDSC) and Cloud Service Provider (CSP) Data Security Contrivance are the two methods which transforms the original data to cipher text. The security algorithm used in CDSC is Linguistic Steganography. Blowfish algorithm is used in CSP Data Security Contrivance to convert the cipher text using a secret-key block cipher of 64-bits. This system is designed to increase security and to improve the transmission performance. Storage-cheating attack model and Privacy-cheating attack model are implemented to protect the data on cloud. **Findings**: This system increases security and improves transmission performance. The blowfish algorithm is secure against unauthorized attacks and runs faster than the popular existing algorithms. Storage-cheating attack model and Privacy-cheating attack model provides the data confidentiality for data stored in the cloud in higher way. **Improvements:** The system can be implemented in real cloud system in the future.

Keywords: Blowfish Algorithm, Cloud Computing, Data Storage in Cloud, Linguistic Steganography, Two Levels Security

1. Introduction

One of the good service offered by cloud computing is SaaS - Storage as a Service. It provides good storage infrastructure to small or middle sized business. Hence the small or middle sized business, need not to invest a lot of capital for their storing their data and technical personnel for maintaining the infrastructure. SaaS is also providing way for all businesses to ease recovery of data from disaster. SaaS providers are aiming secondary storage applications to manage backups of their customer data. The advantage of using the SaaS is the customer can save their huge investment for storage of data and maintaining the storage equipments. The cloud user and the cloud service provider have to sign in the Service Level Agreement (SLA) for their transactions. It contains all terms and conditions with all required details like storage size allocated and cost per unit. The basic requirement for utilizing this facility is trustful connection.

Blowfish encryption algorithm was designed and released by Bruce Schneier in 1993. It was accepted as a strong encryption algorithm. We need not any license to utilize this algorithm. It is well efficient in the area of hardware implementation. Table lookup, XOR and addition are the elementary operators of Blowfish algorithm. Contains of the table is a P-array and 4 S-boxes. In this system the cipher based on Feistel rounds. The F-functions are designed to provide the security with high speed and efficiency in software. It encrypts data on 32-bit microprocessors.

Steganography is one of the good techniques used in security system to encrypt data. There are many type of steganography techniques which uses different types of wrap medium to hide data. It is not easy to detect hidden information by an observer. The steganography technique manipulates properties or the medium to hide data in it. Some of the steganography systems are Linguistic Steganography, Image Steganography and Watermarking.

*Author for correspondence

Linguistic Steganography is a security technique where linguistic properties of a text are modified to hide information. All languages have their own properties. When even small changes made to a text may result in anomalous changes in the document level. Hence finding the original text from the cipher text made by the linguistic steganography is a challenging problem for outside observer.

2. Literature Review

In Steganography Using Many Base Notational System and Human Vision Sensitivity system[1] the author proposed a security system using human vision sensitivity to hide secret bits. To execute this system, first the secret data are converted into a sequence of code to embed in a notation system with many bases. In this case, the particular bases used are firm by the degree of local variation of the pixel magnitudes in the host image. An amendment is done in the Least Significant Bit Matching (LSBM). Jarno[2] have introduced new technique using LSB. This amendment makes available the required choice of a binary function of two cover pixels. In "Visual Cryptographic Steganography in images"[3] has joined the data encoding and hiding process to increase the security level. This system is efficient to prevail over the difficulty of image color modifies after the embedding process. DWT-based color Images Steganography Scheme[4] was proposed to hide data in various dimensions like vertical, diagonal and horizontal components of the sub image. Syntactical Steganography exploit the syntactic structures of a text. The famous algorithm of Context Free Grammers (CFG) based Mimicry[5] developed by Peter Wayner comes under this category.

3. Framework

In (Figure 1) The system architecture depicts the overview of the proposed security system for the data stored in the cloud. In this system there are number of Cloud Data Servers (CDS), CDS1; CDS2; CDS3;….;CDS(N). All the cloud data servers are controlled and maintained by Cloud Service Providers (CSP). Cloud data servers facilitate the cloud users to store their data in the cloud. To avail the resources, the cloud users had to contact the CSP. The Cloud Service Provider allocates the required resource to the cloud users by initiating a Service Level Agreement. The Cloud User (CU) can be companies, institutions etc., which stores the confidential data on cloud. Client Data Security Contrivance (CDSC) provides a security for the client data.

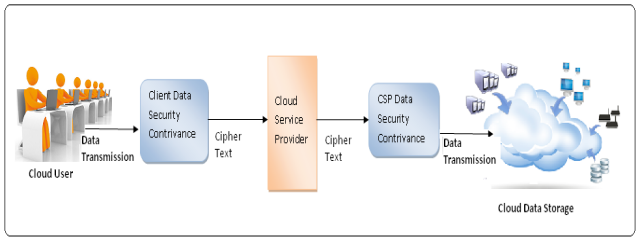

Figure 1. System architecture - a proficient two level security contrivances for data stored in cloud.

DSCC encrypts the data received from the cloud user using linguistic steganography. Cloud Service Provider's Data Security Contrivance at CSP (CSPDSC) encrypts the data received from the DSC-C using blowfish algorithm.

3.1 Attacks on Cloud Data

3.1.1 Storage-cheating Attack Model

The attacks en routes to insecurity for storing data in the cloud, for a case, the antagonist would illogically alter the stored data to compromise the data integrity (malicious case) or reveal the important data to purchase interest (interest-purchasing case). In the malicious case, the conciliated cloud servers would reply to the cloud user storage enquiry with a random number. Cloud users faces difficult due to lack of the corporal possession of the huge size of outsourced data. We take for granted that if the demanded data set is D, the direct returned data set is D' and the unacceptable returned data set is D–D'.

3.1.2 Privacy-cheating Attack Model

Another kind of attacks en route to the cloud data, is privacy-cheating attack. This can anglicised as one more type of storage-cheating attack, we believed that the antagonist may compromise cloud users privacy by disclose their important data to others, e.g. strong condition to public or auction worth to business competitors and this may lead to serious consequences. To provide data confidentiality, one straightforward approach is to save the cipher text in the cloud servers. Such an approach may avoid the regular cloud working out from being further processed. If the data are stored in as actual text in the cloud servers, the antagonist may sell/publish the sensitive data to the opponent. Furthermore, we believe that to sell the sensitive data, the antagonist should provide the equivalent essential proofs to disclose the authenticity of the stored data and computing results to persuade others.

3.2 Design Goals

- Security for Data storage: To build a protocol which will provide security for data stored in cloud, the planned protocol should facilitate that CU to ensure the originality of data.
- Privacy cheating dissuasion: The planned scheme must guarantee that only designated verification parties (e.g., CU) can authenticate the stored data, this will discourage the CSP from compromising the cloud users solitude, even if the attackers compromises the cloud server.
- Efficiency: To reduce the overhead on transmission of the encrypted data.

3.3 Stakeholders

The stakeholders are mentioned and described briefly here are took part in the planned process.

- Cloud Users (CU) – The entity which wish to store their data on cloud.
- Cloud Service Providers (CSP) – The company which offers large scale of storage, computation etc. resources for the
- Client Data Security Contrivance (CDSC) – This system receives data from the CU and encrypts the data with a cover text using Linguistic Steganography.
- CSP Data Security Contrivance (CSPDSC) – This system receives cipher text from the CU and encrypts it with the blowfish security algorithm.
- Linguistic Steganography - Steganography is the art of transmitting and hiding data through apparently safe transporter in an effort to cover up the survival of the data, the word Steganography accurately means covered or hiding writing as derived from Greek. Steganography has its own situate in security. It is not proposed to replace cryptography but supplement it. Encrypting a message with Steganography methods decreases the chance of a message being detected.
- Blowfish - Blowfish encryption algorithm was designed and released by Bruce Schneier in 1993. It was accepted as a strong encryption algorithm. We need not any license to utilize this algorithm. It is well efficient in the area of hardware implementation. Table lookup, XOR and addition are the elementary operators of Blowfish algorithm. The table contains four S-boxes and a P-array. It is a cipher based on Feistel rounds. The design of F-functions provides the safety with superior speed and efficiency in software. It encrypts data on 32-bit microprocessors.

4. System Initialization

4.1 System Initialization Step

The System Initialization step concerned with selection of Cloud Service Provider. After selecting the CSP, the client or CU will initiate the service level agreement with the CSP which contains all the requirements of CU and terms conditions of both the parties.

4.2 User Registration Step

First step of the cloud user who requires the cloud services has to register their requirement and identification with CCSC in order to send and receive the data.

4.3 Our Security Contrivances for Cloud Storage Protocol Include the Following Steps

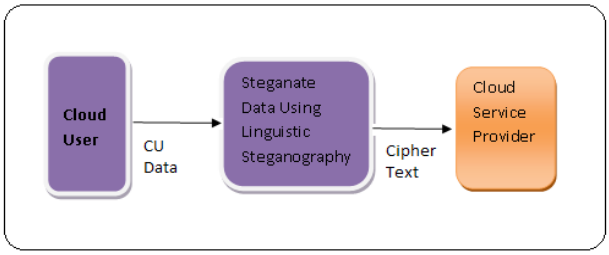

Figure 2. Client data security contrivance.

1. Receiving data from cloud user
2. Steganate the data using linguistic steganography
3. Transmitting the steganated data to CSP

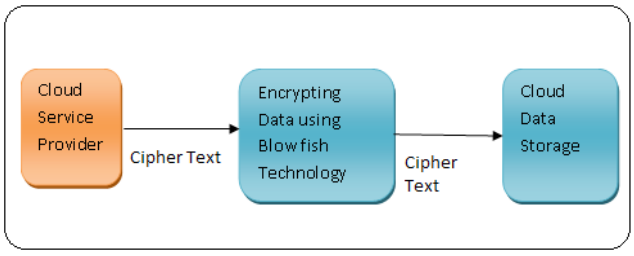

Figure 3. CSP data security contrivance.

1. Receives cipher text from CDSC
2. Encrypts the cipher text with blowfish algorithm
3. Stores the cipher text in the cloud server.

5. Experimental Analysis

This is a short description about the established Distributed File System (DFS) in cloud and Hadoop DFS of Google. Then the introduction to the developed test bed: Cloud Security Contrivance which incorporated with the security module to HDFS. A short introduction to established DFS system such as HDFS, Yahoo, Google, IBM are using cloud to store and handle their large amount of data. The DFS in cloud master consist of three modules they are Interface, Storage and Management. Interface Module facilitates the client by providing web browser front and APIs which suits for their platform and OS. The task of Storage Module is to maintain the Virtual File System organization and divide files into blocks and uploading them to selected slave servers. The management module governs the storage resources and monitors the storage space and transmission speed.

5.1 Experimental Environment

To provide more security features in the current system Data Security Contrivance module is included. This module also supports encrypted channel for data upload. Our experimental lab consists of 4 computers with Intel Core processor i5-4690k at the speed of 3.5 GHz with 8 GB RAM memory. One among the four computers acts the role of the master server in the cloud computing, which assigns storage space capacity and storage data index for the remaining three slave servers. The Cloud users can give their storage requirement through a wired or wireless communication. When the file size large, it can be sliced into several blocks. We defined the block size as 64 MB, where 60 MB is used to for file contents and rest 4 MB is used for security head. The security head consists of symmetric key parameters. Now, the data can be uploaded by cloud users to the cloud servers with cryptographic techniques. The Master server verifies the blocks and stores the data in cloud storage servers.

6. Experiment Results

We started our experiment by examining the system efficiency under various traffic loads. The initial load is started at 100 MB and steadily increasing to 800 MB. The traffic load overhead can be analysed with the impact of security overhead.

6.1 First Traffic Load's Impact

Figure 4 shows the speed of the system performance and it was measured when uploading the files. Uploading speeds can be estimated with security in Data Security Contrivance. The range of the uploading speeds is 15-17 Mb/s, which are almost 2-3 Mb/s lower than the original protocols.

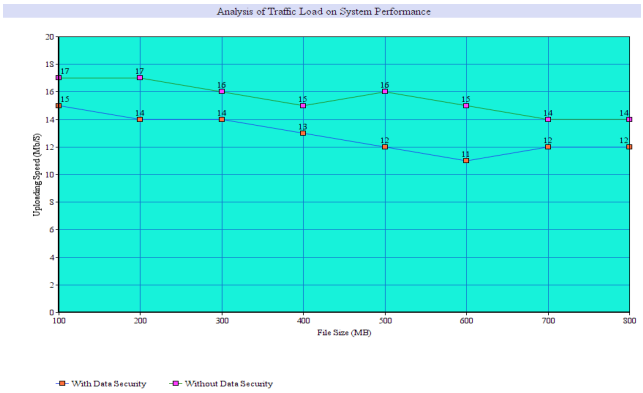

Figure 4. Analysis of traffic load on system performance.

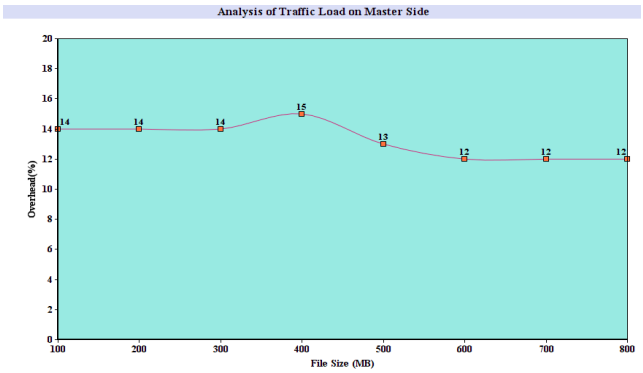

Figure 5. Traffic analysis on client side.

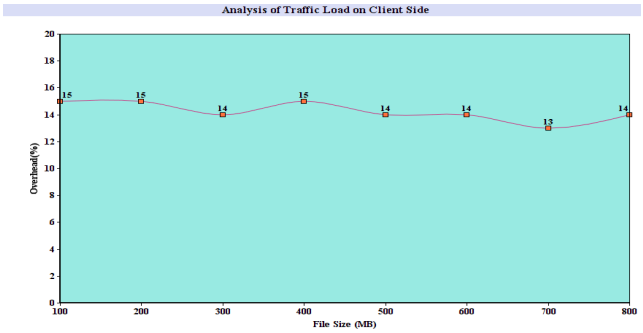

Figure 6. Traffic analysis on master side.

6.2 Second Traffic Load's Impact

The (Figures 3 and 4) shows the security overheads of the cloud user side and master side respectively. Security overhead escort to the system delay in the client side and in the master side.

The percent level of security overhead = the time of security operations / the time of file transmissions. The curve become stable after a period of time, which shows the security overhead are not more than 12%. It was also found that when large files are uploaded the security overhead would become more stable that is the transmission time is high and session establishment time is less.

6.3 Overall Overhead Comparison

To find the efficiency of the Data Security Contrivance, the total time of uploading files are estimated in two cases. First one is the original protocol and the next one is DSC. The result of the observation is nearly 15% in the best case and 29% in the worst case.

In summary, the experiment results conclude that Data Security Contrivance is indeed a feasible solution for secure data storage in the cloud computing.

7. Conclusion

In this paper, we have proposed, Data Security Contrivance to protect data from a privacy cheating attack for data stored in the cloud. To increase the efficiency, different users requests can be simultaneously submitted to DSC. By the extensive security analysis and performance simulation in our developed Data Security Contrivance proves that the protocol is effective and efficient for attaining a secure cloud computing. In our future work, we persist to consider some detailed security issues in the above security model. Furthermore, we plan to execute it in the genuine cloud platform such as Amazon and versa stack.

8. References

1. Armbrust M, Fox A, Griffith R. A view of cloud computing, Communications of the ACM. 2010 Apr; 53(4):50–8.
2. Ateniese G, Burns R, Curtmola R Herring J. Provable data possession at untrusted stores. Proceedings of the 14th ACM Conference on Computer and Communications Security (CCS'07), USA; 2007. p. 213–21.
3. Ateniese G, Di Pietro R, Mancini L, Tsudik G. Scalable and efficient provable data possession. Proceedings of the 4th International Conference on Security and Privacy in Communication Networks, Istanbul, Turkey; 2008. p. 742–9.
4. Muthuramaligam S, Nachiar TS. Enhancing the security for manet by identifying untrusted nodes using uncertainity rules. Indian Journal of Science and Technology. 2016 Jan; 9(4). DOI: 10.17485/ijst/2016/v9i4/87043.
5. Lakshmipriya B, Sri RL, Balaji N. A novel approach for performance and security enhancement during live migration. Indian Journal of Science and Technology. 2016 Jan; 9(4). DOI: 10.17485/ijst/2016/v9i4/87031.
6. Shastri S, Thanikaiselvan V. PVO based reversible data hiding with improved embedding capacity and security. Indian Journal of Science and Technology. 2016 Jan; 9(4). DOI: 10.17485/ijst/2016/v9i4/87191.
7. Boneh D, Lynn B, Shacham H, Short signatures from the Weil pairing. Journal of Cryptology. 2004 Apr; 17(4):297–319.
8. Dean J, Ghemawat S, MapReduce: Simplified data processing on large cluster. Communications of The ACM. 2008; 51(1):107–13.
9. Du JJ, Mangal M, Murugesan M. Uncheatable grid computing. Proceedings of the 24th International Conference on Distributed Computing Systems (ICDCS'04), Japan; 2004. p. 866-74.
10. Huang Q, Yang G, Wong DS, Susilo W. Efficient strong designated Verifier signature schemes without random oracle or with non-delegatability. International Journal of Information Security. 2011 May; 10(6):373–85.