International Journal of Emerging Technology in Computer Science & Electronics (IJETCSE)
ISSN: 0976-1353 Volume 26 Issue 5 – AUGUST 2019.

A Three-Layer Privacy Preserving Cloud Storage Scheme Based on Computational Intelligence in Fog Computing

Y. KRISHNAVENI [#1] and H. PARVEEN BEGUM [*2]

[#] *M. Phil. Scholar, Dept. of Computer Science, PRIST University, Madurai, India*

[*] *Asst. Prof. Dept. of Computer Science, PRIST University, Thanjavur, India*

***Abstract*— Cloud computing technology developed in multiples now-a-days. This results in the application development for the cloud technology. The applications that run in the cloud also multiplied in numbers. With the volatile growth of unstructured data, cloud storage technology gets more attention and better development. However, in current storage schema, user's data is totally maintained in cloud servers. In other words, users mislay their right of control on data and face privacy leakage risk. Privacy protection schemes are depends on encryption technology, but these kinds of methods cannot effectively resist attack from the inside of cloud server. In order to solve this problem, we present a three-layer storage framework based on fog computing. The proposed framework can both take full advantage of cloud storage and protect the privacy of data. Then, we can store a small part of data in local machine and fog server in order to protect the privacy. Moreover, depend on computational intelligence, this algorithm can calculate the distribution proportion stored in cloud, fog, and local machine, respectively. The theoretical safety analysis and experimental assessment, the feasibility of our scheme has been validated, which is really a powerful supplement to existing cloud storage scheme.**

Index Terms*— Cloud computing, cloud storage, fog computing, privacy protection.

I. INTRODUCTION

Recent years the development of cloud computing technology. With the volatile growth of unstructured data, cloud storage technology gets more attention and better development. However, in current storage schema, user's data is totally maintained in cloud servers. In other words, users mislay their right of control on data and face privacy leakage risk. Privacy protection schemes are commonly depends on encryption technology, but these kinds of methods cannot effectively resist attack from the inside of cloud server. In order to solve this problem, we present a three-layer storage framework based on fog computing. The proposed framework can both take full advantage of cloud storage and protect the privacy of data. Besides, Hash-Solomon code algorithm is designed to split data into different parts. Then, we can store a small part of data in local machine and fog server in order to protect the privacy. Moreover, it depends on computational intelligence, this algorithm can compute the distribution proportion stored in cloud, fog, and local machine, respectively. The main analysis and experimental assessment, the feasibility of our scheme has been validated, which is really a powerful supplement to existing cloud storage scheme. This framework makes full use of fog server's storage and data processing capability. This framework includes three layers, the cloud server, the fog server and the local machine. Each server saves a certain part of data, the storage proportion is determined by users' allocation strategy. Firstly, user's data will be encoded on user's local machine. Then, for example, let 1% encoded data be stored in the machine. Then upload the remainder 99% data to the fog server. Secondly, on the fog server, it do similar operations to the data which comes from user's machine. There will be about 4% data stored in the fog server and then upload the remainder data to the cloud server. The above operations are based on Hash-Solomon code. Hash-Solomon code is a kind of coding methods based on Reed Solomon code. After being encoded by Hash-Solomon code, the data will be divided into k parts and generates m redundant data. Hash-Solomon code has such property, in these k+m parts of data, if someone has at least k parts, he can recover the complete data.

II. LITERATURE REVIEW

A. T. Wanget al., "Reliable wireless connections for fast-moving rail users based on a chained fog structure,"Inf. Sci., vol. 379, pp. 160–176, 2017.

Currently, 3G and 4G networks give customers with high-speed wireless services almost everywhere. However, the wireless connection is generally unstable and unreliable, especially for fast-moving end users (e.g., those on trains and buses). To explore the severity of this problem, we conducted real experiments on fast-moving trains to investigate the quality of 3G connections. 1) From the outlook, the 3G connections were not steady and suffered from frequent disruptions of connectivity, and 2) from the spatial perspective, the connections that were introduced in different train compartments were largely independent. These two details are helped us to propose a brand-new fog computing structure, which acts as an intermediate layer between the end users and the 3G infrastructure. This new fog structure creates a series of mutually chained network gateways that are located in different compartments. This formation creates the aforementioned problem of unstable connectivity and

thus ensures reliable wireless service for fast-moving users, such as passengers on trains. We performed a series of theoretical and empirical analyses to evaluate the performance of the newly proposed structure. All of the experimental results are introduced that our proposed fog structure greatly improves the reliability of wireless connections on fast-moving trains.

B. J. Zeng, T. Wang, Y. Lai, J. Liang, and H. Chen, "Data delivery from WSNs to cloud based on a fog structure," inProc. Int. Conf. Adv. Cloud Big Data, 2016, pp. 104–109.

Recent years, with the emerging technology of cloud computing, the powerful computing and storage capability of cloud computing injects new vitality into wireless sensor networks (WSNs) and motivates a series of new applications. However, the data delivery from WSNs to Cloud it creates a bottleneck for the reason of the poor communication ability of WSNs, especially for delay-sensitive applications, which limits their further development and applications. To address this problem, we propose a fog structure which composes of multiple mobile sinks. Mobile sinks act as fog nodes to bridge the gap between WSNs and Cloud. They cooperate with each other to set up a multi-input multi-output (MIMO) network, aiming at maximizing the throughput and minimizing the transmission latency. The problem is proved to be NP-hard and we design an approximation algorithm to solve this problem with several provable properties. We compare our method to several traditional solutions. Extensive experimental results suggest that the proposed method significantly outperforms the traditional solutions.

C. J. Shen, D. Liu, J. Shen, Q. Liu, and X. Sun, "A secure cloud-assisted urban data sharing framework for ubiquitous-cities,"Pervasive MobileComput., vol. 41, pp. 219–230, 2017.

With the accelerated process of urbanization, more and more people tend to live in cities. In order to deal with the big data that are generated by citizens and public city departments, new information and communication technologies are utilized to process the urban data, which makes it more easier to manage. Cloud computing is a novel computation technology. After cloud computing was commercialized, there have been lot of cloud-based applications. For that reason the cloud service is provided by the third party, the cloud is semi-trusted. Due to the features of cloud computing, there are many security issues in cloud computing. Attribute-based encryption (ABE) is a encouraging cryptography technique which can be used in the cloud to solve many security issues. In this paper, we present a framework for urban data sharing by exploiting the attribute-based cryptography. In order to stable the real world ubiquitous-cities utilization, we extend our scheme to support dynamic operations. In specific, from the part of performance analysis, it can be included that our scheme is secure and can resist possible attacks. Furthermore, experimental results and comparisons show that our scheme is more efficient in terms of computation.

D. Z. Fu, F. Huang, K. Ren, J. Weng, and C. Wang, "Privacy-preserving smart semantic search based on conceptual graphs over encrypted outsourced data,"IEEE Trans. Inf. Forensics Security, vol. 12, no. 8, pp. 1874–1884, Aug. 2017

Searchable encryption is a main research area in cloud computing. Therefore, in this paper, we present a content-aware search scheme, which can make semantic search more smart. First, we create conceptual graphs (CGs) as a knowledge representation tool. Then, we present our two schemes (PRSCG and PRSCG-TF) based on CGs according to different scenarios. In order to conduct numerical calculation, we transfer original CGs into their linear form with some modification and map them to numerical vectors. Finally, we choose a real-world data set: CNN data set to test our scheme. We also examine the privacy and efficiency of proposed schemes in detail. The experiment results show that our introduced schemes are efficient.

E. Z. Fu, K. Ren, J. Shu, X. Sun, and F. Huang, "Enabling personalized search over encrypted outsourced data with efficiency improvement," IEEE Trans. Parallel Distrib. Syst., vol. 27, no. 9, pp. 2546–2559, Sep.2016.

In cloud computing, searchable encryption scheme over outsourced data is a main research field. However, most old works on encrypted search over outsourced cloud data follow the model of "one size fits all" and ignore personalized search intention. Furthermore, most of them support only exact keyword search, which greatly affects data usability and user experience. So how to structure a searchable encryption scheme that supports personalized search and improves user search experience remains a very challenging task. In this paper, we identify and solve the difficulty of personalized multi-keyword ranked search over encrypted data (PRSE) while preserving privacy in cloud computing. Based on WordNet, we build a user interest model for individual user by analyzing the user's search history, and adopt a scoring mechanism to express user interest smartly. To find the limitations of the model of "one size fit all" and keyword exact search, we propose two PRSE schemes for different search intentions. Large experiments on real-world dataset validate our analysis and show that our proposed solution is very efficient and effective.

III. PROPOSED WORK

A. Diffie-Hellman Key Agreement protocol

The second module includes a secure communication using Diffie – Hellman protocol for mutual agreement process. Here we want to compute key-parameters to generate a shared secret key. An email or SMS is sent to the client, contains a name, IP, amount of data downloaded at the time of data access. Authentication is done based on IP, user search behavior, amount of data, task division. Verification code generation and its comparison need to done to perform mutual authentication.

B. Encryption and Decryption

Multimedia medical data encryption and decryption can be done using Blowfish algorithm, which is better than other encryption algorithms such as AES, DES, and 3DES in terms of its key length, block size, confidentiality, number of rounds, memory usage and computation time.

Blowfish is designed in 1993 by Bruce Schneier and

included in a large number of cipher suites and encryption products. Blowfish gives a good encryption rate in software and no effective cryptanalysis of it has been found to date. However, the Advanced Encryption Standard (AES) now receives more attention, and Schneier recommends Two fish for modern applications.

Schneier designed Blowfish as a general-purpose algorithm, intended as an alternative to the aging DES and free of the problems and constraints associated with other algorithms. At the time Blowfish was released, many other designs were proprietary, encumbered by patents or were commercial or government secrets. Schneier has stated that, "Blowfish is unpermitted, and will remain so in all countries. The algorithm is hereby placed in the public domain, and can be freely used by anyone. Notable features of the structure include key-dependent S-boxes and a highly complex key schedule.

1) Data Encryption

The algorithm consists of 16 rounds. A key-dependent permutation and a key and data-dependent substitution are carried out in each round during encryption and decryption. All working operations are XORs and additions on 32-bit words. The steps are shown in algorithm.

1. Divide 64-bit plaintext into two 32-bit halves: file1, file2
2. For i = 1 to 16 do steps 3 to 5
3. file1 = file1 XOR Pi
4. file2 = F(file1) XOR file2
5. Swap file1 and file2
6. Swap file1 and file2 to undo last swapping.
7. file2 = file2 XOR P17
8. file1 =file1 XOR P18
9. Concatenate file1 and file

2 The function F is as follows:

1. Split file1 into four eight-bit quarters: a, b, c, and d
2. F(file1) = ((S_1,a + S_2, b mod 2^{32}) XOR S_3,c) + S_4, d mod 2^{32}

2) Data Decryption

The matched files are retained from the cloud server are sent to the authorized data user. The files are in ciphertext form. The Blowfish decryption algorithm is used here to decrypt the file and give the original result. The encryption procedure has used for decryption. However, the input of the sub-keys P1, P2,…, P18 are applied in reverse order.

IV. ALGORITHM

A. AES

The Advanced Encryption Standard, or AES, is a symmetric block cipher selected by the U.S. government to be classified information and is developed in software and hardware throughout the world to encrypt sensitive data.

The National Institute of Standards and Technology (NIST) started development of AES in 1997 when it announced the need for a successor algorithm for the Data Encryption Standard (DES), which was starting to become vulnerable to brute-force attacks.

This new, advanced encryption algorithm would be unarranged and had to be "capable of protecting sensitive government information well into the next century," according to the NIST announcement of the procedure for development of an advanced encryption standard algorithm. It was intended to be simple to implement in hardware and software, as well as in restricted environments (for example, in a smart card) and offer good defenses against various attack techniques.

AES features

The procedure for this new symmetric key algorithm was fully open to public scrutiny and comment; this ensured a thorough, transparent analysis of the designs submitted.

NIST specified the new advanced encryption standard algorithm must be a block cipher capable of handling 128 bit blocks, using keys sized at 128, 192, and 256 bits; other criteria for being selection as the next advanced encryption standard algorithm included:

Security: Competing algorithms were to be judged on their ability to resist attack, as compared to other submitted ciphers, though security strength was to be considered the most important factor in the competition.

Cost: Intended to be released under a global, nonexclusive and royalty-free basis, the candidate algorithms were to be evaluated on computational and memory efficiency.

Algorithm and development characteristics to be evaluated included the flexibility of the algorithm; suitability of the algorithm to be implemented in hardware or software; and overall, relative simplicity of implementation.

Choosing AES algorithms

Fifteen competing symmetric key algorithm designs were analysed to preliminary analysis by the world cryptographic community, including the National Security Agency (NSA). In August 1999, NIST chosen five algorithms for more extensive analysis. These were:

MARS, presented by a large team from IBM Research RC6, presented by RSA Security Rijndael, presented by two Belgian cryptographers, Joan Daemen and Vincent RijmenSerpent, presented by Ross Anderson, Eli Biham and Lars KnudsenTwofish, presented by a large team of researchers from Counterpane Internet Security, including noted cryptographer Bruce Schneierkey and algorithm setup time; and resistance to several attacks, both in hardware- and software-centric systems. Members of the global cryptographic community managed detailed analyses (including some teams that tried to break their own submissions).

After much feedback, debate and analysis, the Rijndael cipher -- a mash of the Belgian creators' last names Daemen and Rijmen -- was selected as the proposed algorithm for AES in October 2000 and published by NIST as U.S. FIPS PUB 197. The Advanced Encryption Standard became successful as a federal government standard in 2002. It is also included in the International Organization for Standardization (ISO)/International Electrotechnical Commission (IEC) 18033-3 standard, which specifies block ciphers for the purpose of data confidentiality.

1) ADVANTAGES

➨As it is implemented in both hardware and software, it is most robust security protocol.

➨It utilize higher length key sizes such as 128, 192 and 256 bits for encryption. Therefore it makes AES algorithm more robust against hacking.

➨It is effective security protocol used for wide various of applications such as wireless communication, financial transactions, e-business, encrypted data storage etc.

➨It is one of the spread commercial and open source solutions used all over the world.

➨No one can hack your personal information.

➨For 128 bit, about 2128 attempts are required to break. This makes it very complex to hack it as a result it is very safe protocol.

2) DISADVANTAGES

➨It uses too simple algebraic structure.

➨Every block is regularly encrypted in the same way.

➨Hard to implement with software.

➨ AES in counter mode is difficult to implement in software taking both performance and security into considerations.

B. BCH Algorithm

The Bose, Chaudhuri, and Hocquenghem (BCH) codes creates a large class of powerful random error-correcting cyclic codes. This class of codes is a remarkable formulation of the Hamming codes for multiple-error correction. Binary BCH codes were discovered by Hocquenghem in 1959 and independently by Bose and Chaudhuri in 1960. Formulation of the binary BCH codes to codes in pm symbols (where p is a prime) was obtained by Gorenstein and Zierler. Among the nonbinary BCH codes, the most important subclass is the class of Reed-Solomon (RS) codes. Between all the decoding algorithms for BCH codes, Berlekamp's iterative algorithm, and Chien's search algorithm are the most efficient ones.

1) ADVANTAGES

1. The principal advantage is the ease with which they can be decoded using 'syndrome decoding' method.
2. Authorize very easy electronic hardware to perform the task, obviating the need for a computer, and meaning that a decoding device may be made small and low-powered.
3. Highly flexible, permitting control over block length and acceptable error thresholds, meaning that a custom code can be designed to a given specification.
4. Reed-Solomon codes are BCH codes, are used in applications such as satellite communications, compact disc players, DVDs, disk drives, and two-dimensional bar codes.
5. BCH codes are also useful in theoretical computer science.
6. Low amount of redundancy.
7. Easy to implement in hardware.

2) DISADVANTAGES

Complexity

Iterative and complex decoding algorithm

Decoder cannot decide whether a decoded package is false or not.

C. HMAC Algorithm

A hash-based message authentication code (HMAC) algorithm is used to prove integrity and authentication of the message of b-bits length. Here a hash function and a secret key are used to check whether the uploaded data altered or not. Different parties will hash the message again themselves with the secret key. The received and computed hashes will match if it is authentic. The key used is a shared key between parties. We have to pad zeros on the left side of the secret key until it becomes b-bits. HMAC algorithm uses two passes of computations to create HMAC code. During first pass we XOR padded secret key with the i_pad. The output obtained in the above step is appended with the plain text then apply a secure hash algorithm (SHA-512) which produces n-bits output. Then during second pass we XOR padded secret key with the o_pad. The output obtained is appended with the output of the first pass then apply SHA-512.

1) ADVANTAGES

The authors state that it is possible that some attacks may work against HMAC but fail against NMAC. Also, it is stated that having a single l-bit long key instead of two randomly chosen key does not compromise on the security. Overall, NMAC is a faster scheme to implement. On the other hand, HMAC requires only one l-bit long key, as opposed to two keys as in NMAC and hence simpler computations.

2) DISADVANTAGES

The underlying hash must be modified to key the Initial Variable (IV) which is not too difficult in software.

The HMAC function is slower than the NMAC function as it requires two more computation of the compression function.

If the length of key is less than l-bits, the strength of the keyed IV is reduced. • A periodic refreshment of keys is required.

It is inconsistent as there are some attacks that work against HMAC but fail against NMAC.

V. CONCLUSION

Working with computer provides us a pleasant experience and delivers the expected results excellently. This thesis "Fog Computing" is to mark the user to have the basic knowledge of computer. This thesis is of great importance and it is widely used countrywide. It helps people to purchase their home appliances through online. This thesis is done in the computerized form. It is a fast process and therefore saves time and money.

REFERENCES

[1] N. Aaraj, S. Ravi, S. Raghunathan, and N. K. Jha, "Architectures for efficient face authentication in embedded systems," inProc. Design, Autom. Test Eur., Mar. 2006, vol. 2, pp. 1–6.

[2] M. D. Marsico, M. Nappi, and D. Riccio, "FARO: Face recognition against occlusions and expression variations," IEEE Trans. Syst., Man, Cybern. A, Syst., Humans, vol. 40, no. 1, pp. 121–132, Jan. 2010.

[3] A. F. Abate, M. Nappi, D. Riccio, and G. Tortora, "RBS: A robust bimodal system for face recognition,"Int. J. Softw. Eng. Knowl. Eng., vol. 17, no. 4, pp. 497–514, 2007.

[4] N. J. Belkin, P. B. Kantor, E. A. Fox, and J. A. Shaw, "Combining evidence of multiple query representation for information retrieval," Inf. Process. Manag., vol. 3, no. 31, pp. 431–448, 1995.

[5] R. M. Bolle, J. H. Connell, S. Pananti, N. K. Ratha, and A. W. Senior, "The relation between the ROC curve and the CMC," inProc. 4th IEEE Work. Automat. Identification Adv. Technol., 2005, pp. 15–20.

[6] D. Delgado-Gomez, F. Sukno, D. Aguado, C. Santacruz, and A. ArtesRodriguez, "Individual identification using personality traits,"J. Netw. Comput. Appl., vol. 33, no. 3, pp. 293–299, May 2010.
[7] M. D. Marsico, M. Nappi, and D. Riccio, "HERO: Human ear recognition against occlusions," inProc. IEEE Comput. Soc. Workshop Biometrics—In Assoc. IEEE Conf. Comput. Vis. Pattern Recognit.—CVPR, San Francisco, CA, 18 Jun. 2010, pp. 320–325.
[8] R. Distasi, M. Nappi, and D. Riccio, "A range/domain approximation error based approach for fractal image compression," IEEE Trans. Image Process., vol. 15, no. 1, pp. 89–97, Jan. 2006.