Modern Applied Science; Vol. 12, No. 6; 2018
ISSN 1913-1844 E-ISSN 1913-1852
Published by Canadian Center of Science and Education

Security in Cloud Computing Using Hash Algorithm: A Neural Cloud Data Security Model

Osama Harfoushi[1] & Ruba Obiedat[1]

[1] Department of Business Information Technology, The University of Jordan, Amman, Jordan

Correspondence: Osama Harfoushi, Department of Business Information Technology, The University of Jordan, Amman, Jordan. E-mail: o.harfoushi@ju.edu.jo

Received: April 8, 2018 Accepted: April 16, 2018 Online Published: May 31, 2018

doi:10.5539/mas.v12n6p143 URL: https://doi.org/10.5539/mas.v12n6p143

Abstract

Cloud computing is the delivery of computing resources over the Internet. Examples include, among others, servers, storage, big data, databases, networking, software, and analytics. Institutes that provide cloud computing services are called providers. Cloud computing services were primarily developed to help IT professionals through application development, big data storage and recovery, website hosting, on-demand software delivery, and analysis of significant data patterns that could compromise a system's security. Given the widespread availability of cloud computing, many companies have begun to implement the system because it is cost-efficient, reliable, scalable, and can be accessed from anywhere at any time. The most demanding feature of a cloud computing system is its security platform, which uses cryptographic algorithm levels to enhance protection of unauthorized access, modification, and denial of services. For the most part, cloud security uses algorithms to ensure the preservation of big data stored on remote servers. This study proposes a methodology to reduce concerns about data privacy by using cloud computing cryptography algorithms to improve the security of various platforms and to ensure customer satisfaction.

Keywords: cloud computing, cryptography, algorithms, symmetric, asymmetric, hashing

1. Introduction

Since the invention and implementation of cloud computing, the majority of cloud providers have tried to implement better security measures to ensure customer satisfaction, effective and efficient data movement, and high levels of security. For these reasons, the most brilliant minds in the technology discipline developed numerous security measures (Bhardwaj et al., 2016). These measures were developed through the implementation of symmetric, asymmetric, and hash algorithms. Among the three, symmetric and asymmetric were used on a large scale, but hashing was implemented on a small scale. With the increasing use of symmetric and asymmetric algorithms, human beings deduced better ways of intercepting, manipulating, and changing information; therefore, the security of cloud computing became an area of increasingly intense concern (Khan and Tuteja, 2015).

This study aims to understand why the two models of algorithms are vulnerable to threats, such as malware attacks, cyber-attacks, hacking, and changes in the security protocol. This paper draws on advanced qualitative research by gathering information from various peer-reviewed articles, books, and other publications to understand why asymmetric and symmetric algorithms are vulnerable (Mantri et al., 2011). In addition, the methodology uses three different examples of algorithms to explain why the hash algorithm is different from the other two.

The research focuses on two models that use asymmetric algorithm Rivest-Shamir-Adleman (RSA) and symmetric algorithm Data Encryption Standard (DES), examining their functional features and elaborating on how they are implemented in cloud security (McAndrew, 2016). Finally, given the problems with the two methods, this paper proposes the Neural Cloud Data Security Model, illustrating its unique features, as a means to resolve the issues raised by the asymmetric and symmetric algorithms.

2. Literature Review

Cloud computing is the use of integrated systems over the Internet. These methods include analytics, big data, servers, storage, databases, and networking. In essence, cloud computing establishes the best enterprise and individual use of software and hardware that is controlled and managed by an external third party located in a remote location. Moreover, the system ensures that the users can access information and computer resources from

any position and area as long as an Internet network is available (Rass and Slamanig, 2013).

Also, the platform ensures that the collection of resources—such as data storage capacity space, networking between various individuals and firms, computer engineered power, and user applications—is operational for all clients. Moreover, it is better and more convenient for ensuring regular and on-demand Internet access that shares one structure that connects to all the computer resources. The process of resource sharing is rapidly released with minimal effort from either the management or the user/client (Singh and Supriya, 2013).

Since many companies rely on the movement of data, it is essential that this information is protected from any unauthorized access and modification or denied access. The majority of institutions implement a protection mechanism in their systems using symmetric and asymmetric algorithms. Conversely, hash algorithms are seldom used in designing security cloud systems. Cloud computing enhances the tools of cryptography to ensure the safety of data and the databases.

The security goals of cloud cryptography are authentication, confidentiality, non-repudiation, and integrity of the moved and stored information on the remote servers (Hashizume et al., 2013). First, authentication is the process of identifying whether something or someone is what and who it is supposed to be. Second, confidentiality defines the method that is directly linked to lack of privacy and database access through hacking. Third, non-repudiation confirms why an individual or company cannot decline their agreement and originality of a seal, signature, and information they created. Lastly, integrity assures the value, accuracy, and consistency of data stored in a database.

With the increase in the implementation and employment of cloud services, most providers lack a proper security system to protect data from threats and attacks. At the same time, most companies employ cryptographic algorithms to help preserve the ciphertext (Mantri et al., 2011). During the cyphering or encrypting of the version, it is advisable to use extended algorithms because of its complexity and capability to be un-hackable or attacked. When the algorithm is extended, then the system becomes more effective and efficient. Cloud cryptography uses algorithms to create a defense mechanism to protect the data from being hacked by any person. To ensure that the data and database retain confidentiality, cryptography enhances the implementation of symmetric, asymmetric, and hashing algorithms. On the other hand, it manages the integrity element by using hashing algorithms (Bhardwaj et al., 2016).

The asymmetric algorithm is a utility used by cryptography since its primary and private key can be divided into two: primary and secondary key configurations. Anyone can own the public key, while the private key is to be hidden and outside of people's knowledge. This function has two main uses: authentication and confidentiality. The encryption of works differs from symmetric encryption because the person with the public key has the authority to encrypt the text, while the second part with the private key can decrypt the document (Hashizume et al., 2013). Examples of systems developed using the asymmetric algorithm are Elliptic-curve cryptography, Rivest-Shamir-Adleman (RSA), and asymmetric utilities, among others.

In addition, concerns have been raised about the asymmetric algorithm's integrity. Research indicates that with the method using both primary and secret keys, computation of the asymmetric key requires that it be longer than that of the private key to ensure equilibrium between the two. Without the two keys being in balance, the entire system remains vulnerable to attack. Also, public-key cryptography tends to be more vulnerable to severe attacks, especially in cases involving an intermediary that can subject the system to malware infections (Rass and Slamanig, 2013).

On the other hand, Symmetric Algorithm involves one shared private key used to encrypt and decrypt information. Also, it can process significant data from a computing standpoint. Moreover, the key has limited authority and lower overhead on the method, and it is efficient and effective for encrypting and decrypting messages (Buchanan, 2017). Symmetric algorithms encrypt plaintexts as a group or block of 64-bit units of fixed numbers. However, despite its extensive use, the symmetric system has problems (Rass and Slamanig, 2013).

One problem relates to the exchange of crucial shared function over the unsecured network is that the sender and receiver share the same private key during the decryption and encryption process. In such cases, a third party can gain access to the critical algorithm in the event that an unsecured network is used for data movement between the service providers and the client. Also, to ensure the data is secure the sender needs to change the original word (Dhivakar, 2014).

Another problem can arise if the sender alters the problem of confirming in case the message content: In the event that the word being sent is intercepted by a hacker and the information is changed, the key used for modification becomes compromised on the side of the receiver, and the system remains compromised (Rao and Selvamani, 2015).

A final problem concerns utility available for cracking the symmetric encryption: Various tools have been developed to hack and break the algorithm, thus, exposing the plain text to third parties (Jegadeeswari et al., 2016).

The Hash Algorithm is a unique means of applying security to cloud-based systems. Given its exclusive platform, providers who understand its importance implement it because it is associated with the Neural Cloud Data Security Model. First, it functions with the same hash algorithm; therefore, the sender and the receiver have to use the same algorithm to test the authenticity of the message, and if the algorithm is identical, then the news will not be intercepted or changed. Although several studies identify its weaknesses, research identifies a 1 percent to 10 percent chance of vulnerability. Furthermore, it does not use the encryption model of data security, hence, reducing its complexity.

With the adjustments in technology, many users and companies have begun implementing cloud services for personal or business purposes. In addition, the Y-generation is addicted to the use of technology, which provides cloud service providers with a stable market. The invention and innovation of cloud services created a dynamic change in the global political, social, and economic system (Hashizume et al., 2013). With the increase of providers, various policy issues arose. First, as stated earlier, cloud security is one of the major faults of the entire system. Although much research has been carried out to understand the source of the problems, it is difficult to focus on one cause alone.

According to Violino (2018), data breaching was the most used platform to attack any cloud service. The main objectives of data breaching can be classified into three groups: (1) Targeted attacks: where hackers are after a particular piece of information; (2) Simple human error: where users forget to protect data or use small ciphered keys that can be easily cracked; (3) Client mistake or poorly designed security system.

Targeted attacks related to cloud computing include insufficient identity, credentials, and account management. For example, hackers can create fake identifications and pretend to be legit users, and operators and service providers can gain access to, modify, or manipulate data (Violino, 2018). Furthermore, this group has advanced management schemes, such as creating control panels and management user interfaces, which can track and access data. Moreover, these individuals can release malware software to that can later affect the entire cloud network, thus causing problems within the discourse (Singh and Supriya, 2013).

Cloud computing is a vast field that supports an array of software uses and configurations. Most cloud services deal with on-demand software. Given that these providers deal with software, hackers have deduced methods of using fake and insecure interfaces and apps to attract new clients (Khan and Tuteja, 2015). These apps expose the cloud system to viruses. They use the same software to assist in provisioning, managing, and monitoring information movement within the network. Therefore, providers and programmers need to design and implement better software that protects against malicious attacks (Mantri et al., 2011).

The cost-efficient nature of these services has led to an increase of registered members in a single cloud. With the rise in the numbers of users comes congestion within a specific protocol; at the same time, counterfeit providers use the same information they gather from these individuals to intercept their messages or manipulate them (Mantri et al., 2011). To protect the data from falling into the wrong hands, scientists and software programmers have developed various ways of cryptography in information, thus the introduction of cryptosystems.

Given that the majority of security systems use symmetric and asymmetric algorithms, and few providers employ hashing algorithms, more systems have been victims of these attacks. Currently, the hash algorithm has proved useful because of its role in the generation of the Neural Cloud Data Security Model, also known as a hybrid security system. The system assures integrity, flexibility, reduced latency, and cost-effectiveness.

This literature review has expanded on some of the significant problems facing cloud computing security. Many of these vulnerabilities occur due to a faulty system, human error, and, at times, cyberbullying (McAndrew, 2016). Cryptography remains the most frequently implemented method used to enhance security measures but with the numerous defaults in symmetric and asymmetric algorithms (Yan, 2012). The methodology section outlines the author's study of asymmetric and symmetric algorithm patterns and proposes a better system that can ensure maximum cloud computing security.

3. Methodology

The methodology proposes an advanced cryptosystem model based on security solutions to the problems experienced by RSA and DES algorithms' functionality. Moreover, to develop the system, the researcher selected the two most used algorithm methods and analyzed them according to vulnerability. These systems were RAS, designed using an asymmetric algorithm, and DES, which uses the symmetric algorithm for security, accountability, and credibility properties (Buchanan, 2017). Moreover, even though more cloud-based systems use symmetric and

asymmetric architecture, the hash algorithm identifies a new paradigm because of its compatibility property (McAndrew, 2016). The primary reason for choosing the model was due to the fact that technology is evolving: scientists and engineers are no longer establishing considerable electronics to support innovation and creativity. Currently, security is more software than hardware oriented, and the hash algorithm is more compatible with the system since its use embraces improved open-architecture security models (Mantri et al., 2011). Moreover, since cloud computing has reduced the use of bulky hardware, an appropriate model, such as the Neural Cloud Security System can be implemented to enhance the accountability, security, and integrity of data.

Note: The methodology will analyze the functionality and problems of the DES and RAS cryptosystems. Furthermore, it will identify the key reason for proposing the Neural Cloud Security System and illustrate why this model is the perfect system to replace the other two cryptosystems.

3.1 Asymmetric Algorithm: RSA Cryptosystem

The RSA cryptosystem was first designed by three students at the Massachusetts Institute of Technology (MIT)—Ron Rivest, Adi Shamir, and Leonard Adleman in 1977 (Dhivakar, 2014). The system uses a series of positive integers; thus, it depends on the use of different but mathematically linked keys. These keys are identified as public and private keys (Buchanan, 2017). Anyone can have access to the public key, but the private one remains a secret. In addition, the system is used in protocols, such as SSH, OpenPGP, SSL/TLS, and S/MINE, among others (Singh and Supriya, 2013). The main reason for using the algorithm in these systems is the fact that its public and primary keys can encrypt and decrypt any message (Gupta, 2014). Furthermore, the model is used to code programs such as browsers (Rass and Slamanig, 2013). The algorithm's properties are developed in positive integers, and the algorithm for the public-key cryptography comprises the public and private keys as illustrated in Figure 1.

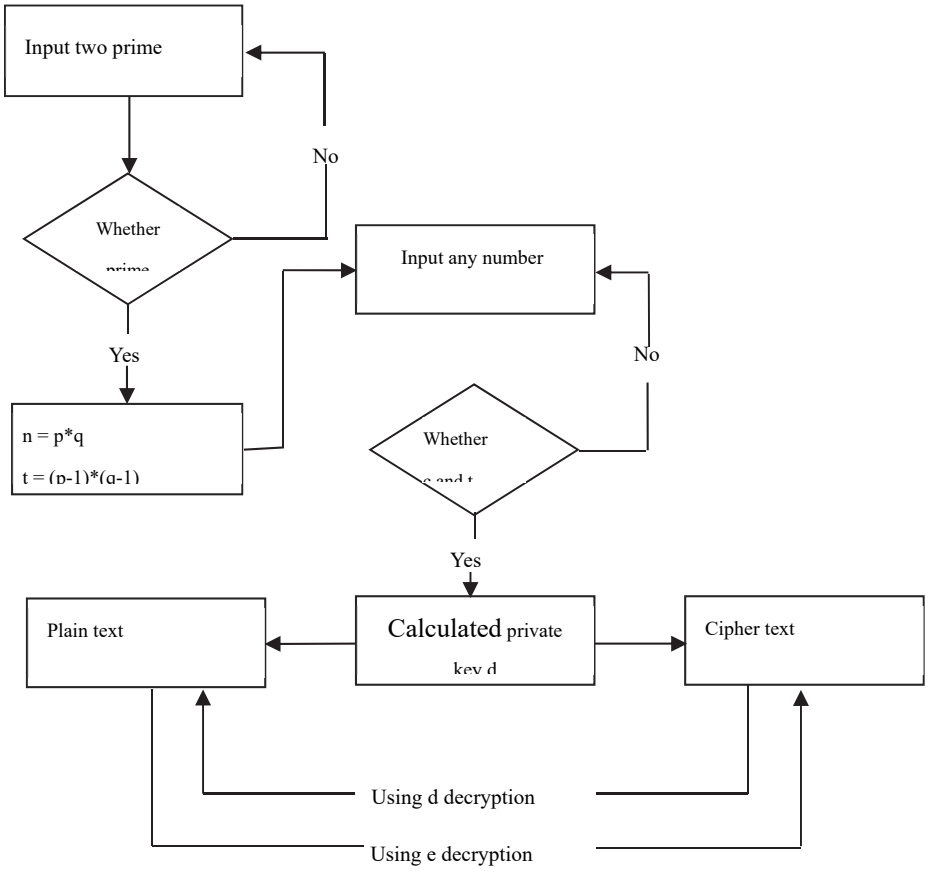

Figure 1. RAS algorithm block diagram (Singh & Supriya 2013)

The RSA algorithm is a perfect security system, but its protocol failures enhance its weaknesses and vulnerability

to attacks (Singh and Supriya, 2013). Most protocol failures are generated directly from errors created by mathematical calculations and approximation that does not depend on restructuring the algorithm's strength. The mathematical challenge and manipulation issues an attack, which adapts the platform of introducing additional data from the protocol failure to bypass the security without following the factorization process (Stanoyevitch, 2010).

3.2 Symmetric Algorithm: Data Encryption Standard Algorithm

The Data Encryption Standard (DES) algorithm, as shown in Figure 2, was published by the National Institute of Standards and Technology in 1977 (Stanoyevitch, 2010). This algorithm uses a symmetric-key block cipher that undergoes 16 rounds through the Feistel structure. Also, each block size is 64-bits and so is the critical length. The valid range of ciphertext usually is 56-bits because eight out of the 64-bits are not used in the encryption process. The first stage is the initial permutation on the 64-bit block of data, followed by a split of two 32-bit blocks of data; thus, L0 and R0 are passed into the Feistel rounds. Each round is identical, and this ensures that the security algorithm increases, while the temporal efficiency decreases. At the 16th round the L15 and R15 concatenation is permuted using a function that is of the same reverse of the first permutation (Singh and Supriya, 2013).

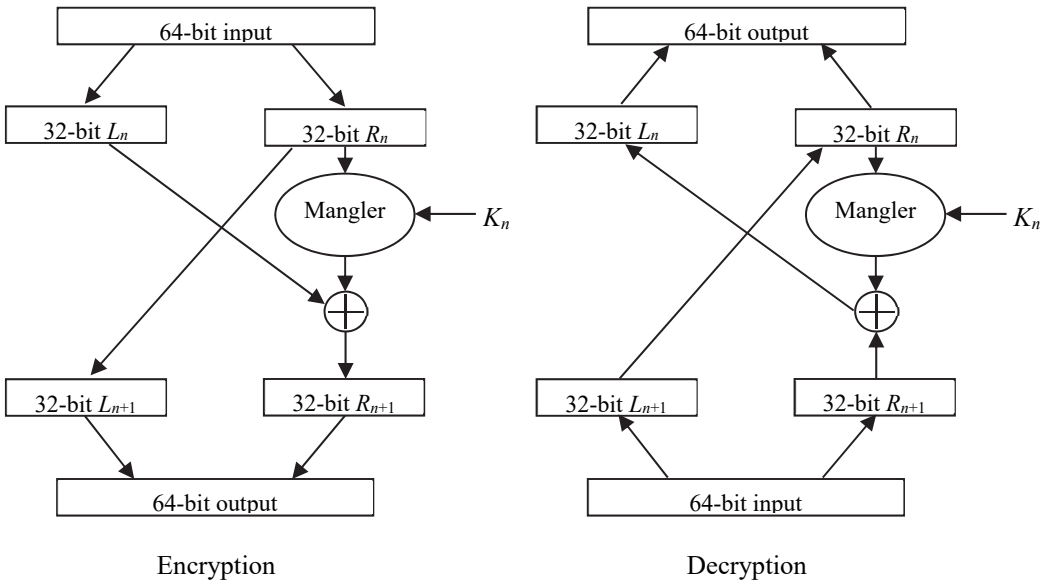

Figure 2. DES Algorithm Block Diagram (Stanoyevitch 2010)

For decades, researches have debated why DES failed; to some extent, researchers have narrowed their study of the algorithm but not the device created to handle the encryption and decryption of the sensitive information. First, DES failed because of the key length. With the use of numerous keys, the system is incompatible with many applications because it is hardware based. Furthermore, it is more vulnerable; therefore, most providers do not use it to move information through public networks like the Internet. Since the key length and design are the causes of the failure of DES, its generation of weak binary digit values makes it easy to hack. DES uses the symmetric algorithm to function, and it uses 64-bits, and out of 64-bits only 56-bits are used; thus, the division of the binary digits ensures each block contains 16-bits. Since the generation of the keys is in random, 0 and 1, many hackers use cryptanalysis to interpret the pattern of the production of keys and plan attacks to the system.

3.3 Hash Algorithm

According to the accumulated research, we identified the hash function as a new algorithm that offers improved dimensions and properties to produce improved security results. First, it is a fundamental building block for modern security models (Bhardwaj et al., 2016). Moreover, it is used to convert sizeable amounts of random information into small fixed data (Gupta, 2014). Also, its algorithm enhances friendly terminologies such as the digest (data output). Nevertheless, the system does not rely on either primary or secondary key to operate; thus, its coding enhances one-way operation. The one-way operation enables its security structure to maintain the process where input data cannot be generated from a specific digest (Mantri et al., 2011). The algorithm makes and verifies

digital signatures; it conducts message integrity checks, derives sub-keys in key-generated protocols and algorithms, and creates pseudorandom figures to ensure absolute security (Gupta, 2014).

3.4 Neural Cloud Security Proposed Model

This study focused on the following areas of cloud computing security: the provision of data isolation and protection of crucial information; the creation of an algorithm that uses the neural network concept of essential production; and the establishment of a recognized way of competing for the network without any hacking.

The neural system is efficient and effective for nearly all types of problems because of its high confidentiality levels. The Sensitive Data Component (SDC) provides information privacy and efficient isolation. Moreover, it is used to encrypt and decrypt the vital information by through the use of the Counter Propagation Neural Network. It generates a cost-efficient, effective, and significant storage system that enhances security (Singh and Supriya, 2013).

Absolute privacy and security is accomplished by breaking down the vital information. This stores data more efficiently and confidentially via virtual servers. The essential data is secured using the cryptographic algorithm with the Internet and stored in protected form to enhance confidentiality levels (Jegadeeswari et al., 2016).

The proposed model includes an SDC. The SDC is responsible for decomposing the essential datasets into fragments. With the presence of the neural network, the particles are ciphered using cryptographic algorithm present in the servers. As shown in Figure 3, The Neural Cloud Data Security Model with the two-layer neural network is responsible for the encryption and decryption of the data to produce the original data (Jegadeeswari et al., 2016).

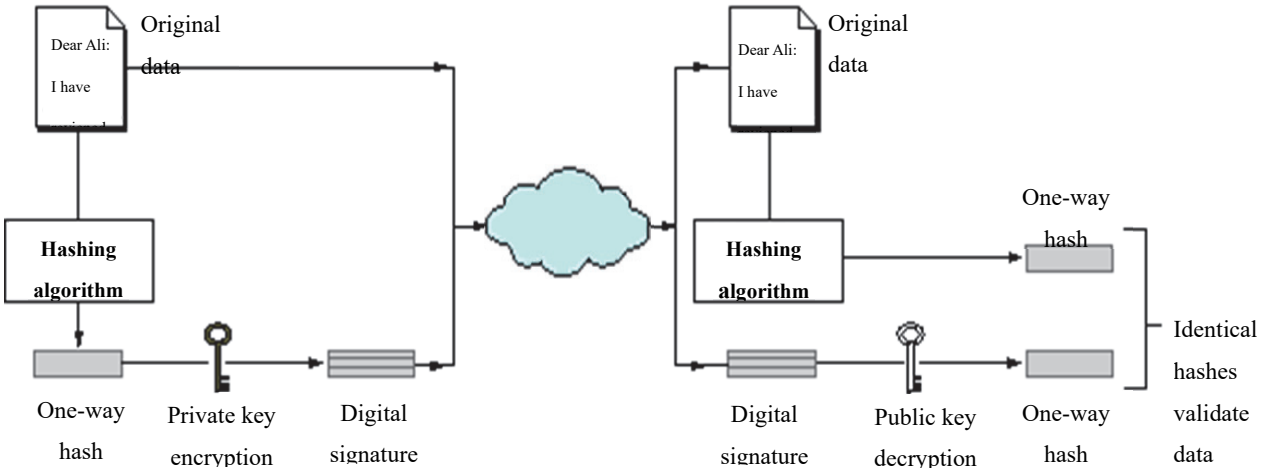

Figure 3. Architecture of a Neural Cloud Data Security Model (Jegadeeswari et al. 2016)

The following two algorithms suggest a chain of encryption and decryption method within the SDC of Neural Cloud Data Security Model, and the sensitive data component is the original message sent by the sender and that received by the recipient. These algorithms increase the security of the data over cloud computing systems.

Algorithm 1: The algorithm for the Sensitive Data Component (sending the message)

Input: Information retrieved from the cloud servers

Output: Encrypted information

Phase 1. Begin.

Phase 2. Data from the database, only sensitive data are read.

Phase 3. Sensitive data set fragmentation process.

Phase 4. Storage of fragmented data sets.

Phase 5. The encryption of the essential data set.

5.1. Use of cryptographic algorithm to implement encryption within the neural network.

5.2. The public key encrypts the data while the private key decrypts it.

Phase 6. Hashing structure ensures the information is stored on the servers.

Phase 7. Stop.

The data input to the model is received from the virtual servers, and the one sent out is the protected information stored in the dynamic table. The model studies the report data from the virtual database where it is the critical context. The raw block of data is then divided vertically and horizontally. These games can be learned using the dynamic hash function where key = m mod n. The vital information is later encrypted using cryptographic algorithms (Jegadeeswari et al., 2016). Algorithm 2 is the process that ensures the original message/sensitive message is allocated a private key and a digital signature.

Algorithm 2: The process for the dynamic hashing fragment model

Input: Cloud data set indexing value

Output: Important encrypted data

Phase 1. The sensitive data sets are retrieved from various data centers to the cloud database.

Phase 2. The horizontal and vertical fragments of the key and non-key data sets.

Phase 3. The arrangement of the data sets is in a dynamic hashing type.

Phase 4. The hashing function retrieves the essential information from the virtual servers.

Phase 5. The separated data set output is taken to the Cloud Data Security Model.

Phase 6: Stop.

4. Conclusion

The primary function of the cloud is to enhance confidentiality, efficiency, and integrity when accessing information from a cloud database. Numerous threats and attacks from various people have raised the alarm on the security levels implemented by most cloud providers. Since most companies employ asymmetric and symmetric algorithms to support the security level of data movement and storage, numerous cases of security issues have occurred since these algorithms are vulnerable. This study briefly discussed the Neural Data Security Model, which has better features since it uses the hashing algorithm. Moreover, the model promotes high privacy and security in storage and data movement. The research paper focused on the theoretical elements of the Neural Cloud Data Security Model. In summary, the model proved to be more cost-efficient, credible, and confidential compared to the other two models.

References

Bhardwaj, A., Subrahmanyam, G., Avasthi, V., & Sastry, H. (2016). Security algorithms for cloud computing, *International Conference on Computational Modeling and Security, 85*, 535–542.

Buchanan, B. (2017). Cryptography. S.L.: River Publishers Series in Information Science and Technology.

Dhivakar, A. (2014). To Implement a Multi-Level Security in Cloud Computing using Cryptography Novel Approach: Security in Cloud Computing. Munich: GRIN Verlag.

Gupta, P. C. (2014). Cryptography and Network Security. Delhi: PHI Learning Pvt. Ltd.

Hashizume, K., Rosado, D. G., Fernández-Medina, E., & Fernandez, E. B. (2013). An analysis of security issues for cloud computing. *J. Internet Services and Appl., 4*(5), 1–13.

Jegadeeswari, S., Dinadayalan, P., & Gnanambigai, N. (2016). Neural-based security approach for cloud databases using counter propagation. *Indian J. Sci. Technol., 9*(16), 1–10.

Khan, S. S., & Tuteja, R. R. (2015). Security in cloud computing using cryptographic algorithms. *Int. J. Innov. Res. Comput. Commun. Eng., 3*(1), 148–154.

Mantri, A., Kendra, S. N. S., Kumar, G., & Kumar, S. (2011). High-performance architecture and grid computing. Conference proceedings from High-Performance Architecture and Grid Computing conference, Chandigarh, India, July 19–20, 2011. Berlin: Springer Science and Business Media.

McAndrew, A. (2016). Introduction to Cryptography with Open-Source Software. Boca Raton, FL: CRC Press.

Rao, R. V., & Selvamani, K. (2015). Data security challenges and its solutions in cloud computing. *Procedia Comput Sci., 48,* 204-209.

Rass, S., & Slamanig, D. (2013). Cryptography for Security and Privacy in Cloud Computing. London: Artech House.

Singh, G., & Supriya. (2013). A study of encryption algorithms (RSA, DES, 3DES, and AES) for information security. *Int. J. Comput. Appl., 67*(19), 33–38.

Stanoyevitch, A. (2010). Introduction to cryptography with mathematical foundations and computer implementations. Boca Raton, FL: CRC Press.

Violino, B. (2018). *The Dirty Dozen: 12 Top Cloud Security Threats for 2018*. Retrieved February 7, 2018, from https://www.csoonline.com/article/3043030/security/12-top-cloud-security-threats-for-2018.html.

Yan, S. Y. (2012). Computational number theory and modern cryptography. London: John Wiley & Sons.