https://doi.org/10.48048/tis.2022.3970

New Approach Based on Homomorphic Encryption to Secure Medical Images in Cloud Computing

Ali Kartit

LTI Laboratory, ENSA, Chouaib Doukkali University, El Jadida, Morocco

(Corresponding author's e-mail: alikartit@gmail.com)

Received: 2 December 2020, Revised: 14 May 2021, Accepted: 22 May 2021

Abstract

In recent years, there has been a growing demand for the adoption of cloud in healthcare to process medical data. Unfortunately, this emerging new paradigm faces several challenges, as customer data is stored on remote servers rather than on premise solutions. This is considered to be the main root cause of the major security vulnerabilities encountered in outsourced calculations. Indeed, to solve this problem we used encryption; customers should encode their sensitive data when considering adopting cloud services. One of the biggest challenges is establishing controls that completely protect secret data from insider attacks, while also supporting the computations. In addition, applying complex encryption schemes can have a negative effect on system performance.

This study focuses in particular on homomorphic techniques and their main applications, as well as their limitations. Unlike traditional encryption methods, homomorphic schemes are used not only to reduce the security risks associated with cloud technology, but also to process cipher texts. However, current efforts in this area need to be further developed to strike the right balance between privacy risks and data utility. In this regard, we offer a hybrid approach that offers the possibility of quickly processing health records in a secure manner. Our main contribution is the proposal of a new method based on Hadoop MapReduce functions in conjunction with a multi-agent system to maintain data confidentiality. In addition, and to design intelligent distributed computing for efficient management of Virtual Machine (VM) workloads, we used a method based on the Bat Algorithm (BA).

Keywords: Cloud computing, Medical image processing, Homomorphic encryption, MapReduce, Hadoop framework, BA algorithm

Introduction

A significant advantage of using the electronic health record (EHR) is to increase practice efficiency and cost savings [1]. For a successful implementation of the EHR, blockchain [2] and cloud computing [3] are the main technologies for efficiently storing, sharing and managing clinical data. The contribution to the secure use of cloud services in an intelligent health environment is the main objective of this work. The obvious solution to secure cloud storage is the use of classic cryptographic algorithms such as RSA (Rivest, Shamir and Adleman), DES (Data Encryption Standard), and AES (Advanced Encryption Standard). However, to secure the image processing process usually requires more innovative techniques to secure cloud services. There are now a variety of methods to solve this problem, including service oriented architecture (SOA) [4], homomorphic encryption techniques [5,6], shamir secret sharing schemes (SSS) [7,8] and the segmentation methods [9-11]. In general, the existing approaches are not sufficiently mature for practical uses, especially in the case of medical data with high level security requirements. For this reason, we must take the necessary alternative measures to protect the privacy of health data when using remote services. To a certain extent, homomorphic encryption remains the most promising method for securing externalized data files, regardless of its shortcomings (It is still, despite dramatic improvement over the years, incredibly slow and non-performant, making it a non-starter for most business applications). In this regard, we provide a simple way to improve the performance of cloud-based image processing while protecting privacy. It involves a distributed computing model with a set of processes that cooperate to achieve a particular level of performance. In this case, the present study deals with the use of homomorphic methods for efficient image processing in cloud-based applications, mainly focusing on data security and privacy protection.

In mathematics, homomorphism is essentially a map between 2 groups. At the same time, it is designed to keep all the basic algebraic structure [12,13].

Definition 1. We assume that (G, *) represents a set G and an operation *, which combines any 2 different elements, a and b, to form another element, denoted a * b. Let's consider 2 distinct groups (G, *) and (H, ◊), we define f () as a map from (G, *) to (H, ◊), called also Operation Preserving (OP) mappings. We say that f () is homomorphism if it satisfies the following condition (1);

$$f(): G \rightarrow H \quad (1)$$

$$f(a * b) = f(a) \lozenge f(b), \text{ for all } a, b \in G$$

Based on these considerations, this function is a structure preserving a map between 2 algebraic structures as illustrated in **Figure 1**.

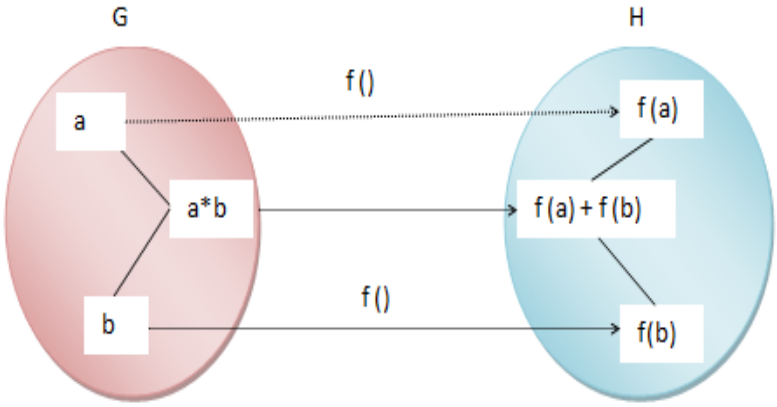

Figure 1 The principle of group homomorphism.

In the context of cryptography, let E and D denote homomorphic encryption and decryption function respectively. Given 2 plaintexts m1 and m2 and 2 corresponding ciphertexts c1= E (m1) and c2 = E (m2).

$$E: P \rightarrow C \text{ (P the plaintext space and C the ciphertext space)}$$

$$E^{-1} = D: C \rightarrow P$$

Definition 2. A scheme is considered homomorphic with respect to an operation ◊ on P if the relation 2 is satisfied;

$$D(E(m1) * E(m2)) = D(E(m1 \lozenge m2)) = m1 \lozenge m2 \quad (2)$$

for some operation ∗ on C.

In practice, this technique allows a user to perform certain algebraic operations on encrypted data. In a cloud environment, these algorithms have the potential to provide solutions that completely protect medical data when using cloud services [14]. Simply put, they are designed to process digital data remotely without compromising patient privacy. For example, we assume that x and y are 2 pixels of a medical image. Ideally, we first encrypt these 2 pixel values using the public key before starting the transfer of medical data to the public cloud. Thanks to the homomorphic properties, the arithmetic operations performed on the original image are equivalent to another performed on the corresponding cipher text. Therefore, applying a homomorphic encryption method in cloud applications is a promising approach to protect the medical image while it is computed offsite, as shown in **Figure 2**. Note that a public key (pk) is used for encryption, and the corresponding secret key (sk) is used for decryption.

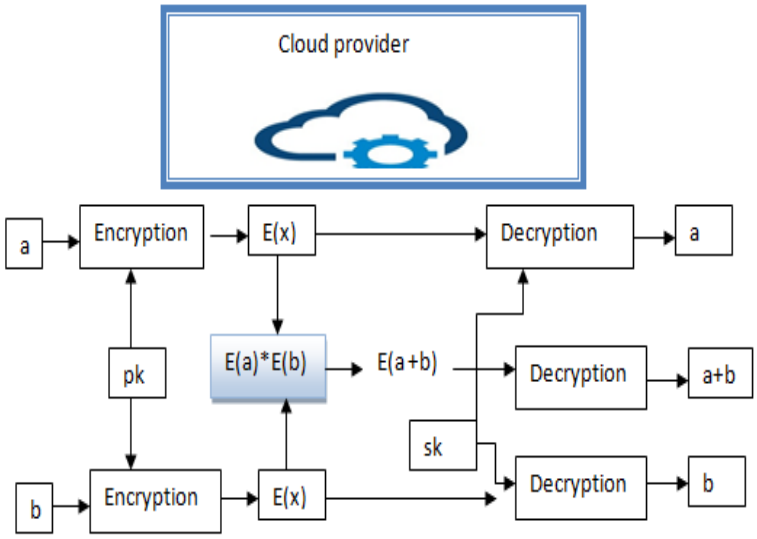

Figure 2 The usage of homomorphic in cloud computing.

Ideally, partially homomorphic (PH) allows only 1 operation on the encrypted pixels. In this context, the Paillier cryptosystem [15] is used to exploit 2 pixel values in order to obtain the encrypted result which, once decrypted, corresponds to the result of the addition operation associated with these 2 values, as shown. relation 3.

$$\begin{aligned} c1 \cdot c2 &= (g^{m_1} r_1^n)(g^{m_2} r_2^n) \pmod{n^2} \\ c1 \cdot c2 &= g^{m_1+m_2}(r_1 r_2)^n \pmod{n^2} \\ c1 \cdot c2 &= E(m_1+m_2) \end{aligned} \quad (3)$$

with: (g,n) is the public key, plaintext m < n, select a random r < n and split m into m_1, m_2.
c1, c2: ciphertexts, m_1, m_2: plaintexts.

Fortunately, RSA algorithm [16] offers also the possibility to perform homomorphic multiplication on encrypted data, as shown in 4.

$$\begin{aligned} c1 \cdot c2 &= m_1^e . m_2^e \pmod{n} \\ c1 \cdot c2 &= (m_1 . m_2)^e \pmod{n} \\ c1 \cdot c2 &= E(m_1 . m_2, pk) \end{aligned} \quad (4)$$

with: (e,n) is the public key, plaintext m_i < n, select a random r < n and split m into m_1, m_2.
c1, c2: ciphertexts, m_1, m_2: plaintexts, pk: public key.

Unlike PH (Partially Homomorphic) encryption, FHE (Fully Homomorphic Encryption) supports various mathematical operations. It has allowed the evaluation of arbitrary circuits composed of multiple types of gates of unbounded depth, and is the strongest notion of homomorphic encryption. There is currently a wide range of different FHE models [17]. In this context, Gentry [18] proposes the first conceptual scheme which solved this open problem, supporting both addition and multiplication in the ciphertexts format. Primarily, this method is based on somewhat homomorphic (SW) schemes. Recently, a new method based on the Gentry framework has been suggested by Dijk *et al.* [19] to allow calculations on cipher texts. Among the various FHE techniques, Enhanced Homomorphic Encryption Scheme (EHES) [20] is widely used to process cipher texts, allowing several arithmetic operations without data decryption. This ensures that medical records are properly protected from unreliable clouds.

In light of these facts, homomorphic encryption methods would undoubtedly reduce the security risks and potential threats associated with cloud services. Most importantly, our approach is designed to allow a third party to process encrypted medical images without decrypting them. Here is a very simple example, shown in **Figure 3**, of how the homomorphic encryption scheme might work in healthcare.

First, we code a medical image to get an encrypted form E(x). Second, we perform pixel-based image processing operations, such as h linear image filtering, to obtain F2(E (x), h). Finally, we use the private key to get the processed image, i.e. y = D (F2 (E (x), h).

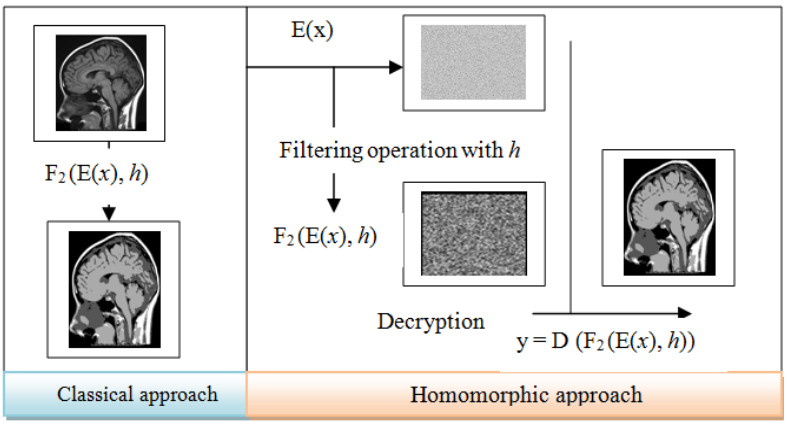

Figure 3 Image processing using homomorphic encryption.

In this article, we start with a discussion of existing research, and then we provide an analysis of current frameworks to assess their limitations and make appropriate recommendations. In this context, the authors of [21] present a new scheme based on homomorphic methods to process encrypted images and prevent malicious disclosure of data. Their framework is based on the Learning With Error (LWE) technique to improve classical homomorphic schemes. In this case, it is possible to perform basic mathematical operations on encrypted images. Ideally, customers should encode digital recordings before uploading them to the cloud. In the next step, users perform addition and multiplication operations on the data stored in cloud computing. Therefore, the proposal is an adequate solution to address privacy issues in cloud services. However, this model does not always provide satisfactory results, especially in terms of system performance.

In [22], the authors mainly rely on the Residue Number System (RNS) technique to facilitate the evaluation of useful mathematical functions. In essence, such a solution allows cloud providers to perform some specific arithmetic operations on encrypted images remotely without having the private key for decryption. In fact, the proposition is considered to be homomorphic with regard to the operations of addition, subtraction and multiplication. Most importantly, it ensures that medical information is only viewed by authorized users. Experiments show that the proposed techniques allow cloud providers to apply the Sobel filter to encrypted images without decrypting them. However, the main disadvantage of this approach is that it takes a long time to analyze digital images.

In order to show the utility of the homomorphic encryption approach, Kanithi and Latha [23] are developing an online tool to remotely process medical images. The main goal of this solution is to maintain confidentiality when outsourcing data processing to untrustworthy cloud providers. In this regard, the authors use Paillier's algorithm with Discrete Wavelet Transform (DWT) to perform efficient data analysis. As a rule, the Paillier cryptosystem is used to make additions to encrypted values. In light of this fact, we can quantify the approximation coefficients and then process them. The simulation results show that this technique can perform 2-D Haar wavelet transformation (HWT) in an encrypted domain. In the same line, it is possible to get the secret image using Paillier and IDWT (Inverse Discrete Wavelet Transform).

In Habeep and Raj [24] develop a framework for effectively processing medical images. In this case, their main purpose is to alleviate privacy concerns when processing data remotely. In this regard, they use an effective approach which is generally based on the DWT / IDWT concept. Specifically, they apply this technique in the last step to handle data expansion issues. To successfully decrypt medical records, they use the inverse multiplicative method (MIM) to minimize quantification effects. Basically, the proposal relies on Paillier's algorithm to calculate the encrypted images. However, this solution is designed for a specific operation and cannot be applied in all other cases. In Yang *et al.* [25], the authors extend Gentry's homomorphic encryption to support floating-point arithmetic operations. Unlike conventional approaches, this framework uses symmetric encryption instead of public key encryption to encode digital

data. Based on the simulation results, the proposal prevents statistical analysis attacks from reaching sensitive data. Given this fact, this framework can be adopted in cloud computing to process medical data. Of course, the computation costs are the main drawback of this solution because it relies heavily on conventional homomorphic encryption schemes.

Although homomorphic techniques appear to be a promising approach, there are only a few successful cloud-based image processing implementations. In reality, there are still many issues facing this new concept that hinder its adoption in processing data in cloud computing. First, the existing schemes are not suitable for large image data, as they encrypt, in most cases, each pixel separately. Second, these methods are designed to perform only simple, basic math operations, such as addition and multiplication. Subsequently, developing robust real world applications using these algorithms is a difficult task. Third, the use of homomorphic cryptosystems to encrypt health records in cloud computing would undoubtedly have a negative impact on system performance [26].

Proposed approach

As noted above, execution time is the main negative aspect of homomorphic encryption. For this reason, these techniques are widely used to encode only simple data like numeric values. To solve this problem, we provide a partitioning method primarily aimed at improving performance for handling large amounts of data. In order to increase the efficiency of the data processing, we divide the health recotds into small portions to support a parallel environment and a distributed platform.

An overview of the proposed method

The central idea of this concept is to distribute the workload among many servers by working in parallel on the data. In fact, managing data at scale requires a high performance-computing environment and as a result, the cloud data center experiences high latency due to workloads [27]. Basically, trust, speed and security are the most influential factors in selecting the right cloud service [28,29]. Thus, we apply a homomorphic schema on each separately created share instead of the entire image to move the workloads to the most appropriate environment. In doing so, this method would produce an encrypted image that requires low computation time not only when encrypting data, but also during processing. Confidentiality can be guaranteed, rendering digital recordings unreadable using homomorphic cryptosystems. The data security block diagram of the proposed method, using the sampling technique and the partitioner, is shown in **Figure 4**.

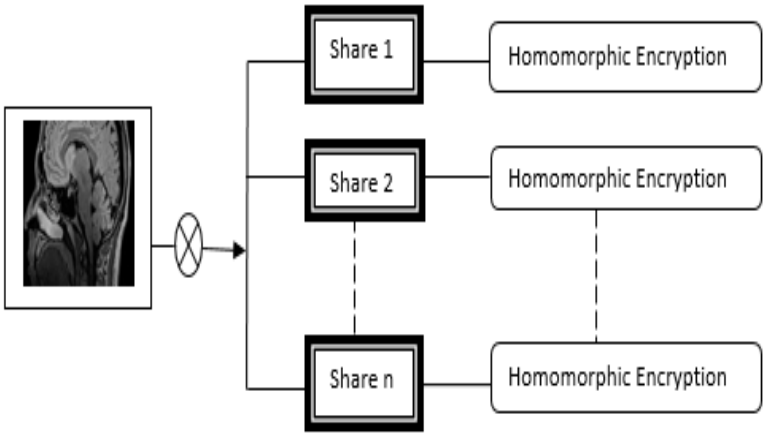

Figure 4 Principle of the proposed method.

Then, each region will be analyzed for a separate agent. To achieve this goal, we introduce a distributed framework based on a multi-agent system [30]. Thus, we use cooperative agents to perform a specific operation on cloud computing [31]. Therefore, an image processing task is typically mapped to different components of a distributed system. In this context, this platform is essentially composed of several agents operating under the control of a defined sub-group manager. The proposed multi-agent system architecture is divided into 3 modules: Master Manager (MA), Region Manager Agent (RMA) and Local Agents (LA), as shown in **Figure 5**.

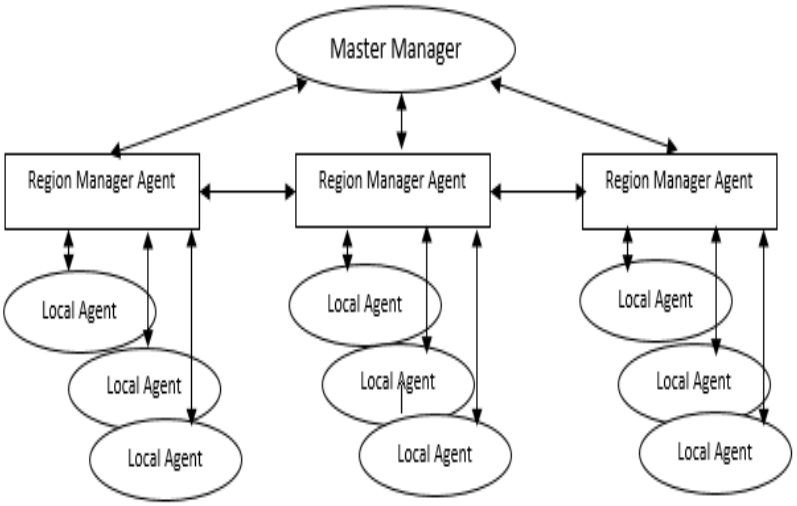

Figure 5 Image processing using multi-agent system.

In such a concept, MA is responsible for overseeing the entire framework. In particular, it creates, initializes or kills RMA processes. In this regard, a set of agents is dedicated to a specific generated region. During this time, each RMA component creates and controls all local agents. In addition, effective collaboration between RMA agents is necessary to achieve a specific goal. Finally, the LA modules operate at the lower level of the image to perform a single medical image processing operation.

To sum up, the secret image is divided into a number of segments. In other words, every node in the cloud platform is not able to reveal information about the secret image. At the same time, it is also useful to perform distributed data processing. For this reason, we introduce a multi-agent system (MAS) in such a way that each region is analyzed by a number of agents. To get the end result, we combine all the separate intermediate results provided by different cloud providers to get the processed image.

Foundations of the proposed framework

For this, we use the MapReduce function to benefit from parallelism during data processing [32]. In this model, we use the Map and Reduce functions to process a huge amount of data in parallel. Functionally, the MapReduce function often divides the input image into independent portions so as to apply the map function in a completely parallel fashion. In this case, the Collapse function combines all of the values from each mapping task to perform a specific operation, and then merges all of the output into a single file. Specifically, a reduction task uses 1 or more keys and their associated attribute values. Therefore, the system performs the grouping operation using a dedicated key for a defined reduction task. In this architecture, the master node is implemented to manage the Map and Reduce functions. Basically, the master node merges files from separate map jobs that are typically dedicated to a particular shrink job, as shown in **Figure 6**.

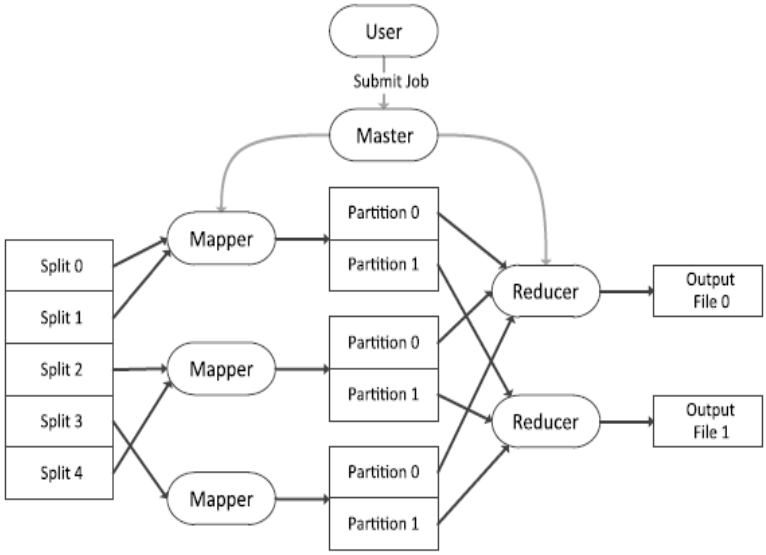

Figure 6 Image processing using MapReduce function.

Based on these considerations, we rely on the Hadoop system to develop secure applications for medical image processing in cloud computing. It achieves this goal by using the Hadoop Distributed File System (HDFS) to save files to various nodes. In this scenario, users upload their digital data to HDFS for processing. To do this, we rely on the MapReduce program to manage input health records very efficiently [33]. First of all, we have divided the secret medical image into several parts. Second, we configure the Map function to encrypt separate regions using homomorphic encryption. Third, we process each small image file. Finally, we combine these intermediate files using the Reduce function to get the processed image. The principle of the proposed framework is presented in **Figure 7**.

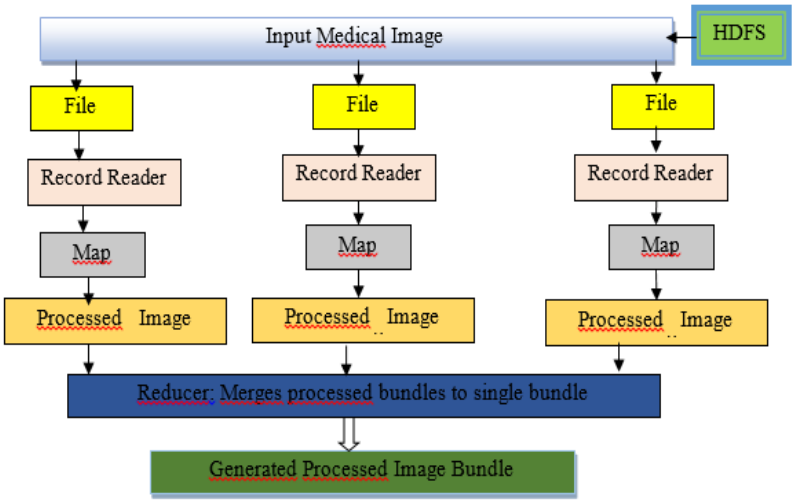

Figure 7 Image processing using Hadoop system.

Essentially, Hadoop Image Processing Interface (HIPI) [34] is an efficient image processing library designed for use with Apache Hadoop MapReduce for parallel programming tasks. The main goal of this solution is to provide a highly parallel distributed framework. Of course, this would improve the performance of the system by distributing the tasks among the various cloud servers available. Since the cloud includes multiple virtual machines (VMs), we suggest using nature-inspired techniques for efficient load balancing of tasks on VMs available in cloud computing. In this regard, particle swarm optimization (PSO), genetic algorithms (GA) and artificial bee colony (ABC) algorithms, BA algorithm

are widely used to improve the workload of hosting server. These methods, in particular ABC, are heavily based on a virtual machine policy to improve load balancing performance and avoid server overload [35]. Concretely, we rely on the BA-based approach [36], presented in [36] to select the appropriate VM instance for each task and assign a server present in the optimal cluster. Accordingly, the fitness function g () for each task K with the sizes SZk on the jth VM is based on the runtime (ETkj) and the cost of execution (ECkj), as shown by equation [37].

$$g(k,j) = \alpha\, ET_{kj} + (1-\alpha) EC_{kj} \quad (5)$$

$$ET_{kj} = \frac{SZ_k}{CP_j}$$

$$EC_{kj} = \frac{ET_{kj}}{\lambda} \times CO_j$$

where α is a time-cost balance factor in a range of [0 1]; COj refers to the cost of the type-j VM instance for a unit time (λ).

Technically, the proposed solution is composed of 3 main components: Clients, datacenter, smart load balancing module, as shown in **Figure 8**.

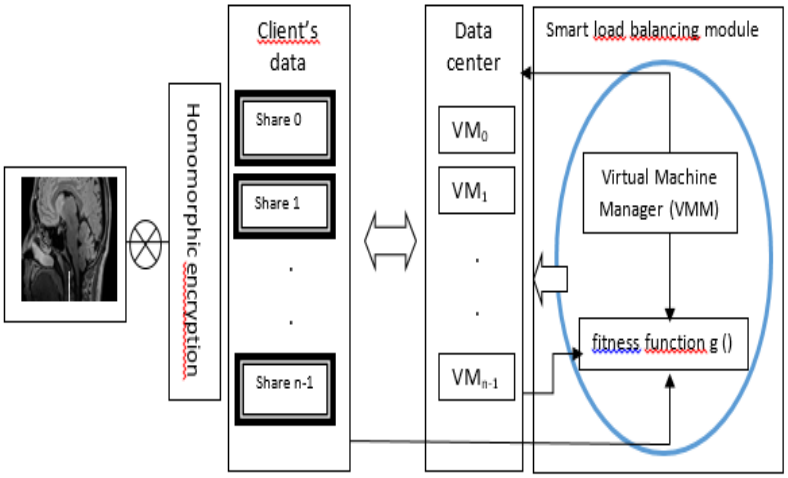

Figure 8 Load balancing policy.

Functionally, the intelligent module acquires information from client data and active servers. This useful information is used to calculate the fitness function. Therefore, the VMM module selects the appropriate virtual machine for each task. For this purpose, the VMM module is designed to continuously monitor changes in the active virtual machine, as well as task requests.

Based on these measurements, our proposal can be used to solve the runtime problem in homomorphic encryption. This would significantly improve the Quality of Service (QoS) in cloud services.

Conclusions

Using cloud services to analyze medical images is a new approach whereby the necessary imaging tools are provided to customers. This is because physicians rely on remote cloud applications rather than on-premise solutions. Therefore, this concept is an effective method that offers both cost savings and productivity gains. In fact, the key goal of cloud computing is to outsource IT services to an external third party. Despite its significant economic benefits, data privacy is a serious issue to consider. To this end, various techniques have been proposed to overcome this challenge. Homomorphic encryption is one. This study aims to explore the opportunities and obstacles to using this type of encryption for cloud-based

image processing. Apart from the homomorphic encryption still encounters several problems, in particular as regards the computation costs. In fact, this limiting factor has negative effects on the practical utility of cloud services. For this reason, we have proposed a new approach to secure the processing of externalized medical images via homomorphic encryption. Specifically, we used the partition technique with a multi-agent system to meet security and confidentiality requirements. The key idea of this solution was to split the input image into several small parts before encrypting them. In this case, we used the multi-agent system to support distributed data processing. Consequently, each region generated is analyzed by a set of agents belonging to a multi-agent system. To implement this solution, we proposed the Hadoop framework. In this case, we have used the MapReduce function to divide each task into several small tasks (subtask) in parallel to improve system performance. In addition, we have introduced a BA-based method to ensure efficient management of VM workload. Therefore, the proposal aims to strengthen the homomorphic approach in cloud services by ensuring both security and performance. As perspectives, we plan to test the proposed framework using the Paillier algorithm for encrypting medical images and the BA-based method for load balancing.

References

[1] R Parks, RT Wigand, MB Othmani, Z Serhier and O Bouhaddou. Electronic health records implementation in Morocco: Challenges of silo efforts and recommendations for improvements. *Int. J. Med. Inform.* 2019; **129**, 430-7.

[2] S Tanwar, K Parekha and R Evans. Blockchain-based electronic healthcare record system for healthcare 4.0 applications. *J. Inform. Secur. Appl.* 2020; **50**, 102407.

[3] J Haskew, G Ro, K Saito, K Turner, G Odhiambo, A Wamae, S Sharif and T Sugshita. Implementation of a cloud-based electronic medical record for maternal and child health in rural Kenya. *Int. J. Med. Inform.* 2015; **84**, 349-54.

[4] HN Moulick and M Ghosh. Medical image processing using a service oriented architecture and distributed environment. *Am. J. Eng. Res.* 2013; **2**, 52-62.

[5] F Farokhi, I Shames and N Batterham. Secure and private cloud-based control using semi-homomorphic encryption. *IFAC-PapersOnLine* 2016; **49**, 163-8.

[6] AM Vengadapurvaja, G Nisha, R Aarthy and N Sasikaladevi. An efficient homomorphic medical image encryption algorithm for cloud storage security. *Proc. Comput. Sci.* 2017; **115**, 643-50.

[7] A Lathey and PK Atrey. Image enhancement in encrypted domain over cloud. *ACM Trans. Multimed. Comput. Comm. Appl*. 2015; **11**, 38

[8] M Mohanty, WT Ooi and PK Atrey. Secret sharing approach for securing cloud-based pre-classification volume ray-casting. *Multimed. Tools Appl.* 2016; **75**, 6207-35.

[9] P Singh, B Raman, N Agarwal and PK Atrey. Secure cloud-based image tampering detection and localization using POB number system. *ACM Trans. Multimed. Comput. Comm. Appl.* 2017; **13**, 23.

[10] M Marwan, A Kartit and H Ouahmane. A cloud-based framework to secure medical image processing. *J. Mobile Multimed*. 2018; **14**, 319-44.

[11] M Marwan, A Kartit and H Ouahmane. A cloud-based solution for collaborative and secure sharing of medical data. *Int. J. Enterprise Inform. Syst*. 2018; **14**, 128-45.

[12] X Yi, R Paulet and E Bertino. *Homomorphic encryption and applications*. Springer, Cham, Switzerland, 2014.

[13] MM Potey, CA Dhote and DH Sharma. Homomorphic encryption for security of cloud data. *Proc. Comput. Sci.* 2016; **79**, 175-81.

[14] M Marwan, A Kartit and H Ouahmane. Applying homomorphic encryption for securing cloud database. *In*: Proceedings of the 4th IEEE International Colloquium on Information Science and Technology, Tangier, Morocco. 2016, p. 658-64.

[15] P Paillier. *Public-key cryptosystems based on composite degree residuosity classes*. *In*: J Stern (Ed.). Advances in cryptology - EUROCRYPT'99. Lecture notes in computer science. Vol 1592. Springer, Heidelberg, Germany, 1999, p. 223-38.

[16] RL Rivest, A Shamir and L Adleman. A method for obtaining digital signatures and public-key cryptosystems. *Comm. ACM* 1987; **21**, 120-6.

[17] P Martins, L Sousa and A Mariano. A survey on fully homomorphic encryption: An engineering perspective. *ACM Comput. Surv.* 2018; **50**, 83.

[18] C Gentry. 2009, A fully homomorphic encryption scheme. Ph. D. Dissertation. Stanford University, California, United States.

[19] MV Dijk, C Gentry, S Halevi and V Vaikuntanathan. *Fully homomorphic encryption over the integers*. *In*: H Gilbert (Ed.). Advances in cryptology - EUROCRYPT 2010. Lecture notes in computer science. Vol 6110. Springer, Heidelberg, Germany, 2010.
[20] VS Gorti and U Garimella. An efficient secure message transmission in mobile ad hoc networks using enhanced homomorphic encryption scheme. *Global J. Comput. Sci. Tech. Netw. Web Secur.* 2013; **13**, 20-33.
[21] RK Challa, J Kakinada, GV Kumari and B Sunny. Secure image processing using LWE based homomorphic encryption. *In*: Proceedings of the 2015 IEEE International Conference on Electrical, Computer and Communication Technologies, Coimbatore, India. 2015, p. 1-6.
[22] M Gomathisankaran, X Yuan and P Kamongi. Ensure privacy and security in the process of medical image analysis. *In*: Proceedings of the 2013 IEEE International Conference on Granular Computing, Beijing, China. 2013, p. 120-5.
[23] SR Kanithi and AG Latha. Secure image processing using discrete wavelet transform and paillier cryptosystem. *Int. J. Mag. Eng. Tech. Manag. Res*. 2015; **2**, 1270-6.
[24] N Habeep and RD Raj. Homomorphic encrypted domain with DWT methods. *Int. J. Inventions Comput. Sci. Eng*. 2014; **1**, 2348-3431.
[25] P Yang, X Gui, J An and F Tian. An efficient secret key homomorphic encryption used in image processing service. *Secur. Comm. Netw.* 2017; **2017**, 7695751.
[26] S Farah, MY Javed, A Shamim and T Nawaz. An experimental study on performance evaluation of asymmetric encryption algorithms. *Recent Adv. Inform. Sci.* 2012; **8**, 121-4.
[27] SJA Nair and TRG Nair. VM placement with effective energy management in cloud using optimal VM allocation framework (OVAF). *Indonesian J. Electr. Eng. Comput. Sci*. 2020; **18**, 1531-8.
[28] WARWM Isa, AIH Suhaimi, N Noordin, AF Harun, J Ismail and RA Teh. The factors influencing cloud computing adoption in higher education institution. *Indonesian J. Electr. Eng. Comput. Sci.* 2020; **17**, 412-9.
[29] S Deshpande and R Ingle. Preferences based customized trust model for assessment of cloud services. *International Journal of Electrical and Computer Engineering.* 2018; **8**, 304-25.
[30] J Mahdjoub, Z Guessoum, F Michel and M Herbin. A multi-agent approach for the edge detection in image processings. *In*: Proceedings of the 4th European Workshop on Multi-Agent System, Lisbon, Portugal. 2006.
[31] FDL Prieta, S Rodríguez, P Chamoso, JM Corchado and J Bajo. Survey of agent-based cloud computing applications. *Future Generat. Comput. Syst*. 2019; **100**, 223-36.
[32] SM Banaei and HK Moghaddam. Hadoop and its role in modern image processing. *Open J. Mar. Sci*. 2014; **4**, 239-45.
[33] MH Almeer. Cloud hadoop map reduce for remote sensing image analysis. *J. Emerg. Trends Comput. Inform. Sci.* 2012; **3**, 637-44.
[34] S Arietta, J Lawrence, L Liu and C Sweeney. Hipi-hadoop image processing interface, Available at: http://hipi.cs.virginia.edu/about.html, accessed January 2020.
[35] A Ullah, NM Nawi, J Uddin, S Baseer and AH Rashed. Artificial bee colony algorithm used for load balancing in cloud computing: Review. *IAES Int. J. Artif. Intell.* 2019; **8**, 156-67.
[36] XS Yang. A new metaheuristic bat-inspired algorithm. *In*: Proceedings of the Nature Inspired Cooperative Strategies for Optimization, Heidelberg, Germany. 2010.
[37] M Adhikari, S Nandy and T Amgoth. Meta heuristic-based task deployment mechanism for load balancing in IaaS cloud. *J. Netw. Comput. Appl.* 2019; **128**, 64-77.