Journal of Scientific & Industrial Research
Vol. 80, August 2021, pp. 692-698

Medical Data Analytics for Secure Multi-party-primarily based Cloud Computing utilizing Homomorphic Encryption

Naresh Sammeta[1#]* and Latha Parthiban[2]

[1]JNT University Kakinada, Kakinada 533 003, AP, India

[2]Dept. of Computer Science, Pondicherry University Community College, Pondicherry 605 014, India

[#]Dept. of Computer Science & Engineering, RMK College of Engineering and Technology, Chennai 601 206, Tamilnadu, India

Received 14 August 2020; revised 26 July 2021; accepted 30 July 2021

Cloud computing has emerged as a vibrant part of today's modern world, providing computer services such as data storage, managing and processing via the internet. For the most part, cloud applications emphasize a multi-tenant structure to provide support for several customers in a single instance. A multi-tenancy situation involving the allocation of resources in cloud storage and the risks associated with it, in which confidentiality or integrity may be compromised. Homomorphic encryption is one such technique which guarantees to franchise in safeguarding information under cryptographic domain. The proposed modified Algebra Homomorphic Encryption scheme based on updated ElGamal (AHEE) encryption scheme is designed in such a way that the cloud administrators do not obtain any information about the medical data. This scheme is quantitatively evaluated using metrics such as encryption time and decryption time. The experimental results using UCI Machine Learning Repository ECG data set show that the proposed scheme achieved shorter encryption time of 6.61 ms and decryption time of 5.94 ms and also analyze this secured datum using big data analytics.

Keywords: AHEE, CSP, Decryption time, Encryption time, Homomorphic encryption

Introduction

Living in 21st century, an eon of cloud, automation and virtual realities, one might never forget the necessity of security and privacy because information breaching has become professional or unprofessional act especially in healthcare firms. In 2017, an insider of Kentucky, Bowling Green based Med Centre Health, owned by Common-wealth Health Corp stated that 698,000 individuals were affected as part of the breach.[1] The key cause of this hack was cloud and local storage with simple encryption. Such issues can be solved with homomorphism, an enciphering technique that creates equivalent forms or preserves the properties and relations even after encryption. This paper aims to ameliorate the standard of advancement security and privacy in the area of cloud breaking traditional methodologies used in native homomorphism by imbibing other arithmetic operators and creating a new relation between them. Research-scientist in the field of homomorphism says that this idea/ concept is too difficult to implement and make work.[2] Considering their dubious remarks as a Kickstarter and visionary to test perseverance, the paper surfed through and sailed mighty oceans to acclaim credit and praise.

Rag or riches, people of all walks of life have arcane matters and this sensitive, confidential, private or unfamiliar information can be left open if no proper steps have been taken to protect them from any fraudulent source. Analysis in property preserved encrypted ciphertext can be done to prove that this data is the same even after encryption. Also, the analyzed report obtained from visualized graph can be sent back to user to understand the same. Through this, datum of various industries can be sent to other reliable or unreliable sources without fear or breach of data. Therefore, the ultimate aim of this paper is to diversify and think of security with privacy in cloud, a desideratum of every individual surviving and fighting to survive in this world.

Related Work

A variety of analysts, including the study of security problems for cloud infrastructure distribution models, recognized multi-tenancy[3] as confidentiality problems in cloud computing and claimed that multi-tenancy is a notable trademark in the cloud that may

*Author for Correspondence
E-mail: samnaresh@gmail.com

lead to confidentiality loss. In the paper, authors give protection use key control technologies and encryption. In any case, with regards to the cloud[4], overseeing and ensuring the encryption keys operable is as critical as encoding the information. This paper analyses the significance of security to cloud. The authors looked at Algebra Homomorphic Encryption scheme based on updated ElGamal (AHEE) for Data security in cloud.[5] They are evaluated with four characters in mind: used keys, scalability, protection and forms of authentication. In this scheme it has been shown that medical data processing is carried out using microservice protection mechanisms and that it needs to provide adequate security and privacy issues to be legitimate and widely recognized health care. The management of private patient information by respectable hospitals is rising in data centers.[6] A methodology that does not subsume decryption secretly conserves data during an age of homomorphism. This program uses Amazon as a hypothesis proof to perform the study of the patient's security and the information it took is an encrypted healthcare data which in effect generated encrypted cardiovascular disease. We use Fully Homomorphic Encryption (FHE) and provide the user with a collection of parameters to pick their preferred FHE form.

The world of encryption was converting plain information from one recognizable format to another, but the pertinacity of making the world's data a more secured one made the author bloom out with a new idea called as homomorphic encryption.[7] PHR data is encrypted prior to outsourcing and searches of encrypted data and keeps the searchable property for the effective recovery of health files using the range query after encryption.[8] This kind of encryption mainly focuses on preserving the properties and relations of the data even after encryption. Data center protection using a dynamic encryption technique as communication and cloud access services for cloud environment protection based on stable hash keys, an efficient algorithm solution is applied.[9] The EHR data has exponentially increased. The amount of storage space must be reduced by effective storage of the records. At the target stage an appropriate solution is proposed to the weight-based deduplication scheme to reduce the unwanted computing storage in the cloud.[10] The keyword binning is proposed an effective randomisation of the query system where record indices are allocated to different buckets depending on the contained keys and a search can only be carried out in the corresponding buckets.[11] The American Scientist proved its possibility by explaining with typical RSA algorithm and a cryptosystem called as Gentry's cryptosystem.[12] The power of possibilities of using such techniques brainstormed many security specialists who in turn created various flavors of it. The scientist made use of FHE where it makes use of two operators generally addition and multiplication. Thus, this paper was indeed one of the most important papers used. Systems these days utilize one of the two strategies of native homomorphism, i.e., partial homomorphism: a technique that uses only one operator either addition or multiplication and fully homomorphism makes use of two operators usually addition and multiplication.[13–17] Examples of algorithm which are partially homomorphic are unpadded RSA, ElGamal, Goldwasser Micali, Paillier and fully homomorphic are Gentry's Cryptosystem, cryptosystem over integers and so on.[18,19] Though they preserve the properties, each of them suffers from any of these listed below.

- Insecure, using one operator as an operative key can be guessed and is more likely to compromise the key through brute force attack.
- It is prone to malleability as equivalent forms after full homomorphism illustrates a particular pattern.

Homomorphic Encryption

Homomorphic encryption is the perfect way to address security threats in cloud storage as its solutions allow encrypted data to be computed without the appropriate secret key to decrypt information. In 2009, the first FHE was attempted by Craig Gentry.[20] Gentry proposed a homomorphic coding system for the broad use of cloud computing in March 2010. Dijk *et al.*[21] proposed a second FHE. Unfortunately, these devices are detrimental to cloud computers. Zhang *et al.* [22] in 2014 proposed a robust cloud computing solution based on the homomorphic property of Paillier. Kocabas and Soyata[23] deployed in 2015 the privacy security scheme for medical cloud computing with absolute homomorphic encryption. Ren *et al.*[24] (2016) suggested XOR encoding for homomorphism to help stable keyword encryption searches cloud storage detail.

Homomorphic encryption is a method of encoding that can be done in various calculations by ciphertext, and that returns a crypt result, which is decrypted in the same way as the plain text operation e.g., if the two numbers 30 and 15 are accessible then both are encrypted to 038077112334522722687414758259114600 and 2659658612023623158221708107731229 75 respectively, the add-on operator shows the 188448688182983682025920831261484929 value, which is decrypted to be 45.

In the proposed model the provider nearly is in responsibility of the majority of the security controls inside the cloud. The customers are in charge of Data security and Analyzer for Data grouping. Likewise, customer is in charge of dealing with their users, and end point devices. Customers ought to guarantee that the provider satisfies its SLAs and meets security, protection, Data ownership, uptime and execution prerequisites.

Au courant promised system takes care of every minute detailed dimension of security to provide an invincible and impervious set of enciphered data. As previously mentioned, it makes use of three keys, a blend of symmetric and asymmetric strategies to protect data, which no prevailing algorithm makes use of. Also, it makes use of four arithmetic operators in its operations to make a complex figure which looks obsolete if he/she does not have the right credentials. Analyzes and assessments may be carried out without deciphering the details for an upgraded framework. In this case time series, seasonal analysis is used for the data of Survival Months and the advantages of proposed systems in Fig. 1.

The new system meticulously uses three unique keys (public, private and secret key between two entities) and provides added security and creates a new relation between addition and multiplication and imbibes it as an encoding factor (neologism indeed). Unassailable as encryption is done in local systems and results of counterpart's text is uploaded in various cloud environments and practical approach to understand the working (only if the key values are known); one step encryption and no mathematical computing decryption.

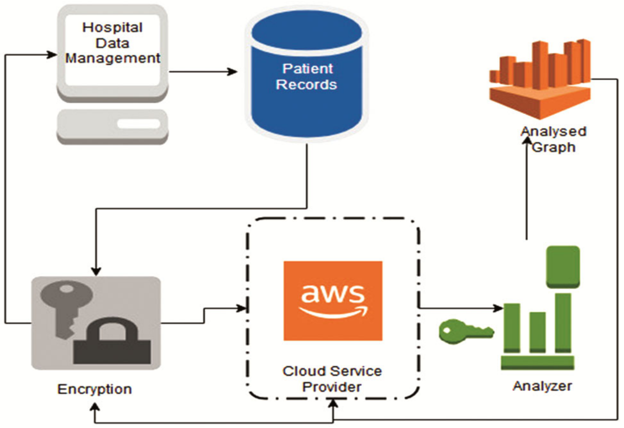

Fig. 1 — EHR system architecture- health information exchange

The system aims to reduce the stress and anxiety of various healthcare firms when trying sending original data for different rating polls to the newspapers and the news communication industry. It creates a more secure equivalent set of data that will work to produce results similar to that of the original datum after analysis. The system can be split into modules which embeds a coalescing binder module too. The modules are data collection and conversion, encryption using modified AHEE algorithm and upload to cloud, performing analysis on the encrypted e-health data.

Data Collection and Conversion

Generally, hospitals or any naive user makes use of tables and excel sheets to store data and which can further be converted into Comma-Separated-Values (CSV) format. To spot the cyber bullying in the data sets ALBERT-based fine-tuning model is proposed by Tripathy *et al.*[25] is a transformer-based architecture and thus has greater spatial interpretation in its untrained nature than other recurring units. A simple step in the User Interface (UI) converts humongous compilation to the expected CSV format and split the Most and the Least Confidential Data. The data set will be split into two files and the health record / dataset input will be read, and the output will be split into patient information and medical records.

Encryption Using Hybrid Algorithm and Updating using Cloud

Two data (Medical Records and Patient Details) in Fig. 2. should be authenticated utilizing the Modified AHEE method and the result would be a homomorphic authenticated file which can be accessed through the cloud.

Step1: Read the dataset, value by value.

Step2: Encrypt them using the basic enciphering technique by using modified AHEE[14] as below. This method is shown to be secure because by selecting the random number k in E_1, will avoid plaintext attacks.

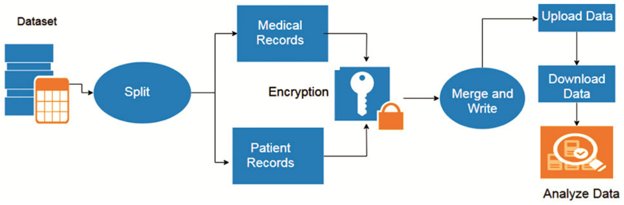

Fig. 2 — Encrypting the e-health data and updating in cloud

Modified AHEE (Proposed) - Key Generation

Step 1: select any three prime numbers – p,q and r, large key size preferably at least 224-bit key size.

Step 2: computing their system modulus N= p*q*r

Step 3: select a random number h and a root *g* of GF(p) where h, g < p.

Step 4: calculate $y = g^h$ mod p.

Encryption

Input message M is made up of multiple bits $m_1, m_2, \ldots m_n$

Step 1: Select a random integer number *i*
E_1 (M) = (M+i*n) mod N.

Step 2: Choose a random integer *k* such that:
E_r (M) =c= (a,b) = (g^k mod n, y^k E_1 (M)mod n)

Decryption

Input message C is made up of multiple bits c_1, $c_2 \ldots c_3$

$M = b \times (a^h)^{-1}$ (mod p)

Output a message is $M = m_1, m_2, \ldots m_n$

Upload the resultant dataset to the cloud

Perform Analysis on e-Health Data

Analyzer can import the dataset from the cloud and interpret the report based on the dataset of the hospital through the graph.

Experimental Results and Discussion

The efficiency of the proposed benign exchange of electronic health records utilising modified AHEE was evaluated with valuation measures in this performance analysis portion. This portion further offers an example of the efficiency of the proposed approach with existing test data and comparative study. A software package was used to build the proposed architecture (python). In the following segment the analyses are stated of the comparative analysis.

Simulation Results

The dataset used during the experiment was provided by Dua and Graff 2019.[26] There would be a contrast between these various methods, given the same data as input. The relation among the five algorithms is studied by changing one parameter at a time and remains unchanged in the other parameter. The calculation in Table 1 is mainly focused on the prime number, the meaning of which is dependent on the number of bits used to generate the integer.

In Fig. 3 the performance of the proposed modified AHEE scheme shows to be the best compared to that of RSA, Paillier, ElGamal and AHEE. The average value of encryption time for the different data sizes is 56.97 ms and decryption time is 54.5 ms for the RSA algorithm. Whereas for the Paillier algorithm encryption time is 23.54 ms, decryption time is 22.67 ms. ElGamal algorithm achieves 20.04 ms encryption time and 18.94 ms decryption time. Encryption time for AHEE algorithm is 17.84 ms whereas decryption time is 17.52 ms. Rather, the proposed modified AHEE scheme achieves lowest encryption time of 6.61 ms and 5.94 ms for decryption time. We thus infer, in terms of encryption time and decryption time that the proposed scheme performs better.

Table 1 — Comparison of Encryption and Decryption time of cryptographic algorithms (in ms)

Metrics	File size (MB)	RSA	Paillier	ElGamal	AHEE	Proposed
Encryption Time	5	3082	1276	1083	938	382
	10	5938	2592	2094	1924	769
	15	8927	3429	3182	2681	954
	20	11948	4662	4049	3731	1342
	25	13865	6125	4983	4429	1628
Decryption Time	5	2934	1219	991	882	354
	10	6139	2481	1893	1892	698
	15	8625	3376	3054	2723	903
	20	11284	4421	3874	3692	1176
	25	12879	5921	4739	4274	1432

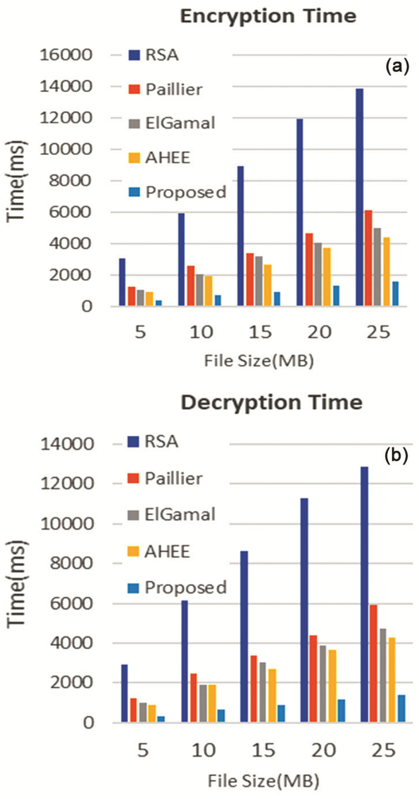

Fig. 3 — comparison of (a) encryption time and (b) decryption time

Data Management Analysis

Visualization of the analyzed data can be seen using tableau public. Firstly, the dataset to be encrypted is loaded (be it from a local machine or remote data server/ repository such as UCI, Kaggle etc.) in cloud. In the next stage, the serving end (such as hospital management) gets to know the public key of the analyst using a third-party authenticator or tries getting a symmetric key using Diffie Hellman key exchange.[27] Along with this key (the recently fetched analyst's key) the server (serving side) generates its own public and private key, the main ingredient of this spark by using modified AHEE. Every hospital must transfer its authenticated records to a cloud-based computer network under the public key of an analyst. Now the encrypted content has been modified to a unique way that is a way which retains all the structural properties and operations it possessed before encryption. The encrypted data can now be stored in public clouds or can be shared to any individual or a group. Any kind of analyzation/ prediction can be done on this data. In general, any kind of data such as data about financial, voting, defence technology, new bots in the market can be encrypted to the preferred form using this technique. Invalid users will not gain access to the dataset as the algorithm provides a provision to check the authenticity of the end user. The type of analyzation considered according to this paper is to understand the efficiency of cardiologists and hospitals based on surgeries performed annually. Considering parameters such as pericardial_effusion, wall_motion_score, survival ratio will determine the completeness and competence of doctors in services. The analyst retrieves the document by entering his private key. The relation between the public key will persist as that is immutable. The program to decrypt will be present in the cloud as an executable file. The analyzer needs to run the program and would require the basic prerequisites which is free of cost. If the password/key is right then the analyst can download a copy of the dataset. After analysis, the result will be uploaded in cloud and the report(text) will be hidden inside the resultant output. This is none other than steganography. The most critical thing to note, is the confidential data such as personal details will still be encrypted (double) and cannot be decrypted at any cost. The state of the fact that decryption is not possible because it has been encrypted using the client side's public key as to decrypt it will require the private (private key will be known only to one analyzation, can be done using various tools. Out of which we have chosen Python StatsModels[28], enables app data to be analyzed, statistical experiments conducted and mathematical models calculated. Matplotlib[29] is a Python programming language plotting module for 2D plots. So now in a situation where we used updated modified Algebra individual).

Homomorphic encryption algorithm to encrypt and decrypt, we require the client to use their credentials to access non-confidential data such as specific valve or wall motion score ranges or even wall activity index. In Fig. 4 column wise data can be stripped, and predictions can be done.

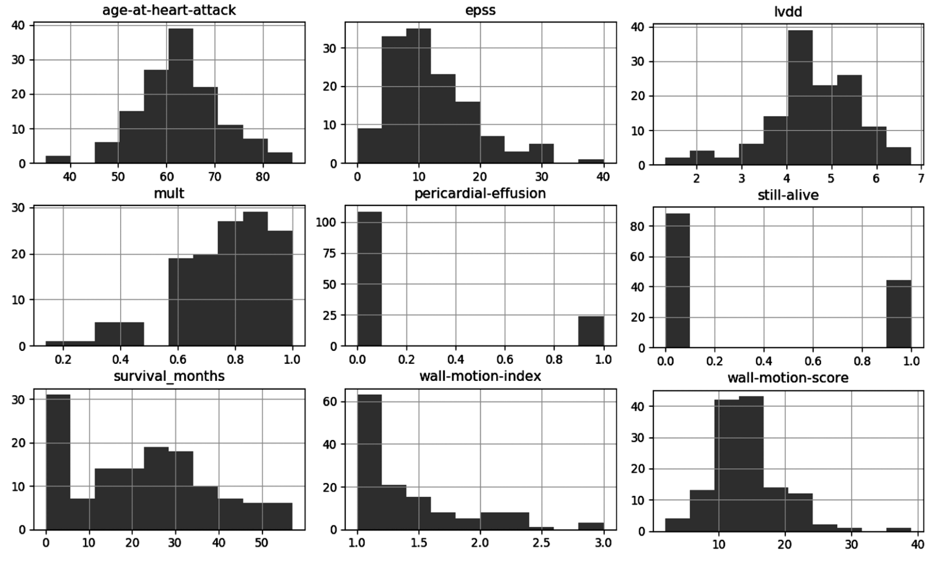

Fig. 4 — Resultant graph analyzed by the analyst without knowing the patient details

The analyst performs various time series operations. In time series, this paper focuses mainly on seasonal value. The analyst can also be from media, as most of them conduct a ranking survey every year to help public understand the standard of each hospital. The reason to why this has to be sent only in encrypted format is the hospitals need to follow their medical ethics of not sharing any details of the patients for any purpose (even the person is a donor). So, in that case how will they send it to the media and that for a commercial purpose as predictions such as successful surgeries that will occur, doctor wise comparisons and many more?

Conclusions

In this work, we have designed and presented a new scheme for ensuring the scalable and secure transfer of EHR data. In this scheme using modified AHEE where the most confidential data gets hidden and acts as black glove where one knows data exist but decrypting or bringing to equivalent form is unattainable. Even if gaps persist the data inside is impermissible. Using this encryption one can send equivalent data to anybody in the world to perform the analysis, especially in the case of surveys and awards. Space/Memory requirements are quite less as the data can be accessed from any remote server using cloud. It does not suffer from malleability as the hacker will not know the strength of the key. The hacker might get irrelevant data even if brute force unethical technique is used. The paper proposed can further be extended like checking a person's validity using fingerprints, iris/pupil tracker and many more. Using other real time datasets such as cancer, tuberculosis, cholera, or any other disease one can perform enormous prediction that will help save a million lives. All these will add feathers to the cap by making the idea more powerful.

References

1 www.careersinfosecurity.com/hacking-incidents-dominate-2017- health-data-breach-tally-a-10424

2 Dowlin N, Gilad-Bachrach R, Laine K, Lauter K, Naehrig M & Wernsing J, Manual for Using Homomorphic Encryption for Bioinformatics, *Proc IEEE*, (2017) 1–16.

3 Karatas G, Can F, Dogan G, Konca C & Akbulut A, Multi-tenant architectures in the cloud: A systematic mapping study, *Int Artif Intell Data Process Symp* (IDAP), Malatya, (2017) 1–4.

4 Kavitha R, Kannan N, Nazneen R & Jubar H A, Cloud computing integrated with testing to ensure quality, *J Sci Ind Res*, **75**(**2**) (2016) 77–81.

5 Xiang G, Yu B & Zhu P, A algorithm of fully homomorphic encryption, 2012 9th *Int Conf Nat Comput Fuzzy Syst Knowl Discov*, Sichuan, (2012) 2030–2033.

6 Barrows R C & Clayton P D, Privacy, confidentiality, and electronic medical records, *J Am Med Inform Assoc*, **3**(**2**) (1996) 139–148.

7 Aguilar Melchor C, Fau S, Fontaine C, Gogniat G & Sirdey R, Recent Advances in Homomorphic Encryption: A Possible Future for Signal Processing in the Encrypted Domain, *IEEE Signal Process Mag*, **30**(**2**) (2013) 108–117.

8 Sangeetha D, Chakkaravarthy S S & Satapathy S C, Vaidehi V, Cruz M V, Multi keyword searchable attribute-based encryption for efficient retrieval of health records in cloud, *Multimed Tools Appl* (2021), https://doi.org/10.1007/s11042-021-10817-z.

9 Kumar P, Gupta A & Kumar S, Dynamic key based algorithm for security in cloud computing using soft computing and dynamic fuzzy approach, *J Sci Ind Res*, **78**(**9**) (2019) 596–600.

10 Pugazhendi E, Sumalatha M & Lakshmi Harika P R, Weight based deduplication for minimizing data replication in public cloud storage, *J Sci Ind Res*, **80**(**3**) (2021) 260–269.

11 Handa R, Rama Krishna C & Aggarwal N, Keyword binning-based efficient search on encrypted cloud data, *Arab J Sci Eng*, **44**(**4**) (2019) 3559–3584.

12 Gentry C, Fully Homomorphic Encryption Using Ideal Lattices, *Proc of the 41st Annual Proc. Annu. ACM Symp Theory Comput* (STOC'09) (ACM Press, New York, NY, USA) 2009, 169–178

13 Rivest R, Adleman L & Dertouzos M, On data banks and privacy homomorphisms, *Int J Found Comput Sci*, **4**(**11**) (1978), 169–180.

14 Elgamal T, A Public key cryptosystem and a signature scheme based on discrete logarithms, edited by Blakley G & Chaum D, *IEEE Trans Inf Theory*, **31** (1985) 469–472.

15 Goldwasser S, Micali S, Probabilistic encryption, *J Comp Syst Sci*, **28**(**2**) (1984) 270–299.

16 Boneh D, The decision Diffie-Hellman problem, *Proc Third Int Symp* (ANTS-III) (1998), 4863.

17 Paillier P & Stern J, Public-key cryptosystems based on composite degree residuosity classes, *Advances in Cryptology* (Springer-Verlag, Berlin, Germany) **1592** (1999) 223–238.

18 Dijk M V, Gentry C, Halevi S, Vaikuntanathan V & Gilbert H, Fully homomorphic encryption over the integers, *Advances in Cryptology* (Springer-Verlag, Berlin, Germany) **6110** (2010), 24–43.

19 Gentry C, Toward basing fully homomorphic encryption on worst-case hardness, *Advances in Cryptology Proc of CRYPTO'10, LECT NOTES COMPUT SC*, Springer-Verlag, 6223 (2010) 116–137.

20 Gentry C, Computing arbitrary functions of encrypted data, *Commun ACM*, **53**(**3**) (2010) 97–105.

21 Dijk M V, Gentry C, Halevi S & Vaikuntanathan V, Fully homomorphic encryption over the integers, *Advances in Cryptology – EUROCRYPT 2010*, (2010) 24–43.

22 Zhang Y, Zhuo L, Peng Y & Zhang J, A secure image retrieval method based on homomorphic encryption for cloud computing, *19th Int Conf Digit Signal Process* (2014).

23 Kocabas O & Soyata T, Utilizing homomorphic encryption to implement secure and private medical cloud computing, *8th IEEE Int Conf Cloud Comput* (2015).

24 Ren S Q, Tan B H, Sundaram S, Wang T, Ng Y, Chang V & Aung K M M, Secure searching on cloud storage enhanced by homomorphic indexing, *Future Gener Comput Syst*, **65** (2016) 102–110.

25 Tripathy J K, Chakkaravarthy S S, Satapathy S C, Sahoo M & Vaidehi V, ALBERT-based fine-tuning model for cyberbullying analysis, *Multimed Syst* (2020).

26 Dua D & Graff C, UCI Machine Learning Repository [http://archive.ics.uci.edu/ml]. Irvine, CA: University of California, School of Information *and Computer Science* (2019).

27 Diffie W & Hellman M, New Directions in Cryptography, *IEEE Trans Inf Theory*, **22**(**6**) (1976) 644–654.

28 www.statsmodels .org /stable/index.html

29 www.matplotlib.org/users /pyplot_tutorial.html.